

# **LAPORAN KERJA PRAKTEK**

## **APLIKASI ASET DENGAN CODEIGNITER VERSI 3.1.4 DAN BOOTSTRAP VERSI 3.3.7 UNTUK MEMUDAHKAN PENGAJUAN PROPOSAL ASET LABORATORIUM KOMPUTER**

Diajukan untuk memenuhi persyaratan kelulusan  
Matakuliah TIF 335 Kerja Praktek

Oleh:

**ANDIKA DWIPUTRA/ 301180004**



**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS BALE BANDUNG  
2021**

**LEMBAR PENGESAHAN**  
**PROGRAM STUDI TEKNIK INFORMATIKA**  
**APLIKASI ASET DENGAN CODEIGNITER VERSI 3.1.4 DAN**  
**BOOTSTRAP VERSI 3.3.7 UNTUK MEMUDAHKAN PENGAJUAN**  
**PROPOSAL ASET LABORATORIUM KOMPUTER**

Oleh:  
ANDIKA DWIPUTRA/ 301180004

disetujui dan disahkan sebagai  
**LAPORAN KERJA PRAKTEK**

Bandung, .....  
Koordinator Kerja Praktek

Yusuf Muharam, M.Kom.  
NIK: 04104820003

**LEMBAR PENGESAHAN**  
**LABORATORIUM KOMPUTER FTI UNIBBA**  
**APLIKASI ASET DENGAN CODEIGNITER VERSI 3.1.4 DAN**  
**BOOTSTRAP VERSI 3.3.7 UNTUK MEMUDAHKAN PENGAJUAN**  
**PROPOSAL ASET LABORATORIUM KOMPUTER**

Oleh:  
ANDIKA DWIPUTRA/ 301180004

disetujui dan disahkan sebagai  
**LAPORAN KERJA PRAKTEK**

Bandung,.....  
Kepala Laboratorium Komputer FTI UNIBBA

Mochamad Ridwan, S.T.  
NIK: 04104808007

## ABSTRAKSI

Kerja Praktek dilaksanakan di Laboratorium Komputer FTI UNIBBA, mulai tanggal 16 Oktober 2021 sampai dengan tanggal 16 Desember 2021. Kerja praktek yang dilakukan adalah Pembuatan Aplikasi Aset Dengan Codeigniter 3.1.4 dan Bootstrap 3.3.7 Untuk Memudahkan Pengajuan Proposal Aset Laboratorium Komputer. Fitur yang ada dalam aplikasi tersebut antara lain data peralatan, merek, kategori, ruang, laboran, kompetensi, login, pengadaan, perbaikan dan data pengembalian. Selama pembuatan aplikasi, metodologi yang digunakan adalah *Software Development Life Cycle (SDLC)* dengan menggunakan model *Waterfall*. Tahap pertama adalah *Requirements Analysis and Definition* yakni mengumpulkan kebutuhan secara lengkap kemudian di analisis. Tahap kedua adalah *System and Software Design* yakni mengalokasikan kebutuhan-kebutuhan sistem baik perangkat keras maupun perangkat lunak. Tahap ketiga adalah *Implementation* yakni pengkodean dari desain ke dalam suatu bahasa pemrograman. Tahap keempat adalah *Integration and System Testing* yakni program digabung dan diuji sebagai sebuah sistem lengkap untuk memastikan apakah sesuai dengan kebutuhan perangkat lunak atau tidak. Tahap terakhir adalah *Operation and Maintenance* yakni tahapan yang paling panjang, sistem dipasang dan digunakan secara nyata. Pada akhir kerja praktek telah berhasil menciptakan sebuah aplikasi untuk mengelola aset laboratorium dan membantu Fakultas Teknologi Informasi di Universitas Bale Bandung khususnya di laboratorium komputer dalam mengelola data aset laboratorium. Kesimpulan dari keseluruhan proses kerja praktek adalah dengan dibuatnya Aplikasi Aset ini dapat mempermudah pendataan dan akses informasi data aset laboratorium komputer serta mempermudah dalam pengajuan proposal aset laboratorium komputer ke pihak Yayasan dan Rektorat Universitas Bale Bandung.

Kata Kunci: *Aplikasi, Laboratorium, CodeIgniter, Bootstrap, Aset*

## KATA PENGANTAR

Puji syukur alhamdulillah penulis panjatkan kehadiran Allah SWT yang telah melimpahkan rahmat-Nya serta hidayah-Nya sehingga penulis dapat menyelesaikan laporan kerja praktek dengan judul “Aplikasi Aset Dengan Codeigniter Versi 3.1.4 Dan Bootstrap Versi 3.3.7 Untuk Memudahkan Pengajuan Proposal Aset Laboratorium Komputer”.

Laporan ini dibuat dengan berbagai observasi dan beberapa bantuan dari berbagai pihak untuk membantu menyelesaikan tantangan dan hambatan selama mengerjakan laporan ini. Oleh karena itu, penulis mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Bapak Yudi Herdiana, S.T., M.T. selaku Dekan Fakultas Teknologi Informasi sekaligus dosen pembimbing kerja praktek.
2. Bapak Yusuf Muharam, M.Kom. selaku Ketua Program Studi S1 Teknik Informatika Universitas Bale Bandung sekaligus Koordinator Kerja Praktek.
3. Bapak Mochamad Ridwan, S.T. selaku kepala Laboratorium Komputer FTI UNIBBA sekaligus pembimbing lapangan.
4. Sahabat sekalian mahasiswa angkatan 2018 yang telah memberikan do'a dan mendorong semangat untuk penyusunan laporan ini.
5. Kedua orang tua yang tidak pernah letih memberikan bimbingan, dukungan, kepercayaan, dan do'a serta nasehatnya untuk keberhasilan penulis.

Laporan ini diharapkan bisa menjadi referensi untuk semua pihak serta bermanfaat bagi yang membutuhkannya. Dan juga Penulis mohon maaf apabila terdapat kesalahan-kesalahan serta kekurangannya.

Salam,  
Penulis

## DAFTAR ISI

<b>ABSTRAKSI</b> .....	i
<b>KATA PENGANTAR</b> .....	ii
<b>DAFTAR ISI</b> .....	iii
<b>DAFTAR GAMBAR</b> .....	v
<b>DAFTAR TABEL</b> .....	vii
<b>BAB I PENDAHULUAN</b> .....	1
I.1    Latar Belakang .....	1
I.2    Lingkup .....	2
I.3    Tujuan .....	3
<b>BAB II LINGKUNGAN KERJA PRAKTEK</b> .....	4
II.1    Struktur Organisasi .....	4
II.2    Lingkup Pekerjaan .....	7
II.3    Deskripsi Pekerjaan .....	7
II.4    Jadwal Kerja .....	8
<b>BAB III TEORI PENUNJANG KERJA PRAKTEK</b> .....	9
III.1    Teori Penunjang .....	9
III.2    Peralatan Pembuatan Aplikasi .....	16
<b>BAB IV PELAKSANAAN KERJA PRAKTEK</b> .....	34
IV.1    Input .....	34
IV.2    Proses .....	35
IV.2.1    Eksplorasi .....	35
IV.2.2    Pembangunan Perangkat Lunak .....	36
IV.2.3    Pelaporan Hasil Kerja Praktek .....	84
IV.3    Pencapaian Hasil .....	84
<b>BAB V PENUTUP</b> .....	90
V.1    Kesimpulan dan Saran Mengenai Pelaksanaan .....	90
V.1.1    Kesimpulan Pelaksanaan Kerja Praktek .....	90
V.1.2    Saran Pelaksanaan Kerja Praktek .....	90
V.2    Kesimpulan dan Saran Mengenai Aplikasi Aset Laboratorium Komputer .....	91

V.2.1	Kesimpulan Aplikasi Aset Laboratorium Komputer .....	91
V.2.2	Saran Aplikasi Aset Laboratorium Komputer .....	91
<b>DAFTAR PUSTAKA.....</b>		<b>ix</b>

## DAFTAR GAMBAR

Gambar II.1 Struktur Organisasi FTI UNIBBA.....	4
Gambar III.1 Model Waterfall .....	11
Gambar III.2 XAMPP .....	17
Gambar III.3 htdocs .....	18
Gambar III.4 phpMyAdmin .....	18
Gambar III.5 <i>Control Panel</i> .....	18
Gambar III.6 Logo Visual Studio Code.....	22
Gambar III.7 Logo/ Icon Google Chrome .....	24
Gambar III.8 Logo CodeIgniter.....	26
Gambar III.9 Logo Bootstrap.....	29
Gambar III.10 Logo/icon Balsamiq.....	32
Gambar III.11 Microsoft Visio.....	33
Gambar IV.1 Use Case Diagram (Awaluddin dkk., 2020) .....	38
Gambar IV.2 <i>Acitivity Diagram Tentang FTI dan Login</i> .....	61
Gambar IV.3 <i>Activity Diagram</i> Peralatan .....	62
Gambar IV.4 <i>Activity Diagram</i> Kategori .....	63
Gambar IV.5 <i>Activity Diagram</i> Ruang .....	64
Gambar IV.6 <i>Activity Diagram</i> Merek.....	65
Gambar IV.7 <i>Activity Diagram</i> Kompetensi.....	66
Gambar IV.8 <i>Activity Diagram</i> Perbaikan.....	67
Gambar IV.9 <i>Activity Diagram</i> Pengembalian .....	68
Gambar IV.10 <i>Activity Diagram</i> Pengadaan .....	69
Gambar IV.11 <i>Activity Diagram</i> Laboran (Awaluddin dkk., 2020) .....	70
Gambar IV.12 <i>Class Diagram</i> (Nuraeni dkk., 2017).....	71
Gambar IV.13 Struktur Menu (Rahayu S., 2018) .....	72
Gambar IV.14 Desain Tampilan Antarmuka Home .....	72
Gambar IV.15 Desain Tampilan Antarmuka Tentang FTI UNIBBA.....	73
Gambar IV.16 Desain Tampilan Antarmuka Login.....	74
Gambar IV.17 Desain Tampilan Antarmuka Dashboard .....	75
Gambar IV.18 Desain Tampilan Antarmuka Data Peralatan.....	76
Gambar IV.19 Desain Tampilan Antarmuka Data Kategori.....	77



Gambar IV.20	Desain Tampilan Antarmuka Data Ruang .....	78
Gambar IV.21	Desain Tampilan Antarmuka Data Merek .....	78
Gambar IV.22	Desain Tampilan Antarmuka Data Kompetensi .....	79
Gambar IV.23	Desain Tampilan Antarmuka Data Perbaikan .....	79
Gambar IV.24	Desain Tampilan Antarmuka Data Pengembalian .....	80
Gambar IV.25	Desain Tampilan Antarmuka Data Pengadaan.....	80
Gambar IV.26	Desain Tampilan Antarmuka Data Laboran.....	81
Gambar IV.27	Database Peralatan .....	84
Gambar IV.28	Database Login.....	84
Gambar IV.29	Database Pengadaan .....	85
Gambar IV.30	Database Pengembalian .....	85
Gambar IV.31	Database Ruang.....	85
Gambar IV.32	Database Perbaikan .....	85
Gambar IV.33	Database Laboran .....	85
Gambar IV.34	Database Merek .....	86
Gambar IV.35	Database Kategori.....	86
Gambar IV.36	Database Kompetensi .....	86
Gambar IV.37	Relasi Database Aplikasi Aset Laboratorium Komputer ..	86
Gambar IV.38	Tampilan Home .....	87
Gambar IV.39	Tampilan Tentang FTI UNIBBA .....	87
Gambar IV.40	Tampilan Login .....	88
Gambar IV.41	Tampilan Dashboard .....	88
Gambar IV.42	Tampilan Kategori.....	88
Gambar IV.43	Tampilan Ruang .....	89
Gambar IV.44	Tampilan Merek.....	89
Gambar IV.45	Tampilan Kompetensi .....	89

## DAFTAR TABEL

Tabel II.1 Waktu kegiatan kerja praktek.....	8
Tabel III.1 Simbol <i>Use case diagram</i> .....	13
Tabel III.2 Simbol <i>Activity Diagram</i> .....	14
Tabel III.3 Simbol <i>Class Diagram</i> .....	15
Tabel IV.1 Kebutuhan Perangkat Keras.....	36
Tabel IV.2 Kebutuhan Perangkat Lunak .....	36
Tabel IV.3 Skenario <i>Use Case</i> Tentang FTI .....	38
Tabel IV.4 Skenario <i>Use Case</i> Login .....	38
Tabel IV.5 Skenario <i>Use Case</i> Memeriksa Status Login .....	39
Tabel IV.6 Skenario <i>Use Case</i> Menampilkan/Melihat Data .....	39
Tabel IV.7 Skenario <i>Use Case</i> Mengubah Data Peralatan .....	39
Tabel IV.8 Skenario <i>Use Case</i> Menghapus Data Peralatan .....	40
Tabel IV.9 Skenario <i>Use Case</i> Mencetak/Export Data Peralatan .....	41
Tabel IV.10 Skenario <i>Use Case</i> Menambah Data Merek .....	41
Tabel IV.11 Skenario <i>Use Case</i> Mengubah Data Merek .....	42
Tabel IV.12 Skenario <i>Use Case</i> Menghapus Data Merek.....	43
Tabel IV.13 Skenario <i>Use Case</i> Menambah Data Kategori.....	43
Tabel IV.14 Skenario <i>Use Case</i> Mengubah Data Kategori .....	44
Tabel IV.15 Skenario <i>Use Case</i> Menghapus Data Kategori .....	45
Tabel IV.16 Skenario <i>Use Case</i> Menambah Data Ruang.....	45
Tabel IV.17 Skenario <i>Use Case</i> Mengubah Data Ruang.....	46
Tabel IV.18 Skenario <i>Use Case</i> Menghapus Data Ruang .....	47
Tabel IV.19 Skenario <i>Use Case</i> Menambah Data Laboran .....	48
Tabel IV.20 Skenario <i>Use Case</i> Mengubah Data Laboran .....	48
Tabel IV.21 Skenario <i>Use Case</i> Menghapus Data Laboran .....	49
Tabel IV.22 Skenario <i>Use Case</i> Menambah Data Kompetensi .....	50
Tabel IV.23 Skenario <i>Use Case</i> Mengubah Data Kompetensi .....	51
Tabel IV.24 Skenario <i>Use Case</i> Menghapus Data Kompetensi.....	51
Tabel IV.25 Skenario <i>Use Case</i> Menambah Data Pengadaan .....	52
Tabel IV.26 Skenario <i>Use Case</i> Mengubah Data Pengadaan .....	53
Tabel IV.27 Skenario <i>Use Case</i> Menghapus Data Pengadaan .....	54

Tabel IV.28	Skenario <i>Use Case</i> Mencetak/Export Data Pengadaan .....	54
Tabel IV.29	Skenario <i>Use Case</i> Menambah Data Perbaikan .....	54
Tabel IV.30	Skenario <i>Use Case</i> Mengubah Data Perbaikan .....	55
Tabel IV.31	Skenario <i>Use Case</i> Menghapus Data Perbaikan .....	56
Tabel IV.32	Skenario <i>Use Case</i> Mencetak/Export Data Perbaikan .....	57
Tabel IV.33	Skenario <i>Use Case</i> Menambah Data Pengembalian.....	57
Tabel IV.34	Skenario <i>Use Case</i> Mengubah Data Pengembalian.....	58
Tabel IV.35	Skenario <i>Use Case</i> Menghapus Data Pengembalian.....	59
Tabel IV.36	Skenario <i>Use Case</i> Mencetak/Export Data Pengembalian..	59
Tabel IV.37	Deskripsi Mockup Antarmuka Home.....	72
Tabel IV.38	Deskripsi Mockup Antarmuka Tentang FTI UNIBBA .....	73
Tabel IV.39	Deskripsi Mockup Antarmuka Login .....	74
Tabel IV.40	Deskripsi Mockup Antarmuka Dashboard.....	75
Tabel IV.41	Deskripsi Mockup Antarmuka Data Peralatan .....	76
Tabel IV.42	Deskripsi Mockup Antarmuka Data Kategori .....	78
Tabel IV.43	Deskripsi Mockup Antarmuka Data Perbaikan .....	80
Tabel IV.44	Perancangan Data Peralatan .....	81
Tabel IV.45	Perancangan Data Merek.....	82
Tabel IV.46	Perancangan Data Ruang .....	82
Tabel IV.47	Perancangan Data Kategori .....	82
Tabel IV.48	Perancangan Data Perbaikan.....	82
Tabel IV.49	Perancangan Data Pengembalian .....	82
Tabel IV.50	Perancangan Data Pengadaan .....	83
Tabel IV.51	Perancangan Data Laboran.....	83
Tabel IV.52	Perancangan Data Kompetensi.....	83
Tabel IV.53	Perancangan Data Login .....	84

# **BAB I**

## **PENDAHULUAN**

### **I.1 Latar Belakang**

Laboratorium komputer merupakan salah satu unit penting pelaksana teknis yang mendukung fungsi di perguruan tinggi sebagai bidang pelayanan dan pengembangan teknologi informasi baik yang berhubungan dengan sistem perangkat lunak, perangkat keras maupun jaringan. Laboratorium komputer berperan penting dalam meningkatkan kualitas pendidikan serta sebagai penunjang kegiatan praktikum mahasiswa dalam menerapkan teori dan konsep yang didapatkan diperkuliahan. Dalam upaya memaksimalkan fungsi dan peran laboratorium, maka sangat perlu untuk memperhatikan kelengkapan sarana prasarana, pemeliharaan, pengawasan serta evaluasi terhadap pemakaian laboratorium.

Pengelolaan data aset merupakan hal yang penting untuk selalu dilakukan pada proses pengontrolan data aset di laboratorium komputer Fakultas Teknologi Informasi UNIBBA agar informasi tentang aset selalu *update* sehingga memberikan informasi yang jelas dan memudahkan dalam melakukan perbaikan dan pengadaan. Oleh karena itulah pengelolaan data aset harus ada karena pada setiap semesternya penggunaan laboratorium komputer akan terjadi penggantian alat atau perangkat keras yang mengalami kerusakan. Pengadaan alat baru tentu juga akan terjadi pada periode pengajaran.

Pada kenyataannya pengelolaan data aset di laboratorium komputer FTI UNIBBA yang selama ini dilakukan masih belum *update*. Sehingga terjadi kesalahan dalam pendataan atau bahkan lupa mendata adanya penggantian barang maupun adanya pengadaan barang baru tidak dapat di hindari, yang dimana akan mengakibatkan pelacakan terhadap barang yang digunakan dan tempat dimana piranti tersebut dipasang atau diinstall akan menjadi sulit atau lambat dilakukan.

Dari jurnal yang ditulis oleh Muhammad Iqbal Awaluddin, Rita Wahyuni Arifin, dan Didik Setiyadi, berdasarkan hasil penelitiannya menyatakan bahwa dengan mengimplementasikan sebuah sistem untuk mengelola data aset laboratorium komputer, dapat mengelola data aset laboratorium komputer secara cepat dan akurat (Awaluddin dkk., 2020).

Oleh karena itulah, berdasarkan permasalahan tersebut dibutuhkan sebuah sistem maupun aplikasi untuk mengatasi permasalahan mengenai pengelolaan aset laboratorium komputer agar lebih terkomputerisasi serta cepat dan akurat. Maka dari itu, penulis menentukan untuk membuat aplikasi aset laboratorium komputer dengan codeigniter versi 3.1.4 dan bootstrap versi 3.3.7 untuk memudahkan pengajuan proposal aset laboratorium, diharapkan aplikasi yang dibuat tersebut dapat digunakan untuk keperluan pendataan aset laboratorium komputer FTI UNIBBA untuk menghindari resiko-resiko yang dapat terjadi, seperti efisiensi waktu saat melakukan pengecekan aset, informasi tidak relevan, kemungkinan adanya duplikasi data, serta ketidaksesuaian data dengan bukti fisik aset di laboratorium.

## **I.2 Lingkup**

Lingkup materi kerja praktek yang dilaksanakan di Laboratorium Komputer FTI UNIBBA adalah pembuatan aplikasi aset laboratorium komputer dengan codeigniter versi 3.1.4 dan bootstrap versi 3.3.7 untuk memudahkan pengajuan proposal aset laboratorium serta untuk menangani semua data aset di laboratorium komputer yang menyangkut hal berikut:

1. Data peralatan(kode alat, nama, kode merek, kode kategori, deskripsi, kode ruang, status),
2. Data merek(kode merek, merek),
3. Data kategori(kode kategori, kategori),
4. Data ruang(kode ruang, ruang, lokasi),
5. Data laboran(id laboran, nama, img, jenis kelamin, alamat, password, kode kompetensi, hp, ket),

6. Data kompetensi(kode kompetensi,kompetensi),
7. Data login(username, password, stts),
8. Data pengadaan(kode baru, tanggal, kode alat, qty, id laboran),
9. Data perbaikan(kode perbaikan, tanggal, kode alat, ket, id laboran)
10. Data pengembalian(kode kembali, kode perbaikan, tanggal, kode alat, id laboran)

Data di atas merupakan daftar database yang akan dikelola di aplikasi aset laboratorium komputer FTI UNIBBA.

### **I.3 Tujuan**

Kerja praktek yang dilakukan di Ruang Laboratorium Komputer FTI UNIBBA dari tanggal 16 Oktober 2021 sampai dengan 16 Desember 2021 bertujuan untuk mempermudah dalam pendataan dan akses informasi aset laboratorium komputer serta memudahkan melakukan usulan pengadaan dan perbaikan aset laboratorium komputer ke pihak Yayasan dan Rektorat Universitas Bale Bandung.

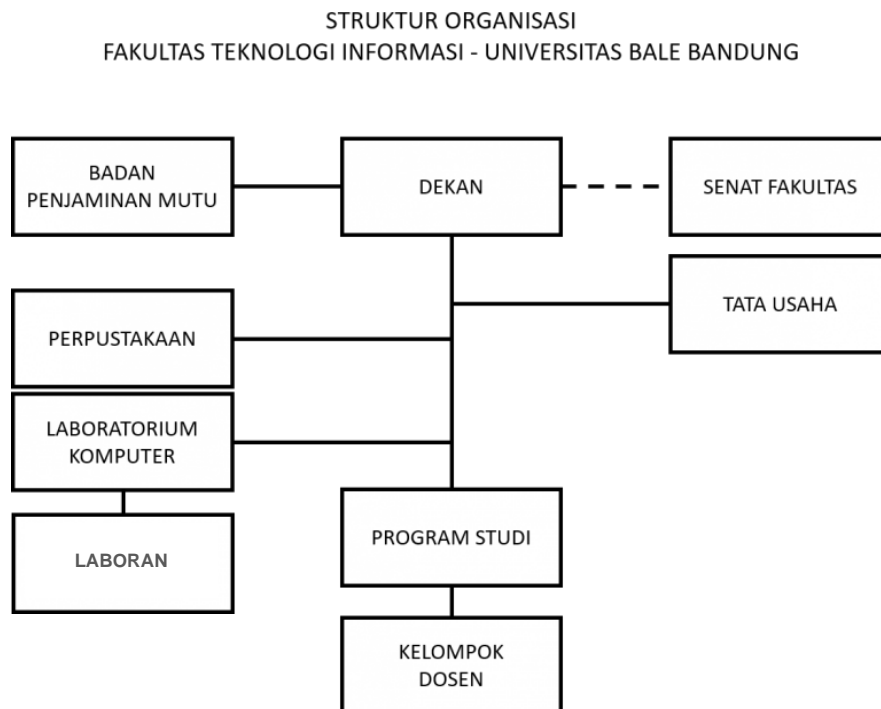
## BAB II

### LINGKUNGAN KERJA PRAKTEK

#### II.1 Struktur Organisasi

Pada tahun 2008 berdasarkan SK DIKTI Nomor 80/D/O/2008 tanggal 22 Mei 2008 Fakultas Teknologi Informasi (FTI) bergabung dengan Universitas Bale Bandung (UNIBBA) dibawah binaan Yayasan Pendidikan Bale Bandung (YPBB) yang didirikan oleh Bapak R.H. Lily Sumantri (beliau pernah menjadi Bupati Kabupaten Bandung pada Tahun 1980). Fakultas Teknologi Informasi memiliki dua program studi yaitu Teknik Informatika dan Sistem Informasi.

Pada tahun 2008 awalnya laboratorium komputer hanya memiliki satu ruangan saja dan juga pada saat itu laboratorium dikelola langsung oleh Rektorat Universitas Bale Bandung. Pada tahun 2012 ruangan laboratorium komputer bertambah satu sehingga memiliki dua ruangan laboratorium komputer dan pada saat itu juga pengelolaan laboratorium komputer diserahkan ke pihak Fakultas Teknologi Informasi. Berikut adalah struktur organisasi Fakultas Teknologi Informasi :



Gambar II.1 Struktur Organisasi FTI UNIBBA

Fakultas Teknologi Informasi merupakan salah satu fakultas yang ada di Universitas Bale Bandung berfokus pada bidang Teknologi Informasi dengan visi dan misi sebagai berikut:

a. Visi

Menjadi Fakultas Teknologi Informasi (FTI) yang unggul dibidang Teknik Informatika dan Sistem Informasi, yang mampu mencetak lulusan dengan keahlian profesional bidang komputer, berperan dalam perkembangan ilmu pengetahuan dan teknologi informasi dan menghasilkan solusi aplikatif bagi dunia usaha/industri yang mampu bersaing dalam tataran global pada tahun 2028.

b. Misi

1. Menyelenggarakan pendidikan bertaraf nasional untuk menghasilkan lulusan yang berbudi pekerti luhur dan berdaya saing global.
2. Menyelenggarakan penelitian yang inovatif dan bermanfaat untuk pengembangan ilmu pengetahuan dan teknologi informasi yang berdaya saing global.
3. Menerapkan ilmu pengetahuan dan teknologi informasi yang inovatif kepada masyarakat, pemerintah, dan sektor swasta sebagai kontribusi pada pembangunan yang berdaya saing global.
4. Mejalin jaringan kerjasama pendidikan, pelatihan, penelitian dan pengabdian masyarakat dengan berbagai pihak untuk meningkatkan kualitas kinerja dan citra.
5. Mewujudkan pelayanan terbaik kepada stakeholders melalui dukungan organisasi dan manajemen fakultas yang efisien dan efektif dengan menerapkan penjaminan mutu dan evaluasi diri secara berkelanjutan.

Teknik Informatika merupakan salah satu program studi di Fakultas Teknologi Informasi yang memiliki visi dan misi sebagai berikut :

a. Visi

Menjadi Program Studi Teknik Informatika yang mampu menghasilkan



lulusan yang unggul dan mandiri bidang rekayasa perangkat lunak di Jawa Barat pada tahun 2028.

b. Misi

1. Mengembangkan sistem dan proses belajar bidang rekayasa perangkat lunak serta mampu mengembangkan pendidikan untuk menghasilkan lulusan yang unggul dan mandiri.
2. Mengembangkan penelitian bidang rekayasa perangkat lunak yang inovatif dan bermanfaat untuk perkembangan teknologi rekayasa perangkat lunak dan masyarakat.
3. Mengembangkan pengabdian kepada masyarakat dalam bentuk pengembangan teori, sistem dan aplikasi serta pemecahan berbagai permasalahan bidang rekayasa perangkat lunak yang sesuai dengan kebutuhan masyarakat.
4. Meningkatkan kerjasama yang berkelanjutan dengan lembaga institusi lain, pemerintah industri, asosiasi bidang keilmuan, dan masyarakat dalam rangka meningkatkan terselenggaranya kualitas pendidikan, penelitian dan pengabdian kepada masyarakat.

Sistem Informasi merupakan salah satu program studi di Fakultas Teknologi Informasi yang memiliki visi dan misi sebagai berikut :

a. Visi

Menjadi Program Studi Sistem Informasi yang mampu menghasilkan lulusan yang unggul di bidang rekayasa Sistem Informasi Manufaktur di Jawa Barat pada tahun 2028.

b. Misi

1. Menyelenggarakan pendidikan Rekayasa sistem informasi di wilayah Kabupaten Bandung.
2. Menyelenggarakan penelitian dalam bidang Rekayasa sistem informasi yang disesuaikan dengan perkembangan sistem informasi di wilayah Kabupaten Bandung.
3. Menyelenggarakan pengabdian masyarakat yang sesuai dengan keterampilan bidang sistem informasi.

4. Melaksanakan administrasi akademik dan nonakademik yang bermutu.
5. Menumbuhkembangkan kegiatan ekstrakurikuler bagi mahasiswa yang mendorong pengembangan potensi mahasiswa.

Laboratorium komputer FTI UNIBBA memiliki visi dan misi yang sejalan dengan visi misi fakultas dan program studi yang ada di FTI. Laboratorium komputer FTI UNIBBA berperan untuk membantu fakultas serta program studi FTI untuk mencapai visi dan misinya.

## **II.2 Lingkup Pekerjaan**

Tempat melaksanakan kerja praktek adalah di bagian Laboratorium Komputer FTI UNIBBA. Nantinya akan membangun aplikasi aset laboratorium komputer dengan codeigniter versi 3.1.4 dan bootstrap versi 3.3.7 untuk memudahkan pengajuan proposal aset laboratorium yang didalamnya terdapat data peralatan, merek, kategori, ruang, laboran, kompetensi, login, pengadaan, perbaikan dan data pengembalian. Proses pembuatan aplikasi tersebut dimulai dari analisis data peralatan.

## **II.3 Deskripsi Pekerjaan**

Deskripsi pekerjaan yang dilakukan selama kerja praktek adalah pembuatan Aplikasi Aset Laboratorium Komputer yang menangani hal berikut:

1. Pencatatan, modifikasi, dan pelaporan data peralatan,
2. Pencatatan, modifikasi, dan pelaporan data merek,
3. Pencatatan, modifikasi, dan pelaporan data kategori,
4. Pencatatan, modifikasi, dan pelaporan data ruang,
5. Pencatatan, modifikasi, dan pelaporan data laboran,
6. Pencatatan, modifikasi, dan pelaporan data kompetensi,
7. Pencatatan, modifikasi, dan pelaporan data login,
8. Pencatatan, modifikasi, dan pelaporan data pengadaan,
9. Pencatatan, modifikasi, dan pelaporan data perbaikan,
10. Pencatatan, modifikasi, dan pelaporan data pengembalian.

## II.4 Jadwal Kerja

Kerja praktek dilaksanakan dari tanggal 16 Oktober 2021 sampai dengan 16 Desember 2021 selama 2 bulan. Waktu kerja praktek adalah dari hari Senin sampai dengan Sabtu, pukul 09.00 sampai dengan pukul 14.00 WIB. Secara umum, kegiatan yang dilakukan selama kerja praktek adalah sebagai berikut:

**Tabel II.1 Waktu kegiatan kerja praktek**

Uraian Kegiatan	Bulan							
	Okt		Nov				Des	
	III	IV	I	II	III	IV	I	II
Pengenalan lingkungan kerja								
Instalasi software yang diperlukan								
Perancangan basis data								
Perancangan prototype								
Pembuatan diagram UML								
Pembuatan aplikasi aset laboratorium								
Penyusunan laporan kerja praktek								

## **BAB III**

### **TEORI PENUNJANG KERJA PRAKTEK**

#### **III.1 Teori Penunjang**

Selama pelaksanaan kerja praktek di Laboratorium Komputer FTI UNIBBA, peserta kerja praktek menggunakan pengetahuan yang diperoleh selama masa perkuliahan sebagai landasan teori pembuatan aplikasi aset laboratorium komputer. Pengetahuan dan teori yang digunakan antara lain:

1. Konsep Algoritma Pemrograman

Teori tentang Algoritma Pemrograman diperoleh pada mata kuliah TIF301 Algoritma Pemrograman 1 dan FTI302 Algoritma dan Pemrograman 2.

2. Konsep Interaksi Manusia dan Komputer

Teori tentang konsep Interaksi Manusia dan Komputer diperoleh pada mata kuliah FTI307 Interaksi Manusia dan Komputer.

3. Konsep Pemrograman Berorientasi Objek

Teori tentang konsep Pemrograman Berorientasi Objek diperoleh pada mata kuliah TIF308 Pemrograman Berorientasi Objek.

4. Konsep Pemrograman Internet

Teori dan konsep Pemrograman Internet yang diperoleh pada mata kuliah FTI319 Pemrograman Internet.

5. Konsep Basis Data

Teori tentang konsep Basis Data diperoleh di mata kuliah TIF310 yaitu Basis Data dan di mata kuliah TIF311 yaitu Sistem Basis Data.

6. Konsep Rekayasa Perangkat Lunak

Teori dan konsep Rekayasa Perangkat Lunak yang diperoleh pada mata kuliah TIF316 Rekayasa Perangkat Lunak.

7. Konsep Proyek Perangkat Lunak

Teori tentang Manajemen Proyek Perangkat Lunak diperoleh di mata kuliah TIF318 Manajemen Proyek Perangkat Lunak dan TIF315 Proyek Perangkat Lunak.

Selain dari teori penunjang di beberapa mata kuliah diatas, ada juga beberapa materi lain yang dijadikan landasan teori pembuatan aplikasi aset laboratorium komputer, diantaranya:

#### 1. Konsep MVC (*Model, View, Controller*)

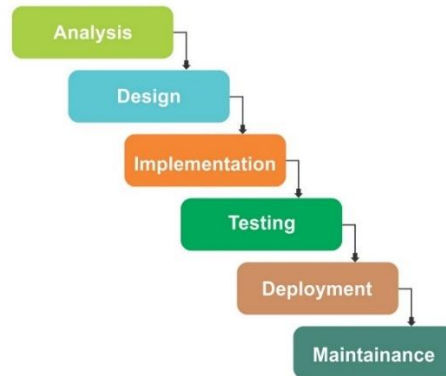
*Model, View, Controller* merupakan suatu konsep dalam pembangunan aplikasi web, berawal pada bahasa pemrograman *SmallTalk*. MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, antarmuka pengguna, dan bagian kontrol aplikasi. Terdapat 3 jenis komponen yang membangun suatu pola MVC dalam suatu aplikasi yaitu:

- *Model*, biasanya berhubungan langsung dengan pangkalan data untuk memanipulasi data (*insert, update, delete, search*), menangani validasi dari bagian *controller*, tetapi tidak dapat berhubungan langsung dengan bagian *view*.
- *View*, merupakan bagian yang menangani logika presentasi. Pada suatu aplikasi web bagian ini biasanya berupa berkas templat HTML, yang diatur oleh *controller*. *View* berfungsi untuk menerima dan merepresentasikan data kepada pengguna. Bagian ini tidak memiliki akses langsung terhadap bagian *model*.
- *Controller*, merupakan bagian yang mengatur hubungan antara bagian *model* dan bagian *view*. *Controller* berfungsi untuk menerima permintaan dan data dari pengguna kemudian menentukan apa yang akan diproses oleh aplikasi.

#### 2. Konsep SDLC (*System Development Life Cycle*)

SDLC atau Siklus Hidup Pengembangan Sistem adalah proses pembuatan dan perubahan sistem serta model dan metodologi yang digunakan untuk mengembangkan sistem-sistem tersebut. Konsep ini umumnya merujuk pada sistem komputer atau informasi. Pengembangan sistem metode SDLC atau sering disebut sebagai pendekatan air terjun (*waterfall*). Metode *waterfall* pertama kali diperkenalkan oleh windows W. Royce pada tahun 1970.

*Waterfall* merupakan model klasik yang sederhana dengan aliran sistem yang linier *output* dari setiap tahap kemudian *input* bagi tahap berikutnya (Kristanto, 2004).



Gambar III.1 Model Waterfall

a. Analisis Sistem

Analisis sistem adalah penguraian dari suatu sistem yang utuh kedalam kegiatan-kegiatan komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan, yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

b. Desain Sistem (*Design*)

Desain Sistem adalah persiapan rancang bangun implementasi yang menggambarkan bagaimana suatu sistem dibentuk yang berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi. Menyangkut di dalamnya konfigurasi komponen perangkat lunak dan perangkat keras dari suatu sistem.

c. Implementasi (*Implementation*)

Menerjemahkan hasil proses perancangan menjadi sebuah bentuk program komputer yang dimengerti oleh mesin komputer.

d. Uji Coba Program (*Testing*)

Uji coba software merupakan elemen yang kritis dari SQA (*Software Quality Assurance*) dan mempresentasikan tinjauan

ulang yang menyeluruh terhadap spesifikasi, desain dan pengkodean. Uji coba mempresentasikan ketidak normalan yang terjadi pada pengembangan software dari tahap awal hingga implementasi.

e. Penyebaran (*Deployment*)

Tahap penyebaran adalah tahap dimana semua elemen dan aktivitas sistem disatukan yang dimana program dapat di operasikan dilingkungannya dan dinyatakan fungsional serta bisa digunakan oleh pengguna lainnya. Tahapannya sebagai berikut:

i. Menyiapkan Fasilitas Fisik

Fasilitas-fasilitas fisik yang disiapkan antara lain: komputer dan peripheralnya, termasuk keamanan fisik untuk menjaga berlangsungnya peralatan dalam jangka waktu yang lama.

ii. Menyiapkan Pemakai

Pemakai disiapkan dengan terlebih dahulu yaitu dengan memberikan pelatihan secara prosedural maupun tutorial mengenai sistem informasi sesuai fungsi tugasnya. Tujuannya adalah agar para pemakai mengerti dan menguasai operasi sistem dan cara kerja sistem serta apa saja yang diperoleh dari sistem.

iii. Melakukan Simulasi

Kegiatan simulasi berupa pengujian sistem secara nyata yang melibatkan personil yang sesungguhnya.

f. Pemeliharaan Sistem (*Maintenance*)

i. Untuk membenarkan kesalahan atau kelemahan sistem yang tidak terdeteksi pada saat pengujian,

ii. Untuk membuat sistem up to date,

iii. Untuk meningkatkan kemampuan sistem.

3. *Unified Modeling Language* (UML)




*Unified Modeling Language* (UML) adalah himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (OOP)

serta aplikasinya. UML adalah metodologi untuk mengembangkan sistem OOP dan sekelompok perangkat *tool* untuk mendukung pengembangan sistem tersebut. UML mulai diperkenalkan oleh *Object Management Group*, sebuah organisasi yang telah mengembangkan model, teknologi, dan standar OOP sejak tahun 1980-an. UML merupakan dasar bagi perangkat (*tool*) desain berorientasi objek dari IBM.




a. *Use Case Diagram*

*Use case diagram* adalah diagram *use case* yang digunakan untuk menggambarkan secara ringkas siapa yang menggunakan sistem dan apa saja yang bisa dilakukannya. Diagram *use case* tidak menjelaskan secara detail tentang penggunaan *use case*, namun hanya memberi gambaran singkat hubungan antara *use case*, aktor, dan sistem. Melalui diagram *use case* dapat diketahui fungsi-fungsi apa saja yang ada pada sistem (Rosa-Salahudin, 2011: 130).

Tabel III.1 Simbol *Use case diagram*

No	Simbol	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem, menghasilkan suatu hasil terukur bagi suatu <i>Actor</i> .
3		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).



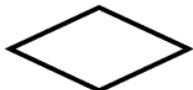




4		<i>Association</i>	Penghubung antara objek satu dengan objek lainnya.
5		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit
6		<i>Extend</i>	Menspesifikasikan bahwa <i>use case target</i> memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang Diberikan

b. *Activity Diagram*

*Activity Diagram* adalah diagram yang menggambarkan *workflow* (aliran kerja) aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan Salahudin 2013).

Tabel III.2 Simbol *Activity Diagram*

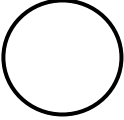

No	Simbol	Nama	Keterangan
1		Status Awal	Aktivitas yang dilakukan sistem, aktivitas diawali dengan kata kerja
2		Aktivitas	Deskripsi dari urutan aksi - aksi yang ditampilkan sistem, menghasilkan suatu hasil yang terukur bagi suatu <i>Actor</i>
3		Percabangan	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu

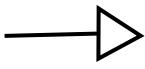



4		Penggabungan	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
5		Status Akhir	Status akhir yang dilakukan oleh sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

c. *Class Diagram*

*Class diagram* merupakan alur jalannya database yang saling terhubung pada sebuah sistem. *Class diagram* menunjukkan visual dari struktur sistem program pada jenis-jenis yang dibentuk. Menurut Satzinger (2011:28) Diagram kelas atau *class diagram* menjelaskan struktur sistem dari segi pendefinisian *class-class* yang akan dibuat untuk membangun sebuah sistem.

Tabel III.3 Simbol *Class Diagram*

No	Simbol	Nama	Keterangan
1	Nama_kelas	Kelas	Kelas pada struktur sistem.
	+atribut		
	+operasi()		
2		Antarmuka / <i>interface</i>	Sama dengan konsep interface dalam pemrograman berorientasi objek
3		Asosiasi berarah / <i>directed association</i>	Relasi antar kelas dengan makna kelas yang digunakan oleh kelas lain, asosiasi biasanya juga disertai dengan multiplicity.

4		Generalisasi	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
5		Kebergantungan / dependency	Relasi antar kelas dengan makna kebergantungan antar kelas
6		Agresi / aggretrion	Relasi antar kelas dengan makna semua-bagian (whole-part)
7		Asosiasi / <i>association</i>	Relasi antar class dengan arti umum

### III.2 Peralatan Pembuatan Aplikasi

Kakas atau *tools* yang digunakan dalam pembangunan aplikasi aset laboratorium komputer antara lain:

#### 1. XAMPP versi 3.2.2

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP Server, MySQL *database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU *General Public License* dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis.

### Perubahan (*changelog*) pada XAMPP versi 3.2.2

- Dikompilasi dengan kompiler Delphi yang diperbarui
- Memperbarui Pustaka (JCL 2.7.0.5449)
- Memperbarui pustaka terjemahan
- Mengatasi perihal semua petunjuk peringatan saat kompilasi
- Menambahkan terjemahan bahasa Polandia
- Menambahkan terjemahan bahasa Prancis
- Terjemahan bahasa Jerman yang diperbarui
- Mengubah font default dari semua komponen antarmuka
- Ukuran komponen antarmuka yang disesuaikan untuk mengakomodasi terjemahan baru
- Menambahkan dialog khusus (Windows Vista dan lebih tinggi)
- Menambahkan parameter file konfigurasi untuk menentukan URL Admin Apache khusus
- Menambahkan parameter file konfigurasi untuk menentukan URL Admin MySQL khusus
- Menambahkan parameter file konfigurasi untuk menentukan URL Admin Tomcat khusus
- Sedikit *tweak* kode



Gambar III.2 XAMPP

#### a. Fungsi XAMPP

Fungsi XAMPP adalah sebagai sebuah server lokal yang berdiri sendiri (disebut juga *Localhost*) yang terdiri dari program Apache HTTP Server, MySQL *database*, PHP dan Perl. Dalam prakteknya XAMPP berfungsi sebagai "*demo*" dari tampilan halaman website.

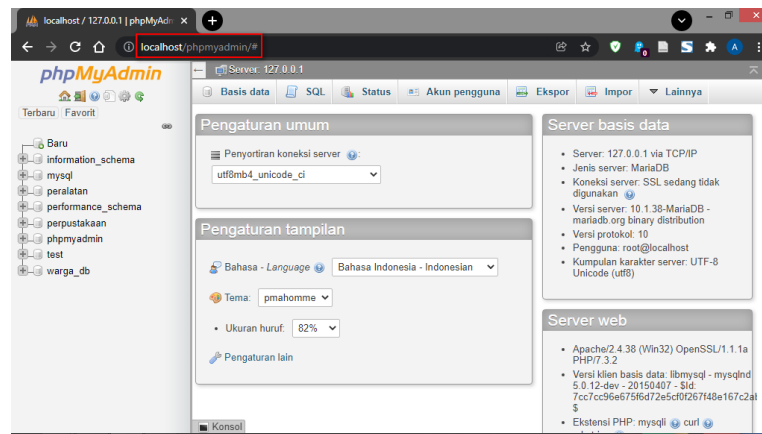
#### b. Fitur Penting Pada XAMPP

- **htdocs** adalah folder tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas PHP, HTML dan *script* lain

Name	Date modified	Type
anonymous	29/11/2021 18:56	File folder
apache	29/11/2021 18:56	File folder
cgi-bin	29/11/2021 19:02	File folder
contrib	29/11/2021 18:56	File folder
FileZillaFTP	29/11/2021 19:02	File folder
htdocs	26/01/2022 18:56	File folder
img	29/11/2021 18:56	File folder

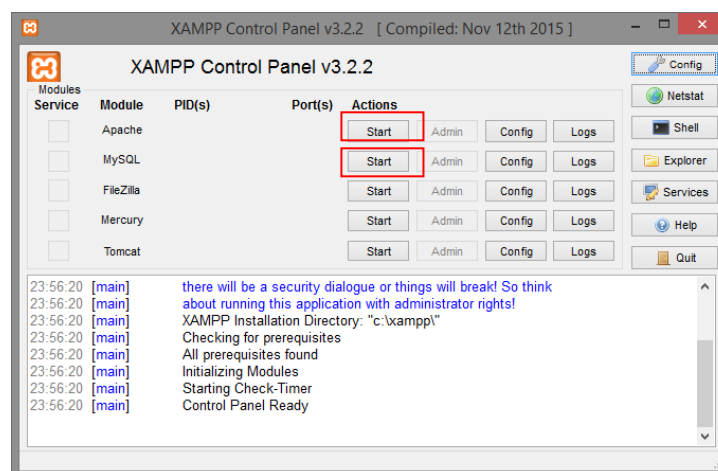
Gambar III.3 htdocs

- **phpMyAdmin** merupakan bagian untuk mengelola basis data MySQL yang ada dikomputer. Untuk mengaksesnya, buka browser lalu ketikkan alamat <http://localhost/phpMyAdmin/>, maka akan muncul halaman phpMyAdmin.



Gambar III.4 phpMyAdmin

- **Control Panel** berfungsi mengelola layanan (*service*) XAMPP. Seperti menghentikan (*stop*) layanan atau memulai (*start*), mengelola modul apa saja yang akan digunakan, mengatur konfigurasi XAMPP, mengetahui informasi port dan lainnya.



Gambar III.5 Control Panel

### c. Tipe Data di XAMPP (MySQL)

Tipe data merupakan jenis data yang digunakan untuk mendefinisikan *field* atau kolom yang digunakan di MySQL. Selain itu, ada pula kolom indeks yang memiliki nilai unik yaitu *primary key* (PK) dan *foreign key* (FK). Berikut adalah tipe-tipe data yang sering digunakan di MySQL:

#### 1) Tipe String (Text)

Tipe data string merupakan tipe data yang digunakan untuk menyimpan data string (text). Berikut tipe *field* (kolom) di MySQL yang termasuk ke dalam kelompok tipe data string:

##### a) Char

Penggunaan : Data string ukuran tetap

Jangkauan : 0 s/d 255 karakter

##### b) VarChar

Penggunaan : Data string ukuran dinamis

Jangkauan : 0 s/d 255 karakter (versi 4.1), 0 s/d 65.535 (versi 5.0.3)

##### c) Text

Penggunaan : Data text

Jangkauan : 0 s/d 65.535 (216 – 1) karakter

#### 2) Tipe Numerik

Tipe data numerik digunakan untuk menyimpan data numerik (angka). Ciri utama data numerik adalah suatu data yang memungkinkan untuk dikenai operasi aritmatika seperti pertambahan, pengurangan, perkalian dan pembagian. Berikut tipe *field* (kolom) di MySQL yang termasuk dalam kelompok tipe numerik:

##### a) Int

Penggunaan : Data bilangan bulat positif dan negatif

Jangkauan : -2.147.483.648 s/d 2.147.483.647

Ukuran : byte (32 bit)

## b) Float

Penggunaan : Data bilangan pecahan positif dan negative yang presisi tunggal

Jangkauan : 3.402823466E+38 s/d -1.175494351E-38, 0, dan 1.175494351E-38 s/d 3.402823466E+38

Ukuran : 4 byte (32 bit)

## c) Double/real

Penggunaan : Data bilangan pecahan positif dan negatif presisi ganda

Jangkauan : -1.79...E+308 s/d -2.22...E-308, 0, dan 2.22...E-308 s/d 1.79...E+308

Ukuran : 8 byte (64 bit)

## d) Decimal/numeric

Penggunaan : Data bilangan pecahan positif dan negatif

Jangkauan : -1.79...E+308 s/d -2.22...E-308, 0, dan 2.22...E-308 s/d 1.79...E+308

Ukuran : 8 byte (64 bit)

3) Tipe *Date and Time*

Untuk menyimpan data tanggal dan waktu. Berikut tipe *field* (kolom) di MySQL yang termasuk dalam kelompok ini:

## a) Date

Penggunaan : Data tanggal

Jangkauan : 1000-01-01 s/d 9999-12-31 (YYYY-MM-DD)

Ukuran : 3 byte

## b) Time

Penggunaan : Data waktu

Jangkauan : -838:59:59 s/d +838:59:59 (HH:MM:SS)

Ukuran : 3 byte

## c) Datetime

Penggunaan : Data tanggal dan waktu

Jangkauan : '1000-01-01 00:00:00' s/d '9999-12-31  
23:59:59'

Ukuran : 8 byte

## d) Year

Penggunaan : Data tahun dari tanggal

Jangkauan : 1900 s/d 2155

Ukuran : 1 byte

4) Tipe Biner (BLOB (*Binary Large Object*))

Tipe ini biasanya digunakan untuk menyimpan kode-kode biner dari suatu file atau object. Berikut tipe *field* (kolom) di MySQL yang termasuk dalam kelompok tipe blob:

## a) Bit (sejak versi 5.0.3)

Penggunaan : Data biner

Jangkauan : 64 digit biner

## b) Tinyblob

Penggunaan : Data biner.

Jangkauan : 255 byte

## c) Blob

Penggunaan : Data biner.

Jangkauan : 2<sup>16</sup> – 1 byte

## d) Mediumblob

Penggunaan : Data biner.

Jangkauan : 2<sup>24</sup> – 1 byte

## e) Longblob

Penggunaan : Data biner.

Jangkauan : 2<sup>32</sup> – 1 byte

## 5) Tipe Data Lain

Di MySQL juga menyediakan tipe data selain ke empat tipe data yang telah disebutkan diatas. Tipe data di MySQL



mungkin akan terus bertambah seiring dengan perkembangan versi MySQL dan XAMPP nya. Berikut ini beberapa tipe data tambahan di MySQL:

a) Enum

Penggunaan : Enumerasi (kumpulan data)

Jangkauan : Sampai dengan 65535 string

b) Set

Penggunaan : Combination (himpunan data)

Jangkauan : Sampai dengan 255 string anggota

## 2. Visual Studio Code

Visual Studio Code adalah Software yang sangat ringan, namun kuat yang berjalan dari desktop. Muncul dengan *built-in* dukungan untuk JavaScript, naskah dan Node.js dan memiliki array beragam ekstensi yang tersedia untuk bahasa lain, termasuk C ++, C# , Python, dan PHP. Visual Studio Code menggunakan open source NET untuk memberikan dukungan ASP. Antarmuka yang mudah untuk bekerja karena didasarkan pada gaya explorer umum, dengan panel di sebelah kiri yang menunjukkan semua file dan folder dan di sebelah kanan yang menunjukkan isi dari file yang telah dibuka. Visual studio code memiliki fungsi yang baik, dengan intellisense dan *autocomplete* bekerja dengan baik untuk JSON, CSS, HTML, Node.js dan lainnya.

Visual Studio Code merupakan kode editor lintas platform yang ringan, yang dapat digunakan oleh siapa saja untuk membangun aplikasi Web.



Gambar III.6 Logo Visual Studio Code

### a. Kelebihan Visual Studio Code

- Text editor gratis

- **Sudah terinstall plugin EMMET** yaitu plugin untuk menghemat waktu dan mempercepat dalam pembuatan sebuah web.
  - **Mudah untuk mengelola *Extensions*** yang dimana pengelolaan extensions sangat mudah dipahami untuk pengguna baru visual studio code. hanya perlu ke menu *ekstensions* atau tekan CTRL + SHIFT + X dan pilih *ekstensions* mana yang sudah terinstall atau mau di install atau bisa juga dengan melakukan pencarian menggunakan kolom pencarian.
  - ***Extensions* yang banyak** : Banyak extensions visual studio code untuk memudahkan pekerjaan contohnya *auto rename tag* yang sangat berguna bagi *web developer*.
  - **Kostumisasi tampilan** : Di visual studio code juga bisa merubah tampilan seperti icon, font, warna atau bisa juga menginstall berbagai tema seperti *material theme* atau tema lainnya.
  - **Terintegrasi dengan GIT** : sudah terintegrasi dengan git sehingga lebih mudah dalam menyelesaikan *conflict* atau mengetahui mana baris yang berubah atau ditambahkan sehingga akan lebih memudahkan dalam kerja tim.
  - ***Snippet*** : Di visual studio code bisa membuat *snippet* sendiri atau bisa menginstall *snippet* dari menu *ekstensions*.
  - **Dukungan bahasa** : Visual studio code sudah mendukung banyak bahasa seperti c, c++, php, javascript dan masih banyak yang lainnya.
  - ***Autocomplete*** : Di visual studio code sudah disediakan fitur *autocomplete* tentunya untuk memudahkan pekerjaan. Untuk memunculkan *autocomplete* dengan menekan CTRL + SPACE.
- b. Kekurangan Visual Studio Code
- **Performa** : Dibandingkan dengan text editor lain seperti sublime text yang masih lancar di *processor* 2 core dengan 2GB RAM sedangkan dengan spek yang sama visual studio

code terkadang *crash* atau *lag*. Jadi untuk performa yang stabil, visual studio code membutuhkan *processor* 4 core dan 4GB RAM.

### 3. Google Chrome

Google Chrome adalah peramban web lintas *platform* yang dikembangkan oleh Google. Peramban ini pertama kali dirilis pada tahun 2008 untuk Microsoft Windows, kemudian di *porting* ke Linux, macOS, iOS, dan Android yang menjadikannya sebagai peramban bawaan dalam sistem operasi. Peramban ini juga merupakan komponen utama Chrome OS, yang berfungsi sebagai platform untuk aplikasi web. Sebagian besar kode sumber Chrome berasal dari proyek perangkat lunak gratis dan sumber terbuka Google, Chromium, tetapi Chrome di lisensikan sebagai perangkat gratis berpemilik. WebKit adalah mesin rendering asli, tetapi Google akhirnya *fork*nya untuk membuat mesin Blink; semua varian Chrome kecuali iOS sekarang menggunakan Blink.



Gambar III.7 Logo/ Icon Google Chrome

Menurut Jubilee (2009:4) “Google Chrome sebagai browser baru mempunyai fasilitas yang lumayan bagus sehingga mampu menarik perhatian pecinta dunia maya dari seluruh penjuru dunia “.

Google Chrome merupakan mesin pencarian yang mampu melakukan penelusuran dalam waktu kurang dari beberapa detik serta memberikan fitur pendukung bagi pengguna seperti mediator layanan *browser*, *file manager*, *downloader* dan lain-lain. Sebagai salah satu layanan *software* yang memungkinkan pengguna *website* menelusuri informasi, media video dan audio, serta data teknis Google Chrome tersedia dan sangat mendukung untuk semua Operasi Sistem Dektop

hingga pengguna smartphone seperti Android dan Apple agar *browser* menjadi terkendali untuk diterima, ditelusuri, disimpan hingga digunakan sebaik-baiknya dalam dunia maya. (McFedries, 2010:2)

a. Kelebihan Google Chrome

- **Tampilan** : Pada *Interface* dari Chrome terlihat bahwa google ingin para penggunanya lebih fokus pada web dan melupakan browser yang digunakan. Ini artinya Google Chrome memiliki tampilan yang nyaman ketika sedang digunakan.
- **TAB Penyamaran** : memungkinkan penggunanya dapat mengakses website tanpa meninggalkan jejak.
- **Aplikasi Web**: terdapat opsi "*Make Application Shortcut*". Dengan opsi tersebut sebuah aplikasi web seperti Google E-Mail (Gmail) dapat dijalankan lewat *shortcut* pada Desktop atau *Start Menu* sehingga terlihat seperti sebuah aplikasi terinstall.
- **Pengelolaan Memory** : Pada setiap TAB yang dibuka di Chrome memiliki proses yang terpisah, sehingga ketika *error/crash* pada salah satu TAB tidak akan menyebabkan seluruh browser *error* (Fitur ini merupakan kekurangan yang terdapat pada FireFox dan browser lainnya)
- **Pencarian** : Chrome memiliki fungsi pencarian yang sangat baik. Contohnya, chrome dapat mendeteksi ketika pengguna pernah melakukan pencarian di suatu website dan memasukkan website tersebut dalam daftar penyedia pencarian.
- **Anti Penipuan** : Chrome menyediakan kemampuan menebalkan *nama domain* sebuah *website*. Contohnya sebuah website penipuan beralamatkan *ibank.klikbca.d60pc.com*, maka *d60pc.com* akan ditebalkan sehingga akan diharapkan para pengguna sadar bawah itu bukan situs/website resmi KlikBCA.

b. Kelemahan Google Chrome

- **Privasi** : google menyimpan 2% *interpretation* pencarian pengguna lengkap dengan alamat IP nya walaupun dalam

beberapa waktu tertentu *interpretation* ini akan di *anonimkan*. Artinya google bisa saja tahu “siapa, mencari apa, dan dimana”

- **Lisensi** : google sempat mencantumkan pada *terms of service* mereka, bahwa semua muatan dari pengguna yang hak ciptanya dimiliki oleh pengguna akan diserahkan haknya pada google. Akan tetapi *indicate* ini telah dicabut oleh pihak google.
- **Celah Keamanan** : beberapa pakar *confidence* menemukan adanya celah/*bugs* pada chrome. sehingga ketika membuka suatu halaman *website* akan membuat *browser* ini menjadi *crash*. Lalu chrome juga memiliki *underline download* otomatis yang dikhawatirkan akan disalahgunakan oleh *hacker*.

#### 4. CodeIgniter versi 3.1.4

CodeIgniter merupakan aplikasi sumber terbuka yang berupa kerangka kerja PHP dengan model MVC (*Model, View, Controller*) untuk membangun situs web *dinamis* dengan menggunakan PHP. CodeIgniter memudahkan pengembang web untuk membuat aplikasi web dengan cepat dan mudah dibandingkan dengan membuatnya dari awal. CodeIgniter dirilis pertama kali pada 28 Februari 2006.



Gambar III.8 Logo CodeIgniter

Menurut Supono dan Putratama (2016:109) “Codeigniter adalah aplikasi *open source* yang berupa *framework* dengan model MVC (*Model, View, Controller*) untuk membangun *website dinamis* dengan menggunakan PHP ”.

Menurut Widodo (2013:6) “Codeigniter merupakan *framework* PHP yang diklaim memiliki eksekusi tercepat dibandingkan dengan *framework* lainnya. Codeigniter bersifat *open source* dan menggunakan model basic MVC (*Model, View, Controller*), yang merupakan model konsep modern *framework* yang digunakn saat ini”.

Sedangkan menurut Purbadian (2016:18) “Codeigniter adalah sebuah *framework* yang dibuat menggunakan bahasa pemrograman PHP yang bertujuan untuk memudahkan para programmer web untuk membuat atau mengembangkan aplikasi berbasis web”.

Berdasarkan pengertian diatas dapat ditarik kesimpulan bahwa Codeigniter adalah aplikasi *open source* yang merupakan *framework* PHP dengan eksekusi tercepat dan menggunakan model basic MVC (*Model, View, Controller*) untuk membangun sebuah website yang *dinamis* dan bertujuan untuk memudahkan para programmer web untuk membuat atau mengembangkan aplikasi berbasis web.

Dengan menggunakan prinsip MVC, suatu aplikasi dapat dikembangkan sesuai dengan kemampuan pengembangnya, yaitu pemrogram yang menangani bagian model dan *controller*, sedangkan desainer yang menangani bagian *view*, sehingga penggunaan arsitektur MVC dapat meningkatkan pemeliharaan dan pengorganisasian kode. Walaupun demikian, tetap dibutuhkan komunikasi yang baik antara pemrogram dan desainer dalam menangani variabel-variabel yang akan ditampilkan.

a. Kelebihan CodeIgniter

- **Berukuran sangat kecil.** File download hanya sekitar 2MB, itupun sudah termasuk dokumentasinya yang sangat lengkap.
- **Dokumentasi yang bagus.** Telah disertakan dengan dokumentasi yang berisi pengantar, tutorial, bagaimana panduan penggunaan, serta referensi dokumentasi untuk komponen-komponennya.
- **Kompatibilitas dengan Hosting.** CodeIgniter mampu berjalan dengan baik pada hampir semua platform hosting. CodeIgniter juga mendukung database-database paling umum, termasuk MySQL.
- **Tidak ada aturan coding yang ketat.** Pengguna bisa menentukan hanya ingin menggunakan *Controller*, tanpa *View*, atau tidak menggunakan Model, atau tidak salah satu

keduanya. Namun dengan menggunakan ketiga komponennya adalah pilihan paling tepat.

- **Kinerja yang baik.** Codeigniter memiliki kinerja yang sangat cepat.
- **Sangat mudah diintegrasikan.** Codeigniter sangat mengerti tentang pengembangan berbagai *library* saat ini. Karenanya Codeigniter memberikan kemudahan untuk diintegrasikan dengan *library-library* yang tersedia saat ini.
- **Sedikit Konfigurasi.** Konfigurasi Codeigniter terletak di folder *application/config*. Codeigniter tidak membutuhkan konfigurasi yang rumit, bahkan untuk mencoba menjalankannya, tanpa melakukan konfigurasi sedikitpun sudah bisa berjalan.
- **Mudah dipelajari.** Disamping dokumentasi yang lengkap, juga memiliki berbagai forum diskusi.

b. Kekurangan Codeigniter

- Codeigniter tidak untuk pembuatan web dengan skala besar.
- ***Library* yang sangat terbatas.** Hal ini dikarenakan sangat sulit mencari plugin tambahan yang terverifikasi secara resmi, karena pada situsnya Codeigniter tidak menyediakan plugin-plugin tambahan untuk mendukung pengembangan aplikasi dengan Codeigniter.
- **Belum adanya editor khusus Codeigniter**, sehingga dalam melakukan *create project* dan modul-modulnya harus berpindah-pindah folder.

Perubahan (*changelog*) Codeigniter 3.1.4:

### Security

- Memperbaiki fungsi kerentanan injeksi header pada fungsi umum `set_status_header()` di bawah Apache.
- Memperbaiki masalah keamanan byte di Encrypt Library (lama) saat `mbstring.func_overload` diaktifkan.
- Memperbaiki masalah keamanan byte dalam fungsi kompatibilitas `password_hash()`, `hash_pbkdf2()` saat `mbstring.func_overload` diaktifka.

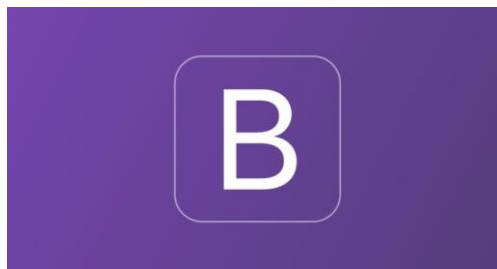
- Perpustakaan Enkripsi yang Diperbarui (lama) untuk memanggil `mcrypt_create_iv()` dengan `MCRYPT_DEV_RANDOM`.

### Perubahan Umum

- Memperbarui Perpustakaan Manipulasi Gambar untuk mengatasi masalah dengan beberapa JPEG saat menggunakan GD.

## 5. Bootstrap versi 3.3.7

Bootstrap adalah kerangka kerja CSS sumber terbuka dan bebas untuk merancang situs web dan aplikasi web. Kerangka kerja ini berisi templat desain berbasis HTML dan CSS untuk tipografi, formulir, tombol, navigasi, dan komponen antarmuka lainnya, serta juga ekstensi opsional JavaScript. Tidak seperti kebanyakan kerangka kerja web lainnya, kerangka kerja ini hanya fokus pada pengembangan *front-end* saja.



Gambar III.9 Logo Bootstrap

Menurut (Nugroho & Setiyawati, 2019), bootstrap adalah framework css untuk membuat tampilan web. Bootstrap menyediakan class dan komponen yang sudah siap dipakai.

### a. Kelebihan Bootstrap

- Dapat digunakan untuk membantu mempercepat saat membuat *front-end* situs web.
- Bootstrap memiliki tampilan yang modern sehingga dapat membuat situs web memiliki penampilan yang menarik.
- Bootstrap akan membuat situs web terlihat lebih responsif. Jadi, ketika menggunakan ponsel cerdas atau komputer untuk membuat situs web, akan menyesuaikan dengan ukuran layar.
- Bootstrap juga akan membuat situs web lebih ringan saat dibuka. Jadi ketika yang mengakses situs web tidak perlu



menunggu lama. Ini karena sebelumnya proses bootstrap dilakukan dengan rapi dan sistematis.

- Dengan bootstrap bisa mendapatkan semua jenis warna, operasi penggunaan, dan juga berbagai variabel yang dibutuhkan. Jadi, tidak perlu khawatir lagi karena hasilnya tentu tidak akan berbeda ketika membukanya di perangkat yang berbeda.
- Bootstrap dikenal lebih lengkap karena itu termasuk CSS, HTML dan Javascript.
- *Open source*, artinya tidak perlu harus mendapatkan lisensi dan bahkan dapat mengembangkan situs web sesuai dengan kebutuhan.
- Kerangka kerja yang dimiliki oleh Bootstrap menggunakan *Less*. *Less* adalah teknologi dari CSS yang terkenal dengan kesederhanaannya.

#### b. Kekurangan Bootstrap

- Karena menggunakan CSS3, bootstrap minimal.
- Karena terlalu fokus pada pengkodean, sehingga membuat pengembang/pemrogram kurang kreatif mendesain situs web sesuai dengan yang benar-benar diinginkan.
- Bootstrap tampaknya tidak dapat menampilkan situs web yang sama di semua browser. Maka hanya browser tertentu yang menunjukkan situs web bootstrap sesuai ukurannya dengan layar dari perangkat yang dipakai.

Perubahan (*changelog*) Bootstrap versi 3.3.7:

#### **JavaScript**

- Memperbaiki navigasi keyboard untuk tombol pilihan (kotak centang, radio, tunggal) setelah klik mouse
- Menghapus `$element` tooltip untuk mencegah kebocoran memori
- Hindari memanggil `jQuery('#')` karena ini adalah kesalahan sintaks di jQuery 3

- Perbaiki pengujian unit JS backport terkait jQuery dari v4
- `button.js`: Setel properti yang dinonaktifkan selain atribut yang dinonaktifkan, untuk kompatibilitas jQuery 3
- Hindari menggunakan `$.offset()` pada SVG karena memberikan hasil yang salah di jQuery 3
- Memperbarui pemeriksaan versi jQuery dan rentang versi ketergantungan jQuery untuk memungkinkan jQuery 3

## CSS

Tidak ada perubahan yang signifikan.

## Aksesibilitas

menambahkan `aria-label` dan info tentang pelabelan komponen pagination

## Dokumentasi

- Bump `html5shiv` ke v3.7.3
- Port v4 browser mendukung format tabel ke v3
- Memperbarui jQuery ke v1.12.4
- Pembaruan untuk mengatasi Browser Bugs

## 6. Balsamiq

Balsamiq merupakan salah satu *tool* yang biasa digunakan oleh para UI desainer ataupun UX desainer untuk merancang desain tampilan aplikasi yang akan dibuat. Menurut website resmi Balsamiq <https://balsamiq.com/>. “Balsamiq Mockups adalah alat wireframing cepat yang membantu Anda bekerja lebih cepat & lebih pintar”. Balsamiq Mockups menciptakan pengalaman sketsa di papan tulis, tetapi menggunakan komputer, membuat mockups menjadi cepat. Menurut (Hanifah, 2015) Balsamiq Mockup adalah salah satu software yang di gunakan dalam pembuatan desain atau prototype dalam pembuatan tampilan user interface sebuah aplikasi.

Aplikasi ini dianggap cukup ramah pemula karena tidak membutuhkan kode untuk bisa mengoperasikannya. Penggunaan *tool* pada aplikasi ini cukup men-*drag* dan *drop* elemen-elemen desain yang diperlukan

sehingga proses desain pun bisa menjadi lebih cepat. Balsamiq juga bisa digunakan untuk membuat *prototype* interaktif untuk situs atau aplikasi yang sedang dirancang. Fitur ini membuat desainer bisa mengetahui *usability* rancangannya sehingga revisi bisa dikurangi pada tahap-tahap selanjutnya.



Gambar III.10 Logo/icon Balsamiq

a. Kelebihan balsamiq :

- Aplikasi desain yang mudah digunakan karena tampilan antarmuka yang mudah dimengerti oleh semua kalangan.
- Elemen-elemen yang diperlukan dalam mendesain tampilan aplikasi cukup lengkap.
- Kustomiasi elemen yang bisa dilakukan dengan mudah.
- Fitur yang sangat memudahkan proses kerja dengan Balsamiq adalah fitur kolaborasinya dengan *sharing control* yang fleksibel dan cepat.
- Hasil *wireframe* yang telah dibuat bisa langsung di-*export* dalam beberapa format, seperti PDF, PNG, dan JSON.

b. Kekurangan balsamiq:

- Sulitnya men-*scroll* library elemen UI yang dimilikinya.
- pembuatan *sitemap* di Balsamiq cukup terbatas.
- Balsamiq hanya bisa memfasilitasi *sitemap* sederhana.
- Jika ingin membuat yang lebih kompleks, butuh *software* atau *tool* lainnya karena Balsamiq berfokus untuk membuat *wireframe*.

## 7. Microsoft Visio

Microsoft Visio merupakan sebuah *software* komputer yang biasanya digunakan untuk membuat diagram, diagram alir, *brainstorm*, UML, skema jaringan dan lain sebagainya. Software ini menggunakan grafik vektor untuk membuat diagramnya.

Dengan software ini dapat membantu pengguna dalam meningkatkan kinerja khususnya dalam pembuatan diagram-diagram/skema-skema. Selain itu ada juga sejumlah template dan elemen yang disediakan, sehingga dapat memungkinkan memudahkan pengguna untuk membuat diagramnya secara rapih, intuitif serta profesional.



Gambar III.11 Microsoft Visio

a. Kelebihan Microsoft Visio

- *User friendly*
- Buat diagram profesional dengan mudah
- Tools yang mudah dipahami
- Banyak pilihan gambar
- Tersedia template menarik
- Fitur kolaborasi

b. Kekurangan Microsoft Visio

- Hanya untuk platform windows
- *Software* berbayar/ bukan *free software*

## **BAB IV**

### **PELAKSANAAN KERJA PRAKTEK**

#### **IV.1 Input**

Dalam mempelajari metodologi pembuatan aplikasi aset laboratorium komputer, diberikan informasi mengenai data yang diperlukan dalam pembuatan aplikasi aset laboratorium komputer yang berisi keterangan mengenai laboratorium komputer FTI UNIBBA terutama mengenai aset laboratorium komputer. Dasar teori yang dipelajari selama perkuliahan menjadi input yang berharga dalam proses pelaksanaan kerja praktek, karena dalam pelaksanaan kerja praktek banyak mengaplikasikan dasar teori yang pernah dipelajari untuk menyelesaikan permasalahan di dunia nyata seperti menyelesaikan kerja praktek ini.

Pada aplikasi aset laboratorium komputer ini, menangani inputan-inputan:

1. Home
2. Tentang FTI UNIBBA  
Menampilkan informasi tentang visi dan misi FTI UNIBBA serta deskripsi lengkap tentang aplikasi aset laboratorium komputer.
3. Form Login (username, password, stts)
4. Dashboard
5. Form Data peralatan(kode alat, nama, kode merek, kode kategori, deskripsi, kode ruang, status),
6. Form Data merek(kode merek, merek),
7. Form Data kategori(kode kategori, kategori),
8. Form Data ruang(kode ruang, ruang, lokasi),
9. Form Data laboran(id laboran, nama, img, jenis kelamin, alamat, password, kode kompetensi, hp, ket),
10. Form Data kompetensi(kode kompetensi, kompetensi),
11. Form Data pengadaan(kode baru, tanggal, kode alat, qty, id laboran)
12. Form Data perbaikan(kode perbaikan, tanggal, kode alat, ket, id laboran)

13. Form Data pengembalian(kode kembali, kode perbaikan, tanggal, kode alat, id laboran)
14. Logout

## IV.2 Proses

Setelah melakukan observasi saat pelaksanaan kerja praktek, selanjutnya proses kerja praktek dapat dibagi menjadi beberapa tahap, yaitu eksplorasi dalam pembuatan aplikasi aset laboratorium komputer, dan pelaporan hasil kerja praktek.

### IV.2.1 Eksplorasi

Tahap eksplorasi dimulai dengan melakukan eksplorasi mengenai metodologi yang akan digunakan dalam pembuatan aplikasi aset laboratorium komputer. Untuk mendukung pelaksanaan metodologi *Waterfall*, diperlukan pula pengetahuan mengenai pemodelan dengan menggunakan *Unified Modelling Language* (UML). Dengan demikian, pendalaman terhadap pemodelan dengan UML pun dilakukan.

Eksplorasi juga dilakukan terhadap teknologi yang akan dipakai dalam pembuatan aplikasi aset laboratorium komputer. Sebagai acuan utama dalam mempelajari pemrograman berbasis web, PHP sebagai bahasa pemrograman yang digunakan pada codeigniter yang dimana sebagai framework *back-end* serta MySQL sebagai database. Lalu untuk tampilan aplikasi, menggunakan template dari bootstrap. Pada tahap ini dilakukan tidak hanya pembelajaran tapi juga pencarian alternatif teknologi yang akan diterapkan. Dengan demikian, tidak seluruh hasil eksplorasi pada akhirnya diterapkan dalam pembuatan aplikasi aset laboratorium komputer. Selama proses eksplorasi ini, dilakukan pula instalasi *tools* yang diperlukan.

Proses eksplorasi masih berlangsung selama pembuatan aplikasi aset laboratorium komputer. Hal ini dimaksudkan untuk menyelaraskan antara hasil eksplorasi dengan penerapannya pada aplikasi yang sedang dibangun.

### 1. Analisis Kebutuhan Sistem

Perencanaan dan pembuatan aplikasi aset laboratorium komputer yang dilakukan dimulai dengan analisis kebutuhan perangkat lunak dan perangkat keras yang diperlukan.

### 2. Kebutuhan Perangkat Keras

Perangkat keras yang digunakan untuk pembuatan aplikasi aset laboratorium komputer ini dengan spesifikasi:

**Tabel IV.1 Kebutuhan Perangkat Keras**

Processor	Intel(R) Atom(TM) CPU N2600 @ 1.60GHz
RAM	2 GB
SSD	256 GB

### 3. Kebutuhan Perangkat Lunak

Berikut kebutuhan perangkat lunak yang digunakan untuk membuat dan menjalankan aplikasi aset laboratorium komputer:

**Tabel IV.2 Kebutuhan Perangkat Lunak**

Sistem Operasi	Windows 8.1
Server	XAMPP versi 3.2.2
Kode Editor	Visual Studio Code
Browser	Google Chrome
Framework Back-End	Codeigniter versi 3.1.4
Framework Front-End	Bootstrap versi 3.3.7

## IV.2.2 Pembangunan Perangkat Lunak

Pembangunan perangkat lunak yang dilakukan dimulai terlebih dahulu dengan melakukan analisis kebutuhan perangkat lunak. Berdasarkan kebutuhan perangkat lunak tersebut, dilakukan perancangan perangkat lunak. Pembangunan perangkat lunak dilakukan berdasarkan perancangan tersebut. Untuk memastikan perangkat lunak yang dibuat sesuai dengan kebutuhan dan berfungsi dengan semestinya, dilakukan beberapa kegiatan pendukung lainnya seperti melakukan pengujian, *bug fixing*, dan optimasi performansi.

Dalam membangun perangkat lunak ini, digunakan metodologi sesuai hasil eksplorasi. Pembangunan perangkat lunak ini juga memanfaatkan berbagai teknologi yang telah dipelajari pada tahap sebelumnya.

Metode perancangan dan pembangunan perangkat lunak yang digunakan adalah Metodologi *Waterfall*, yaitu bertujuan agar memperoleh tahapan perancangan yang lebih baik karena tahapan yang digunakan memiliki proses yang berurutan mulai dari analisa hingga support, sehingga dalam pembuatannya membutuhkan analisa yang penuh mengenai kebutuhan aplikasi yang akan dirancang.

Setiap prosesnya juga memiliki spesifikasinya sendiri, sehingga sebuah sistem dapat bekerja sesuai dengan apa yang dikehendaki (tepat sasaran) dan juga setiap prosesnya tidak saling tumpang tindih, dengan itu digunakannya metode *waterfall* memudahkan dalam pembuatan aplikasi aset laboratorium komputer ini.

1. Analisis Sistem

Analisis sistem dilakukan dengan cara survei dan wawancara dengan pihak kepala pengurus laboratorium komputer FTI UNIBBA mengenai aplikasi aset laboratorium komputer yang akan dibuat. Dilakukan juga pencarian referensi baik dari buku, jurnal maupun dari internet mengenai aplikasi aset laboratorium komputer ini. Setelah melakukan analisa, didapatkan hasil berbagai kebutuhan sistem dan perangkat.

2. Perancangan Aplikasi Aset Laboratorium Komputer

- a. Perancangan Perangkat

Perancangan perangkat yang dilakukan dimulai dengan analisis kebutuhan perangkat lunak (*software*) dan perangkat keras (*hardware*). Selanjutnya, berdasarkan kebutuhan tersebut, dilakukan perancangan aplikasi aset laboratorium komputer.

- b. *Use Case Diagram*

Pada rancangan *Use Case Diagram* dapat dilihat apa saja yang dapat dilakukan admin untuk mengelola data yang terdapat pada Aplikasi Aset Laboratorium Komputer, dirancang tersebut



dijelaskan admin dapat mengelola data peralatan, merek, kategori, ruang, laboran, kompetensi, login, pengadaan, perbaikan dan data pengembalian. Sedangkan untuk laboran, dapat mengelola data yang sama seperti admin kecuali bagian data laboran, kompetensi, dan data ruang.



Gambar IV.1 Use Case Diagram (Awaluddin dkk., 2020)

Berikut adalah skenario jalannya *use case* pada *use case diagram* yang dirancang:

Tabel IV.3 Skenario Use Case Tentang FTI

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. Mengklik Tentang FTI	
	2. Menampilkan halaman tentang FTI

Tabel IV.4 Skenario Use Case Login

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. memasukkan <i>username</i> dan <i>password</i>	
	2. Memeriksa <i>valid</i> tidaknya data

	3. Masuk ke menu data aset laboratorium komputer
Skenario Alternatif	
1. memasukan <i>username</i> dan <i>password</i>	
	2. Memeriksa <i>valid</i> tidaknya data yang dimasukkan dengan <i>database</i>
	3. Menampilkan pesan <i>login</i> tidak <i>valid</i>
4. Memasukkan <i>username</i> dan <i>password</i> yang <i>valid</i>	
	5. Memeriksa <i>valid</i> tidaknya data yang dimasukkan dengan <i>database</i>
	6. Masuk ke menu data aset laboratorium komputer

Tabel IV.5 Skenario Use Case Memeriksa Status Login

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa ke <i>variabel session</i> sebagai penanda login apakah <i>user</i> sudah login
	2. Mengembalikan status login apakah sudah login atau belum

Tabel IV.6 Skenario Use Case Menampilkan/Melihat Data

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data yang akan ditampilkan	
	3. Menampilkan data yang dipilih (semua kolom) dari menu data yang dipilih

Tabel IV.7 Skenario Use Case Mengubah Data Peralatan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data peralatan	
	3. Melihat data peralatan
4. Memilih data yang akan diubah	

	5. Menampilkan data (semua kolom) yang dipilih
6. Mengubah data yang dipilih	
	7. Memeriksa <i>valid</i> tidaknya data yang diubah
	8. Menyimpan data yang diubah ke <i>database</i>
	9. Menampilkan pesan data sukses disimpan
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data peralatan	
	3. Melihat data peralatan
4. Memilih data yang akan diubah	
	5. Menampilkan data (semua kolom) yang dipilih
6. Mengubah data yang dipilih	
	7. Memeriksa <i>valid</i> tidaknya data yang diubah
	8. Menampilkan pesan bahwa data yang diubah tidak <i>valid</i>
9. Memperbaiki data masukkan yang diubah tidak <i>valid</i>	
	10. Memeriksa <i>valid</i> tidaknya data yang diubah
	11. Menyimpan data yang diubah ke <i>database</i>
	12. Menampilkan pesan data sukses disimpan

Tabel IV.8 Skenario Use Case Menghapus Data Peralatan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data peralatan	
	3. Melihat data peralatan
4. Memilih data yang akan dihapus	
	5. Menampilkan pesan konfirmasi penghapusan data
6. Mengkonfirmasi penghapusan data	
	7. menghapus data dari <i>database</i>
	8. Menampilkan pesan data sukses dihapus

Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data peralatan	
	3. Menampilkan data peralatan
4. Memilih data yang akan dihapus	
	5. Menampilkan pesan konfirmasi penghapusan data
6. Tidak mengkonfirmasi penghapusan data	
	7. Kembali ke menu data peralatan

Tabel IV.9 Skenario Use Case Mencetak/Export Data Peralatan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu peralatan	
	3. Menampilkan data peralatan
4. Mengklik cetak data	
	5. Menampilkan data yang akan di cetak/export
6. Memilih cetak/export data	
	7. Mencetak/export data

Tabel IV.10 Skenario Use Case Menambah Data Merek

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data merek	
	3. Menampilkan data merek
4. Memasukkan data sesuai kolom	
	5. <i>Validasi</i> data yang dimasukkan
	6. Menyimpan data ke <i>database</i>
	7. Menampilkan pesan data sukses disimpan
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data merek	
	3. Menampilkan data merek

4. Memasukkan data sesuai kolom	
	5. <i>Validasi</i> data yang dimasukkan
	6. Menampilkan pesan bahwa data yang dimasukkan tidak <i>valid</i>
7. Memperbaiki data masukkan yang tidak <i>valid</i>	
	8. <i>validasi</i> data yang dimasukkan
	9. Menyimpan data ke <i>database</i>
	10. Menampilkan pesan data sukses disimpan

Tabel IV.11 Skenario *Use Case* Mengubah Data Merek

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data merek	
	3. Melihat data merek
4. Memilih data yang akan diubah	
	5. Menampilkan data (semua kolom) yang dipilih
6. Mengubah data yang dipilih	
	7. Memeriksa <i>valid</i> tidaknya data yang diubah
	8. Menyimpan data yang diubah ke <i>database</i>
	9. Menampilkan pesan data sukses disimpan
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data merek	
	3. Melihat data merek
4. Memilih data yang akan diubah	
	5. Menampilkan data (semua kolom) yang dipilih
6. Mengubah data yang dipilih	
	7. Memeriksa <i>valid</i> tidaknya data yang diubah
	8. Menampilkan pesan diubah tidak <i>valid</i>
9. Memperbaiki data <i>inputan</i>	

	10. Memeriksa <i>valid</i> tidaknya data yang diubah
	11. Menyimpan data yang diubah ke <i>database</i>
	12. Menampilkan pesan data sukses disimpan

Tabel IV.12 Skenario Use Case Menghapus Data Merek

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data merek	
	3. Melihat data merek
4. Memilih data yang akan dihapus	
	5. Menampilkan pesan konfirmasi penghapusan data
6. Mengkonfirmasi penghapusan data	
	7. menghapus data dari <i>database</i>
	8. Menampilkan pesan data sukses dihapus
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data merek	
	3. Menampilkan data merek
4. Memilih data yang akan dihapus	
	5. Menampilkan pesan konfirmasi penghapusan data
6. Tidak mengkonfirmasi penghapusan data	
	7. Kembali ke menu data merek

Tabel IV.13 Skenario Use Case Menambah Data Kategori

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data kategori	
	3. Menampilkan data kategori
4. Memasukkan data sesuai kolom	
	5. <i>Validasi</i> data yang dimasukkan
	6. Menyimpan data ke <i>database</i>

	7. Menampilkan pesan data sukses disimpan
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data kategori	
	3. Menampilkan data kategori
4. Memasukkan data sesuai kolom	
	5. <i>Validasi</i> data yang dimasukkan
	6. Menampilkan pesan bahwa data yang dimasukkan tidak <i>valid</i>
7. Memperbaiki data masukkan yang tidak <i>valid</i>	
	8. <i>validasi</i> data yang dimasukkan
	9. Menyimpan data ke <i>database</i>
	10. Menampilkan pesan data sukses disimpan

Tabel IV.14 Skenario Use Case Mengubah Data Kategori

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data kategori	
	3. Melihat data kategori
4. Memilih data yang akan diubah	
	5. Menampilkan data (semua kolom) yang dipilih
6. Mengubah data yang dipilih	
	7. <i>validasi</i> data yang diubah
	8. Menyimpan data ke <i>database</i>
	9. Menampilkan pesan data sukses disimpan
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data kategori	
	3. Melihat data kategori
4. Memilih data yang akan diubah	
	5. Menampilkan data yang dipilih

6. Mengubah data yang dipilih	
	7. <i>validasi</i> data yang diubah
	8. Menampilkan pesan data diubah tidak <i>valid</i>
9. Memperbaiki data masukkan yang diubah tidak <i>valid</i>	
	10. <i>validasi</i> data yang diubah
	11. Menyimpan data ke <i>database</i>
	12. Menampilkan pesan sukses disimpan

Tabel IV.15 Skenario Use Case Menghapus Data Kategori

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data kategori	
	3. Melihat data kategori
4. Memilih data yang akan dihapus	
	5. Menampilkan pesan konfirmasi penghapusan data
6. Mengkonfirmasi penghapusan data	
	7. menghapus data dari <i>database</i>
	8. Menampilkan pesan data sukses dihapus
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data kategori	
	3. Menampilkan data kategori
4. Memilih data yang akan dihapus	
	5. Menampilkan pesan konfirmasi penghapusan data
6. Tidak mengkonfirmasi penghapusan data	
	7. Kembali ke menu data kategori

Tabel IV.16 Skenario Use Case Menambah Data Ruang

Aksi Aktor (Admin)	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login



2. Memilih menu data ruang	
	3. Menampilkan data ruang
4. Memasukkan data sesuai kolom	
	5. <i>Validasi</i> data yang dimasukkan
	6. Menyimpan data ke <i>database</i>
	7. Menampilkan pesan data sukses disimpan
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data ruang	
	3. Menampilkan data ruang
4. Memasukkan data sesuai kolom	
	5. <i>Validasi</i> data yang dimasukkan
	6. Menampilkan pesan bahwa data yang dimasukkan tidak <i>valid</i>
7. Memperbaiki data masukkan yang tidak <i>valid</i>	
	8. <i>validasi</i> data yang dimasukkan
	9. Menyimpan data ke <i>database</i>
	10. Menampilkan pesan data sukses disimpan

Tabel IV.17 Skenario Use Case Mengubah Data Ruang

Aksi Aktor (Admin)	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data ruang	
	3. Melihat data ruang
4. Memilih data yang akan diubah	
	5. Menampilkan data (semua kolom) yang dipilih
6. Mengubah data yang dipilih	
	7. <i>validasi</i> tidaknya data yang diubah
	8. Menyimpan data ke <i>database</i>
	9. Menampilkan pesan data sukses disimpan

Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data ruang	
	3. Melihat data ruang
4. Memilih data yang akan diubah	
	5. Menampilkan data (semua kolom) yang dipilih
6. Mengubah data yang dipilih	
	7. <i>validasi</i> tidaknya data yang diubah
	8. Menampilkan pesan bahwa data yang diubah tidak <i>valid</i>
9. Memperbaiki data masukkan yang diubah tidak <i>valid</i>	
	10. <i>validasi</i> data yang diubah
	11. Menyimpan data ke <i>database</i>
	12. Menampilkan pesan data sukses disimpan

Tabel IV.18 Skenario Use Case Menghapus Data Ruang

Aksi Aktor (Admin)	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data ruang	
	3. Melihat data ruang
4. Memilih data yang akan dihapus	
	5. Menampilkan pesan konfirmasi penghapusan data
6. Mengkonfirmasi penghapusan data	
	7. menghapus data dari <i>database</i>
	8. Menampilkan pesan data sukses dihapus
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data kategori	
	3. Menampilkan data kategori
4. Memilih data yang akan dihapus	

	5. Menampilkan pesan konfirmasi penghapusan data
6. Tidak mengkonfirmasi penghapusan data	
	7. Kembali ke menu data ruang

Tabel IV.19 Skenario Use Case Menambah Data Laboran

Aksi Aktor (Admin)	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data laboran	
	3. Menampilkan data laboran
4. Memasukkan data sesuai kolom	
	5. <i>Validasi</i> data yang dimasukkan
	6. Menyimpan data ke <i>database</i>
	7. Menampilkan pesan data sukses disimpan
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data laboran	
	3. Menampilkan data laboran
4. Memasukkan data sesuai kolom	
	5. <i>Validasi</i> data yang dimasukkan
	6. Menampilkan pesan bahwa data yang dimasukkan tidak <i>valid</i>
7. Memperbaiki data masukan yang tidak <i>valid</i>	
	8. <i>validasi</i> data yang dimasukkan
	9. Menyimpan data ke <i>database</i>
	10. Menampilkan pesan sukses disimpan

Tabel IV.20 Skenario Use Case Mengubah Data Laboran

Aksi Aktor (Admin)	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data laboran	
	3. Melihat data laboran

4. Memilih data yang akan diubah	
	5. Menampilkan data (semua kolom) yang dipilih
6. Mengubah data yang dipilih	
	7. <i>validasi</i> tidaknya data yang diubah
	8. Menyimpan data ke <i>database</i>
	9. Menampilkan pesan sukses disimpan
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data laboran	
	3. Melihat data laboran
4. Memilih data yang akan diubah	
	5. Menampilkan data (semua kolom) yang dipilih
6. Mengubah data yang dipilih	
	7. <i>validasi</i> tidaknya data yang diubah
	8. Menampilkan pesan bahwa data yang diubah tidak <i>valid</i>
9. Memperbaiki data masukkan yang diubah tidak <i>valid</i>	
	10. <i>validasi</i> data yang diubah
	11. Menyimpan data ke <i>database</i>
	12. Menampilkan pesan sukses disimpan

Tabel IV.21 Skenario Use Case Menghapus Data Laboran

Aksi Aktor (Admin)	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data laboran	
	3. Melihat data laboran
4. Memilih data yang akan dihapus	
	5. Menampilkan pesan konfirmasi penghapusan data
6. Mengkonfirmasi penghapusan data	
	7. menghapus data dari <i>database</i>

	8. Menampilkan pesan data sukses dihapus
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data laboran	
	3. Menampilkan data laboran
4. Memilih data yang akan dihapus	
	5. Menampilkan pesan konfirmasi penghapusan data
6. Tidak mengkonfirmasi penghapusan data	
	7. Kembali ke menu data laboran

Tabel IV.22 Skenario Use Case Menambah Data Kompetensi

Aksi Aktor (Admin)	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data kompetensi	
	3. Menampilkan data kompetensi
4. Memasukkan data sesuai kolom	
	5. <i>Validasi</i> data yang dimasukkan
	6. Menyimpan data ke <i>database</i>
	7. Menampilkan pesan data sukses disimpan
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data kompetensi	
	3. Menampilkan data kompetensi
4. Memasukkan data sesuai kolom	
	5. <i>Validasi</i> data yang dimasukkan
	6. Menampilkan pesan bahwa data yang dimasukkan tidak <i>valid</i>
7. Memperbaiki data masukkan yang tidak <i>valid</i>	
	8. <i>validasi</i> data yang dimasukkan
	9. Menyimpan data ke <i>database</i>
	10. Menampilkan pesan sukses disimpan

Tabel IV.23 Skenario Use Case Mengubah Data Kompetensi

Aksi Aktor (Admin)	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data kompetensi	
	3. Melihat data kompetensi
4. Memilih data yang akan diubah	
	5. Menampilkan data (semua kolom) yang dipilih
6. Mengubah data yang dipilih	
	7. <i>validasi</i> tidaknya data yang diubah
	8. Menyimpan data ke <i>database</i>
	9. Menampilkan pesan sukses disimpan
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data kompetensi	
	3. Melihat data kompetensi
4. Memilih data yang akan diubah	
	5. Menampilkan data (semua kolom) yang dipilih
6. Mengubah data yang dipilih	
	7. <i>validasi</i> tidaknya data yang diubah
	8. Menampilkan pesan bahwa data yang diubah tidak <i>valid</i>
9. Memperbaiki data masukkan yang diubah tidak <i>valid</i>	
	10. <i>validasi</i> data yang diubah
	11. Menyimpan data ke <i>database</i>
	12. Menampilkan pesan sukses disimpan

Tabel IV.24 Skenario Use Case Menghapus Data Kompetensi

Aksi Aktor (Admin)	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data kompetensi	
	3. Melihat data kompetensi

4. Memilih data yang akan dihapus	
	5. Menampilkan pesan konfirmasi penghapusan data
6. Mengkonfirmasi penghapusan data	
	7. menghapus data dari <i>database</i>
	8. Menampilkan pesan data sukses dihapus
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data kompetensi	
	3. Menampilkan data kompetensi
4. Memilih data yang akan dihapus	
	5. Menampilkan pesan konfirmasi penghapusan data
6. Tidak mengkonfirmasi penghapusan data	
	7. Kembali ke menu data kompetensi

Tabel IV.25 Skenario Use Case Menambah Data Pengadaan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data pengadaan	
	3. Menampilkan data pengadaan
4. Memasukkan data sesuai kolom	
	5. <i>Validasi</i> data yang dimasukkan
	6. Menyimpan data ke <i>database</i>
	7. Menampilkan pesan data sukses disimpan
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data pengadaan	
	3. Menampilkan data pengadaan
4. Memasukkan data sesuai kolom	
	5. <i>Validasi</i> data yang dimasukkan
	6. Menampilkan pesan bahwa data yang dimasukkan tidak <i>valid</i>

7. Memperbaiki data masukan yang tidak <i>valid</i>	
	8. <i>validasi</i> data yang dimasukkan
	9. Menyimpan data ke <i>database</i>
	10. Menampilkan pesan sukses disimpan

Tabel IV.26 Skenario Use Case Mengubah Data Pengadaan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data pengadaan	
	3. Melihat data pengadaan
4. Memilih data yang akan diubah	
	5. Menampilkan data (semua kolom) yang dipilih
6. Mengubah data yang dipilih	
	7. <i>validasi</i> data yang diubah
	8. Menyimpan data ke <i>database</i>
	9. Menampilkan pesan sukses disimpan
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data pengadaan	
	3. Melihat data pengadaan
4. Memilih data yang akan diubah	
	5. Menampilkan data (semua kolom) yang dipilih
6. Mengubah data yang dipilih	
	7. <i>validasi</i> data yang diubah
	8. Menampilkan pesan bahwa data yang diubah tidak <i>valid</i>
9. Memperbaiki data masukan yang diubah tidak <i>valid</i>	
	10. <i>validasi</i> data yang diubah
	11. Menyimpan data ke <i>database</i>
	12. Menampilkan pesan sukses disimpan



Tabel IV.27 Skenario Use Case Menghapus Data Pengadaan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data pengadaan	
	3. Melihat data pengadaan
4. Memilih data yang akan dihapus	
	5. Menampilkan konfirmasi hapus data
6. Mengkonfirmasi penghapusan data	
	7. menghapus data dari <i>database</i>
	8. Menampilkan pesan data sukses dihapus
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data pengadaan	
	3. Menampilkan data pengadaan
4. Memilih data yang akan dihapus	
	5. Menampilkan pesan konfirmasi penghapusan data
6. Tidak mengkonfirmasi hapus data	
	7. Kembali ke menu data pengadaan

Tabel IV.28 Skenario Use Case Mencetak/Export Data Pengadaan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data pengadaan	
	3. Menampilkan data pengadaan
4. Mengklik cetak data	
	5. Menampilkan data yang akan di cetak/export
6. Memilih cetak/export data	
	7. Mencetak/export data

Tabel IV.29 Skenario Use Case Menambah Data Perbaikan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login

2. Memilih menu data perbaikan	
	3. Menampilkan data perbaikan
4. Memasukkan data sesuai kolom	
	5. <i>Validasi</i> data yang dimasukkan
	6. Menyimpan data ke <i>database</i>
	7. Menampilkan pesan data sukses disimpan
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data perbaikan	
	3. Menampilkan data perbaikan
4. Memasukkan data sesuai kolom	
	5. <i>Validasi</i> data yang dimasukkan
	6. Menampilkan pesan bahwa data yang dimasukkan tidak <i>valid</i>
7. Memperbaiki data masukkan yang tidak <i>valid</i>	
	8. <i>validasi</i> data yang dimasukkan
	9. Menyimpan data ke <i>database</i>
	10. Menampilkan pesan sukses disimpan

Tabel IV.30 Skenario Use Case Mengubah Data Perbaikan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data perbaikan	
	3. Melihat data perbaikan
4. Memilih data yang akan diubah	
	5. Menampilkan data (semua kolom) yang dipilih
6. Mengubah data yang dipilih	
	7. <i>validasi</i> data yang diubah
	8. Menyimpan data ke <i>database</i>
	9. Menampilkan pesan sukses disimpan
Skenario Alternatif	
	1. Memeriksa status login

2. Memilih menu data perbaikan	
	3. Melihat data perbaikan
4. Memilih data yang akan diubah	
	5. Menampilkan data (semua kolom) yang dipilih
6. Mengubah data yang dipilih	
	7. <i>validasi</i> data yang diubah
	8. Menampilkan pesan bahwa data yang diubah tidak <i>valid</i>
9. Memperbaiki data masukkan yang diubah tidak <i>valid</i>	
	10. <i>validasi</i> data yang diubah
	11. Menyimpan data ke <i>database</i>
	12. Menampilkan pesan sukses disimpan

Tabel IV.31 Skenario Use Case Menghapus Data Perbaikan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data perbaikan	
	3. Melihat data perbaikan
4. Memilih data yang akan dihapus	
	5. Menampilkan pesan konfirmasi penghapusan data
6. Mengkonfirmasi penghapusan data	
	7. menghapus data dari <i>database</i>
	8. Menampilkan pesan bahwa data sukses dihapus
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data perbaikan	
	3. Menampilkan data perbaikan
4. Memilih data yang akan dihapus	
	5. Menampilkan pesan konfirmasi penghapusan data

6. Tidak mengkonfirmasi penghapusan data	
	7. Kembali ke menu data perbaikan

Tabel IV.32 Skenario Use Case Mencetak/Export Data Perbaikan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data perbaikan	
	3. Menampilkan data perbaikan
4. Mengklik cetak data	
	5. Menampilkan data yang akan di cetak/export
6. Memilih cetak/export data	
	7. Mencetak/export data

Tabel IV.33 Skenario Use Case Menambah Data Pengembalian

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data perbaikan	
	3. Menampilkan data perbaikan
4. Memilih menu data pengembalian	
	5. Menampilkan data pengembalian
6. Memasukkan data sesuai kolom	
	7. <i>Validasi</i> data yang dimasukkan
	8. Menyimpan data ke <i>database</i>
	9. Menampilkan pesan data sukses disimpan
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data perbaikan	
	3. Menampilkan data perbaikan
4. Memilih menu data pengembalian	
	5. Menampilkan data pengembalian
6. Memasukkan data sesuai kolom	
	7. <i>Validasi</i> data yang dimasukkan
	8. Menampilkan pesan data <i>input</i> tidak <i>valid</i>

9. Memperbaiki data masukkan yang tidak <i>valid</i>	
	10. <i>validasi</i> data yang dimasukkan
	11. Menyimpan data ke <i>database</i>
	12. Menampilkan pesan sukses disimpan

Tabel IV.34 Skenario Use Case Mengubah Data Pengembalian

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data perbaikan	
	3. Menampilkan data perbaikan
4. Memilih menu data pengembalian	
	5. Menampilkan data pengembalian
6. Memilih data yang akan diubah	
	7. Menampilkan data (semua kolom) yang dipilih
8. Mengubah data yang dipilih	
	9. <i>validasi</i> data yang diubah
	10. Menyimpan data ke <i>database</i>
	11. Menampilkan pesan sukses disimpan
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data perbaikan	
	3. Melihat data perbaikan
4. Memilih menu data pengembalian	
	5. Melihat data pengembalian
6. Memilih data yang akan diubah	
	7. Menampilkan data yang dipilih
8. Mengubah data yang dipilih	
	9. <i>validasi</i> data yang diubah
	10. <i>Menampilkan</i> pesan bahwa data yang diubah tidak <i>valid</i>
11. Memperbaiki data masukkan yang diubah tidak <i>valid</i>	

	12. <i>validasi</i> data yang diubah
	13. Menyimpan data ke <i>database</i>
	14. Menampilkan pesan sukses disimpan

Tabel IV.35 Skenario Use Case Menghapus Data Pengembalian

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login
2. Memilih menu data perbaikan	
	3. Melihat data perbaikan
4. Memilih menu data pengembalian	
	5. Melihat data pengembalian
6. Memilih data yang akan dihapus	
	7. Menampilkan pesan konfirmasi penghapusan data
8. Mengkonfirmasi penghapusan data	
	9. menghapus data dari <i>database</i>
	10. Menampilkan pesan bahwa data sukses dihapus
Skenario Alternatif	
	1. Memeriksa status login
2. Memilih menu data perbaikan	
	3. Menampilkan data perbaikan
4. Memilih menu data pengembalian	
	5. Menampilkan data pengembalian
6. Memilih data yang akan dihapus	
	7. Menampilkan pesan konfirmasi penghapusan data
8. Tidak mengkonfirmasi penghapusan data	
	9. Kembali ke menu data pengembalian

Tabel IV.36 Skenario Use Case Mencetak/Export Data Pengembalian

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	1. Memeriksa status login

2. Memilih menu data perbaikan	
	3. Menampilkan data perbaikan
4. Memilih menu data pengembalian	
	5. Menampilkan data pengembalian
6. Mengklik cetak data	
	7. Menampilkan data yang akan di cetak/export
8. Memilih cetak/export data	
	9. Mencetak/export data

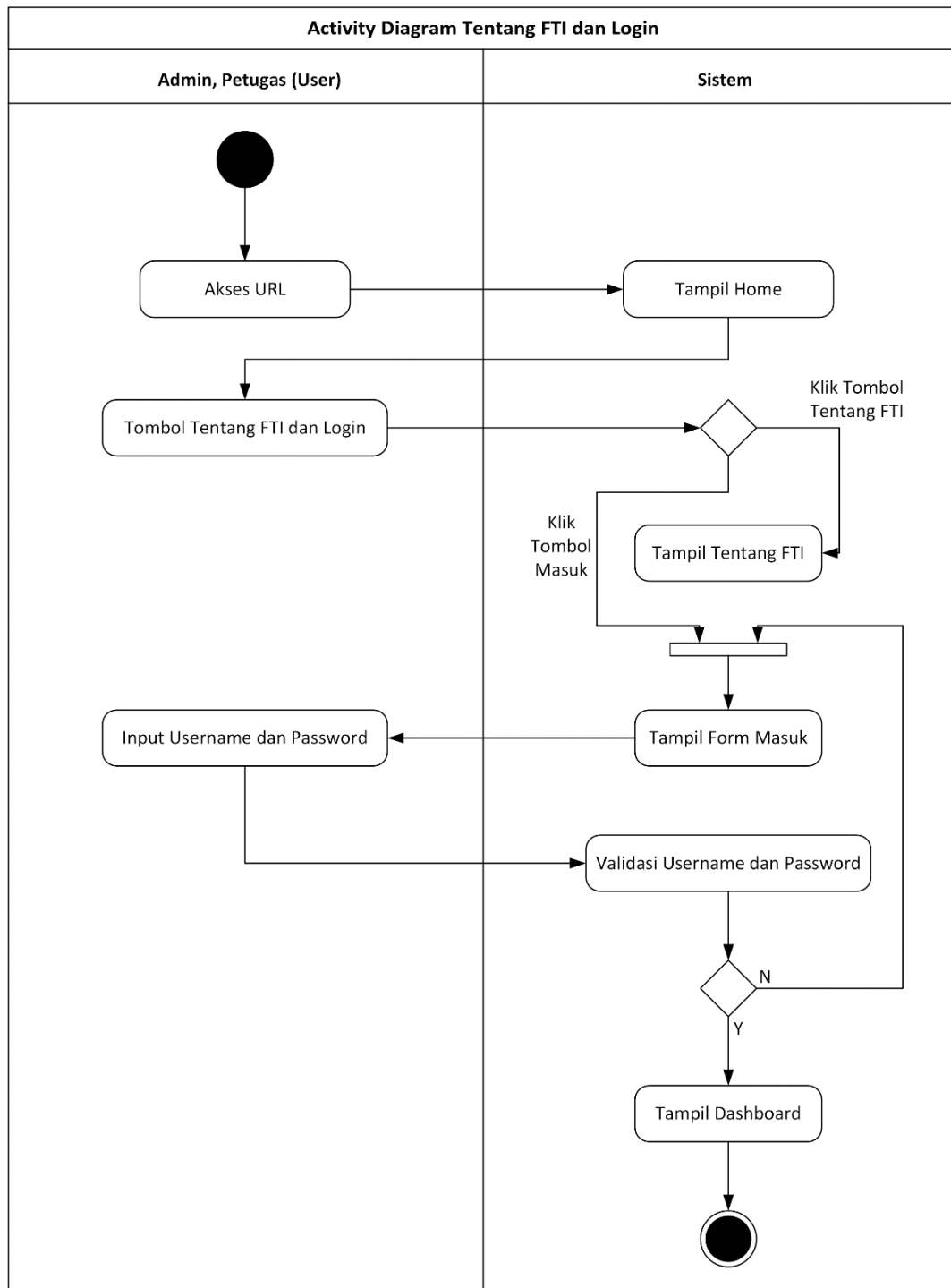
Sumber: (Fitriani dkk., 2018)

b. *Activity Diagram*

Pada activity diagram menjelaskan tentang aktivitas-aktivitas dari *use case diagram* berdasarkan menu yang ada pada aplikasi aset laboratorium komputer. Menu pada aplikasi ini berisi dengan menu-menu:

- 1) Menu home yang memiliki dua menu yaitu menu tentang Fakultas Teknologi Informasi Universitas Bale Bandung dan menu login
- 2) Menu data peralatan
- 3) Menu data kategori
- 4) Menu data ruang
- 5) Menu data merek
- 6) Menu data kompetensi
- 7) Menu data perbaikan
- 8) Menu data pengembalian. Menu ini berada di menu perbaikan
- 9) Menu data pengadaan
- 10) Menu data laboran.

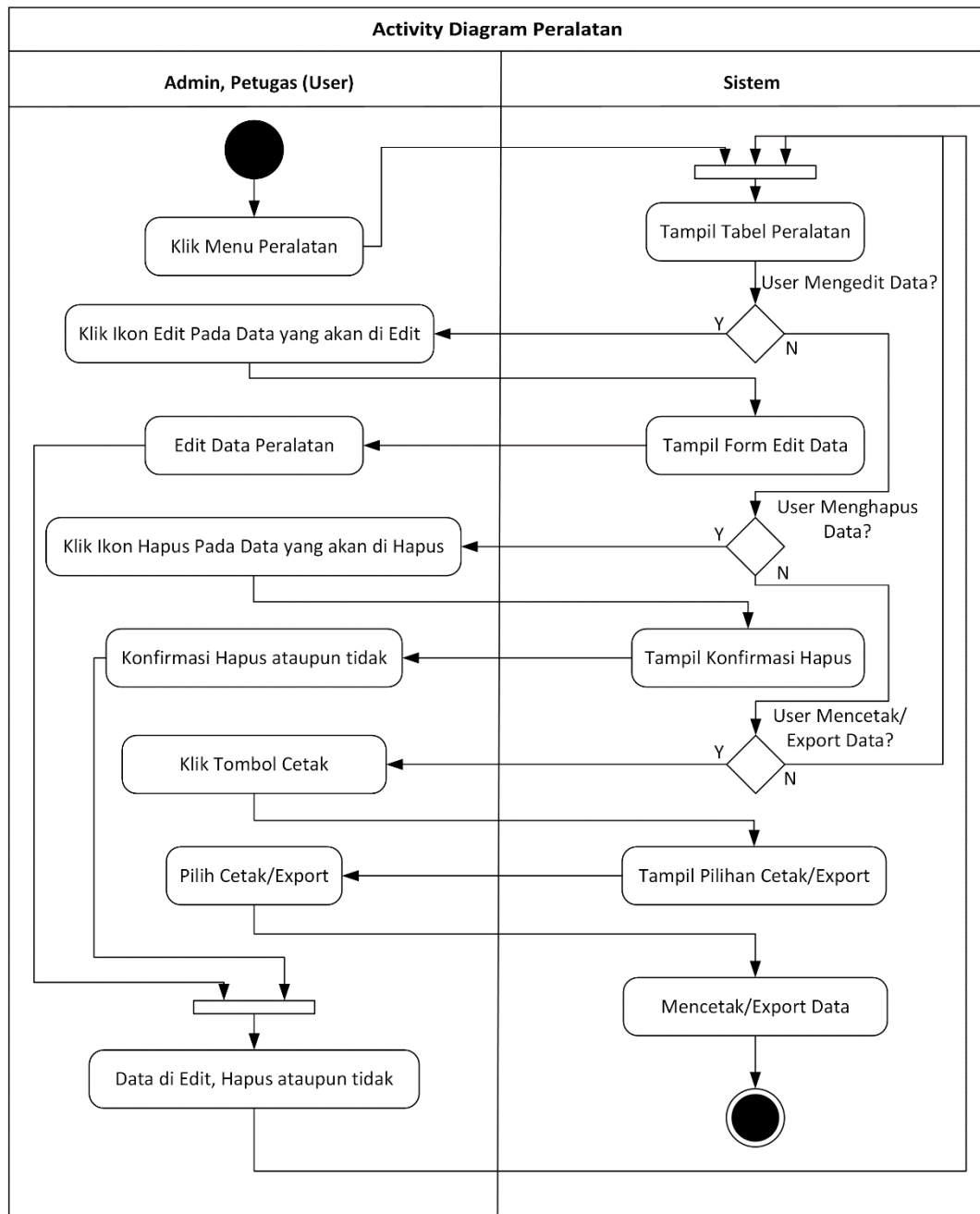
Pada pembuatan *activity diagram*, selain dibuat berdasarkan *use case diagram* juga dibuat berdasarkan skenario *use case diagram* yang telah di buat sebelumnya. Activity diagram yang akan dibuat akan lebih memperjelas alur dari skenario *use case* yang akan diterapkan pada aplikasi yang dibuat.



**Gambar IV.2 Activity Diagram Tentang FTI dan Login**

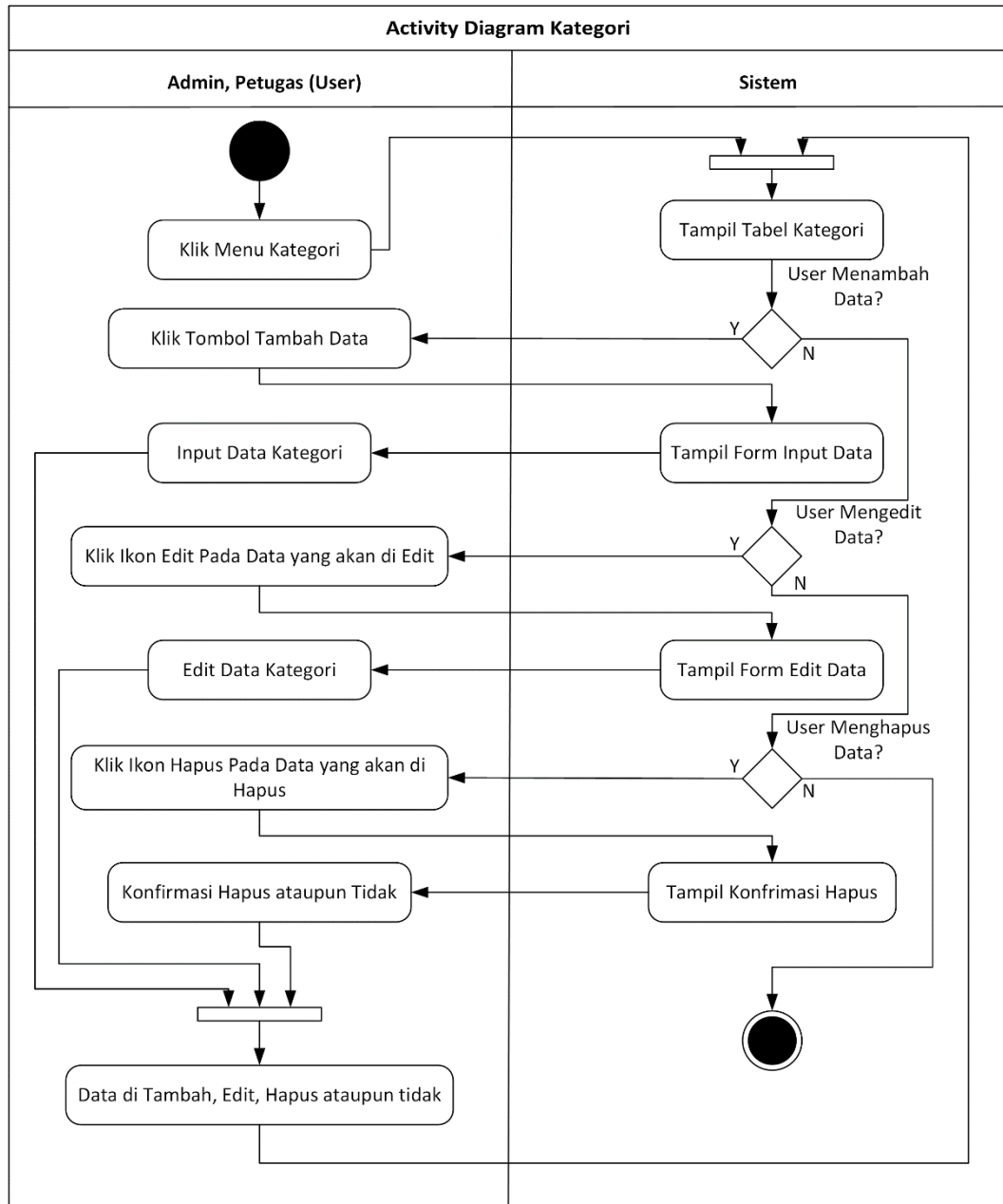
Pada activity diagram ini, menjelaskan alur aplikasi jika *user* mengakses Tentang FTI dan Login. Jika *user* mengakses Tentang FTI, maka *user* akan dialihkan ke halaman yang memuat informasi Tentang FTI dan juga deskripsi lengkap dari aplikasi aset laboratorium komputer. Jika *user* mengakses login, maka akan dialihkan ke halaman form login.





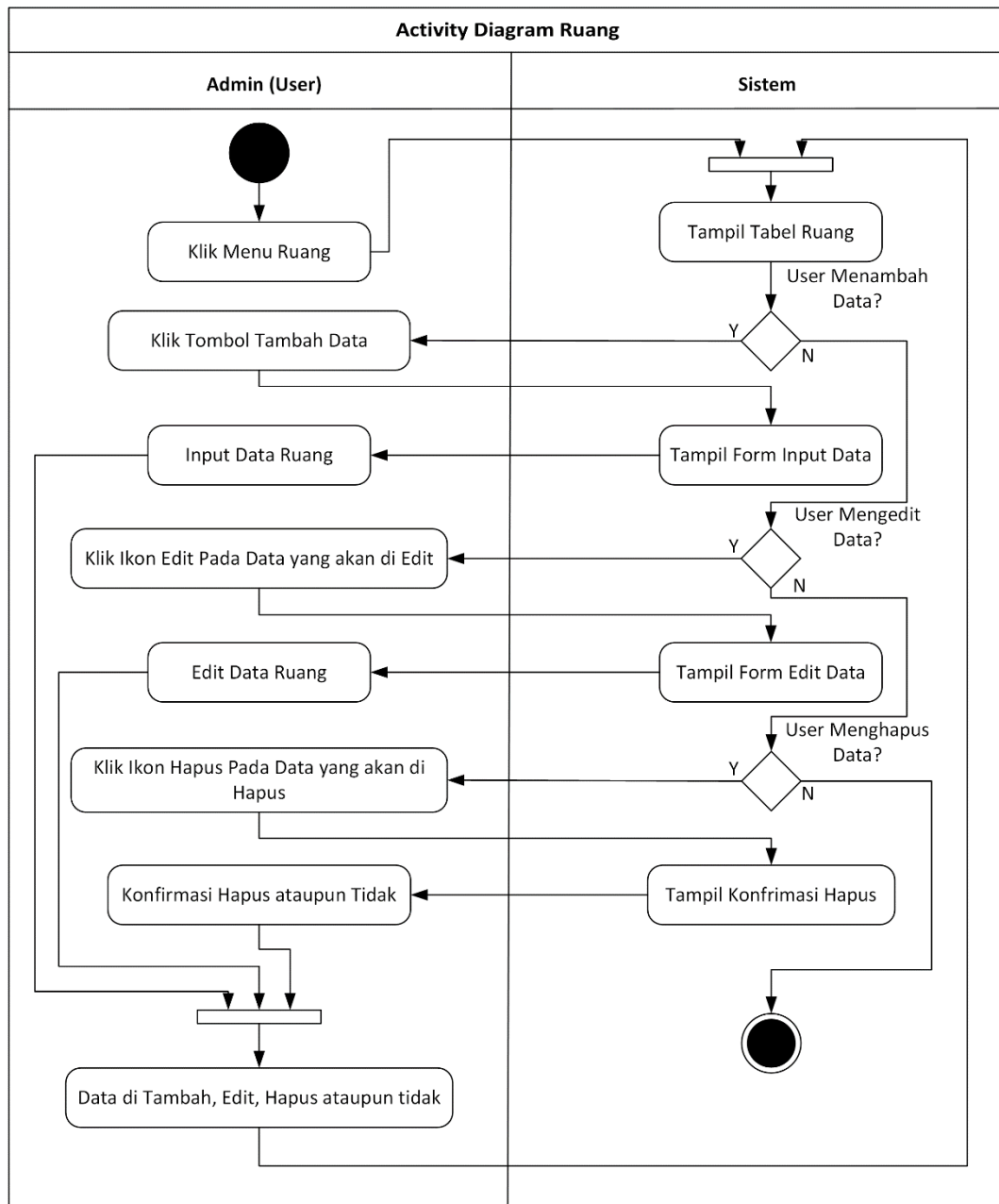
**Gambar IV.3 Activity Diagram Peralatan**

Pada activity diagram peralatan, menjelaskan alur aplikasi jika *user* mengakses menu peralatan. Pada menu peralatan, memiliki 3 *method* yaitu *read*, *update* dan *delete*. Selain ketiga *method* tersebut, pada menu peralatan *user* juga dapat melakukan cetak/export data yang dimana dapat berguna untuk melakukan sebuah pelaporan mengenai data peralatan. Untuk *method create*, menu peralatan tidak memilikinya karena *method create* data peralatan digunakan di menu pengadaan alat.



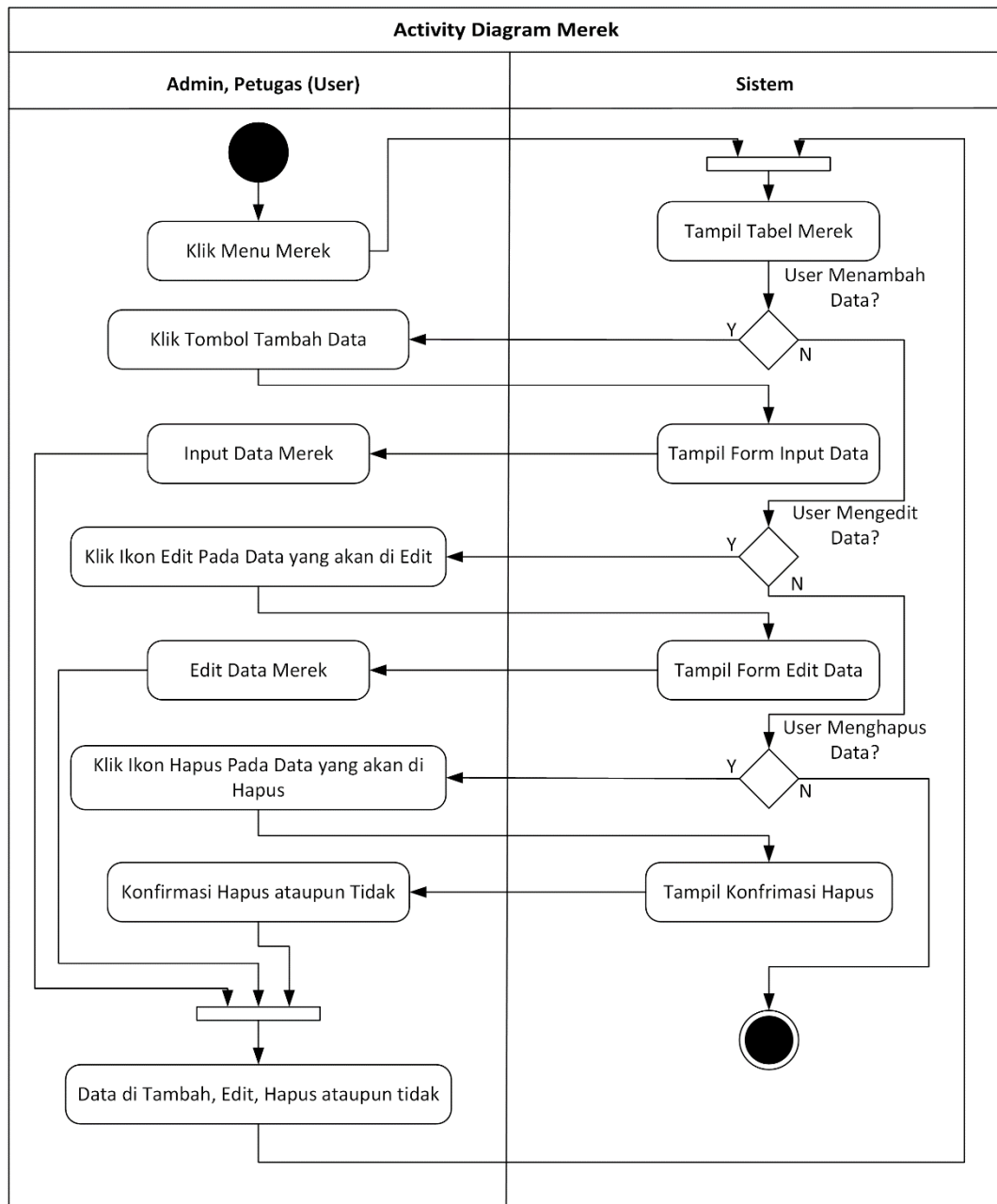
**Gambar IV.4 Activity Diagram Kategori**

Pada activity diagram kategori, menjelaskan alur aplikasi jika *user* mengakses menu kategori. Menu kategori berfungsi untuk mendata kategori dari setiap data peralatan yang terdata, yang dimana dengan adanya informasi kategori, dapat memudahkan *user* untuk melakukan *filter* data peralatan berdasarkan kategori peralatannya sehingga memudahkan dalam mengakses informasi peralatan sesuai dengan yang dibutuhkan. Pada menu kategori, memiliki 4 *method* dalam mengelola datanya yaitu *method create, read, update* dan *delete*.



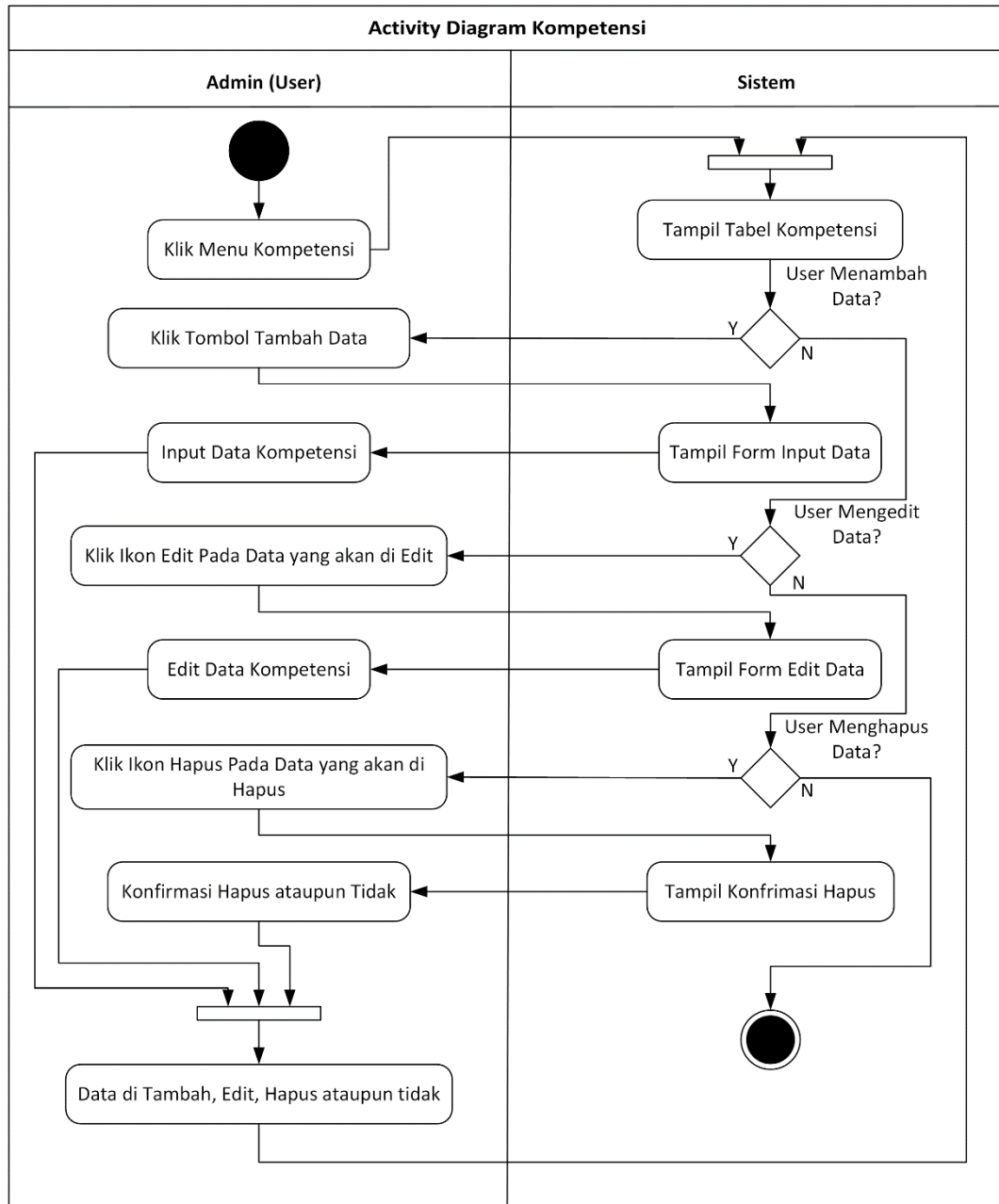
**Gambar IV.5 Activity Diagram Ruang**

Pada activity diagram ruang, menjelaskan alur aplikasi jika *user* mengakses menu ruang. Menu ruang berfungsi untuk mendata ruangan laboratorium komputer yang dimana dengan adanya data mengenai ruangan laboratorium, *user* dapat memiliki informasi mengenai peralatan yang terdata berada di ruangan laboratorium yang mana. Selain itu, *user* juga dapat melakukan *filter* data peralatan berdasarkan ruangan laboratorium. Pada menu ruang, memiliki 4 *method* yaitu *method create*, *read*, *update* dan *delete*. Menu ruang hanya dapat dikelola oleh admin.



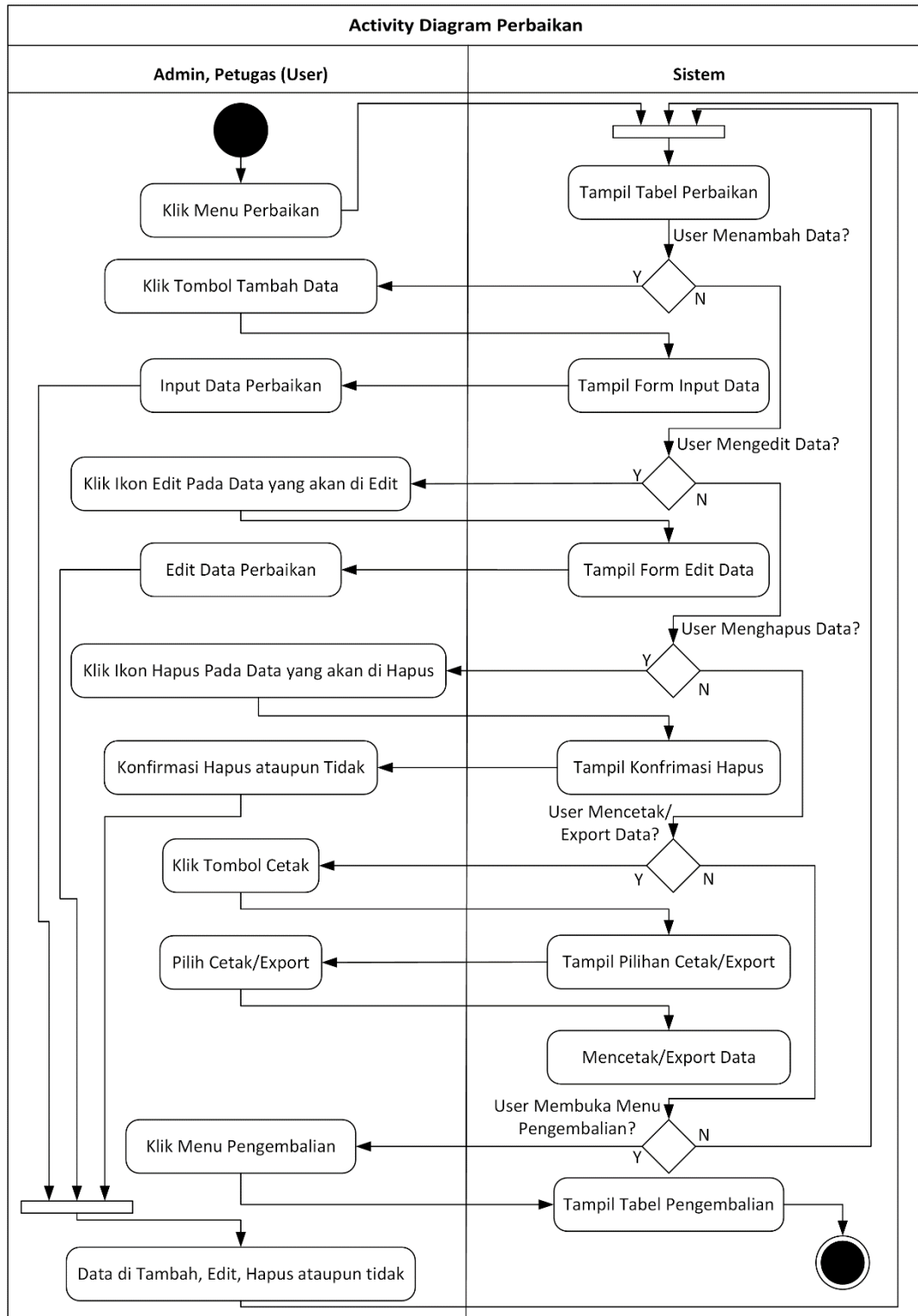
Gambar IV.6 Activity Diagram Merek

Pada activity diagram merek, menjelaskan alur aplikasi jika *user* mengakses menu merek. Menu merek berfungsi untuk mendata merek dari setiap data peralatan yang terdata, yang dimana dengan adanya informasi merek, dapat memudahkan *user* untuk melakukan *filter* data peralatan berdasarkan merek peralatannya sehingga memudahkan dalam mengakses informasi peralatan sesuai dengan yang dibutuhkan. Pada menu merek, memiliki 4 *method* dalam mengelola datanya yaitu *method create, read, update* dan *delete*.



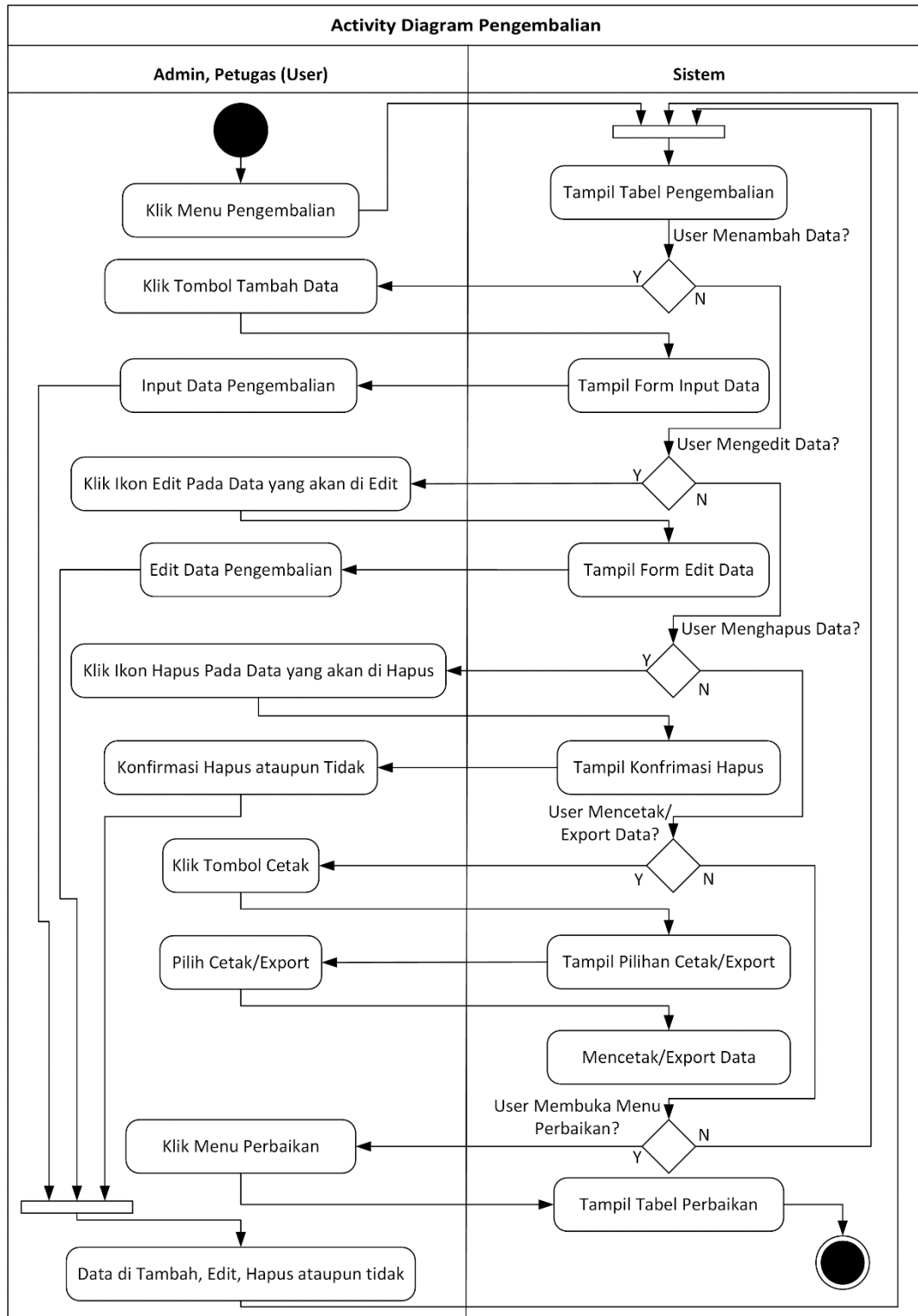
**Gambar IV.7 Activity Diagram Kompetensi**

Pada activity diagram ini, menjelaskan alur aplikasi jika *user* mengakses menu kompetensi. Menu kompetensi berfungsi untuk mendata kompetensi dari laboran yang mengelola laboratorium komputer yang dimana dengan adanya data mengenai kompetensi, *user* dapat memiliki informasi mengenai kompetensi apa yang dimiliki oleh laboran. Selain itu, *user* juga dapat melakukan *filter* data laboran berdasarkan kompetensinya. Pada menu kompetensi, memiliki 4 *method* yaitu *method create*, *read*, *update* dan *delete*. Menu kompetensi hanya dapat dikelola oleh admin.



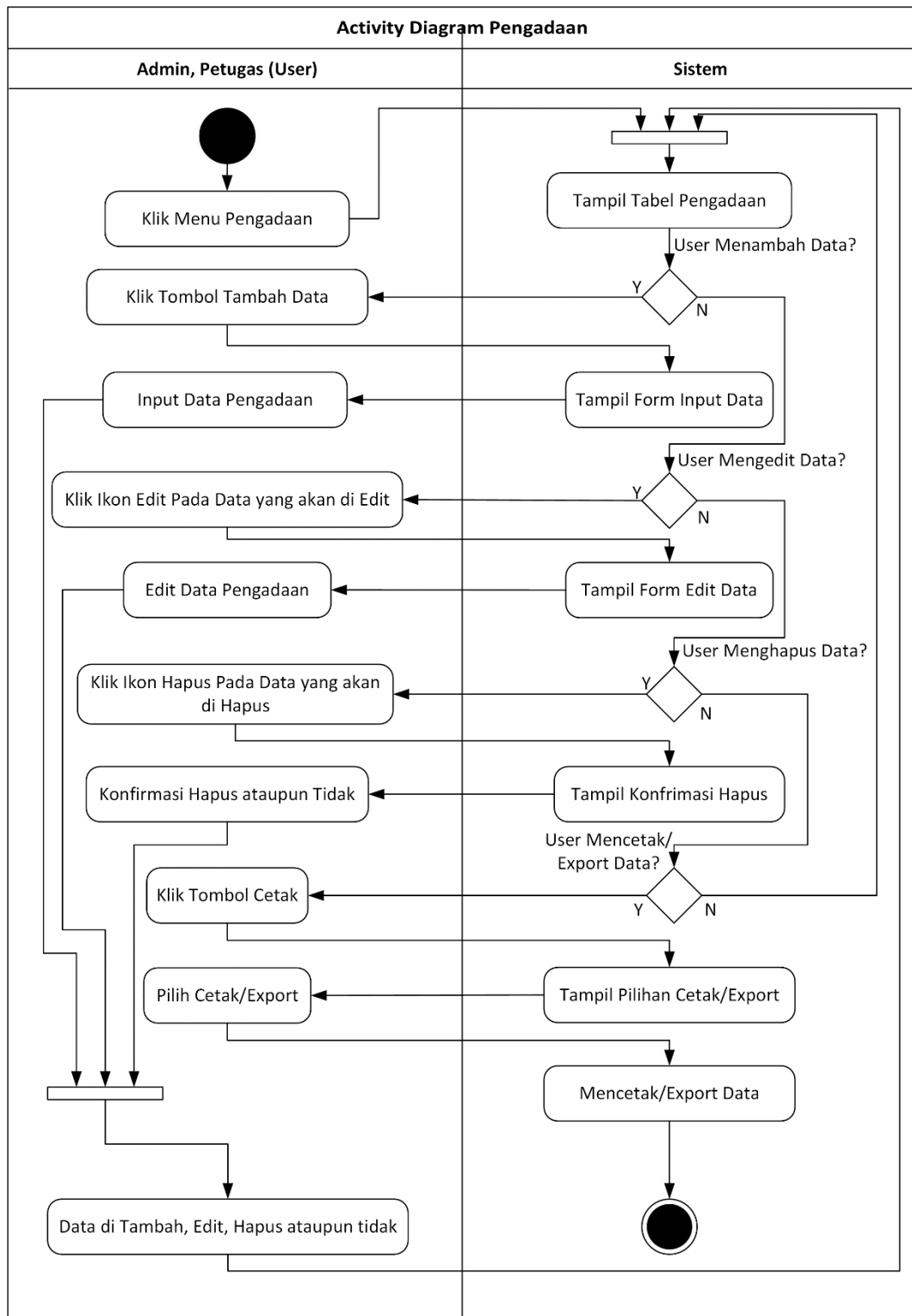
**Gambar IV.8 Activity Diagram Perbaikan**

Pada menu perbaikan, memiliki 4 *method* untuk mengelola datanya yaitu *method create, read, update* dan *delete* serta dapat juga melakukan cetak/export data perbaikan untuk melakukan pelaporan mengenai data perbaikan. Pada menu perbaikan juga terdapat menu pengembalian.



**Gambar IV.9 Activity Diagram Pengembalian**

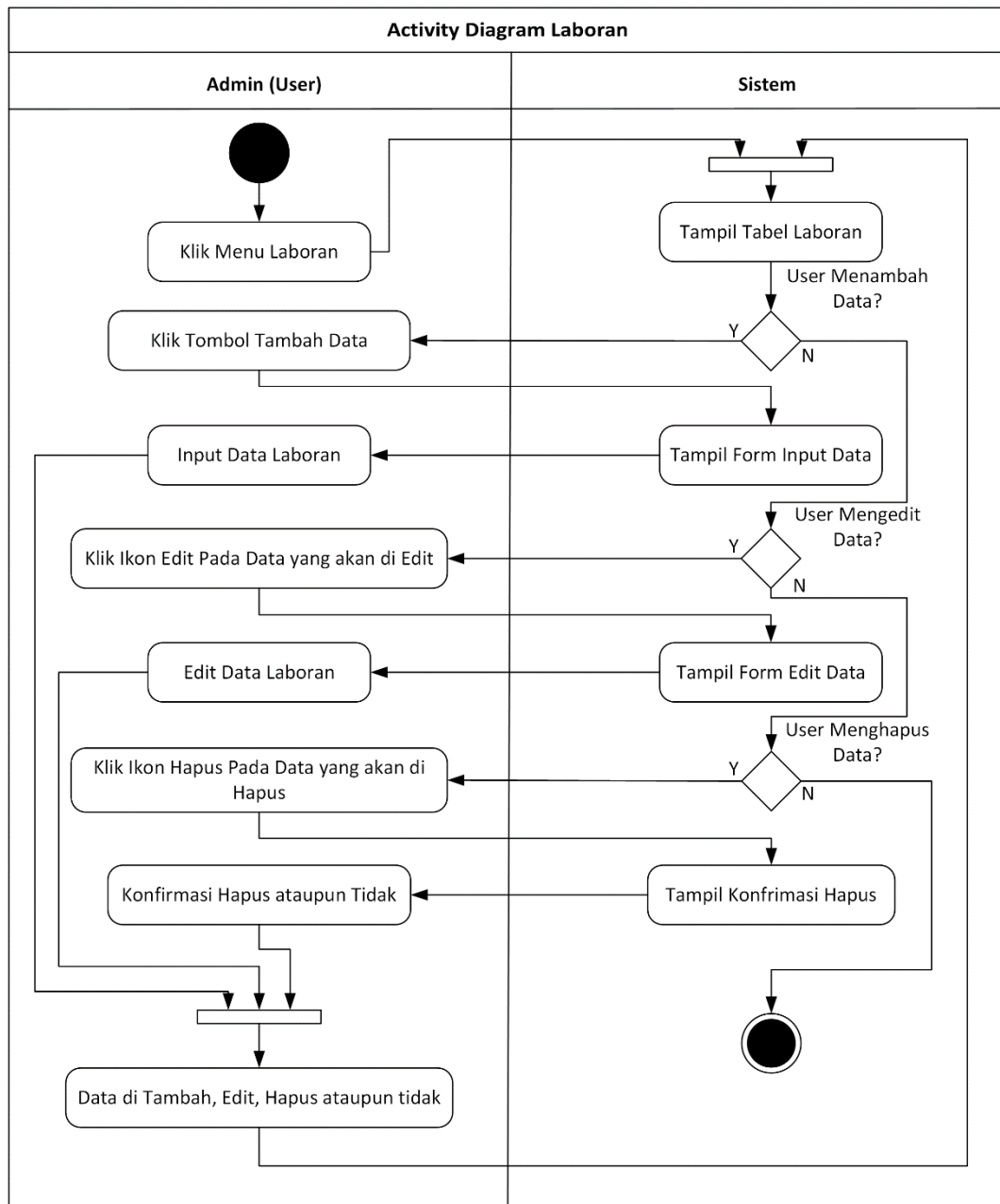
Pada menu pengembalian, memiliki 4 *method* yaitu *create*, *read*, *update* dan *delete* serta dapat juga melakukan cetak/export data pengembalian. Menu pengembalian terletak di menu perbaikan dan tidak bisa diakses secara langsung dari halaman daftar menu aset laboratorium komputer.



**Gambar IV.10 Activity Diagram Pengadaan**

Pada menu pengadaan, memiliki 4 *method* yaitu *method create*, *read*, *update* dan *delete* serta dapat juga melakukan cetak/export data pengadaan untuk melakukan pelaporan mengenai data pengadaan.



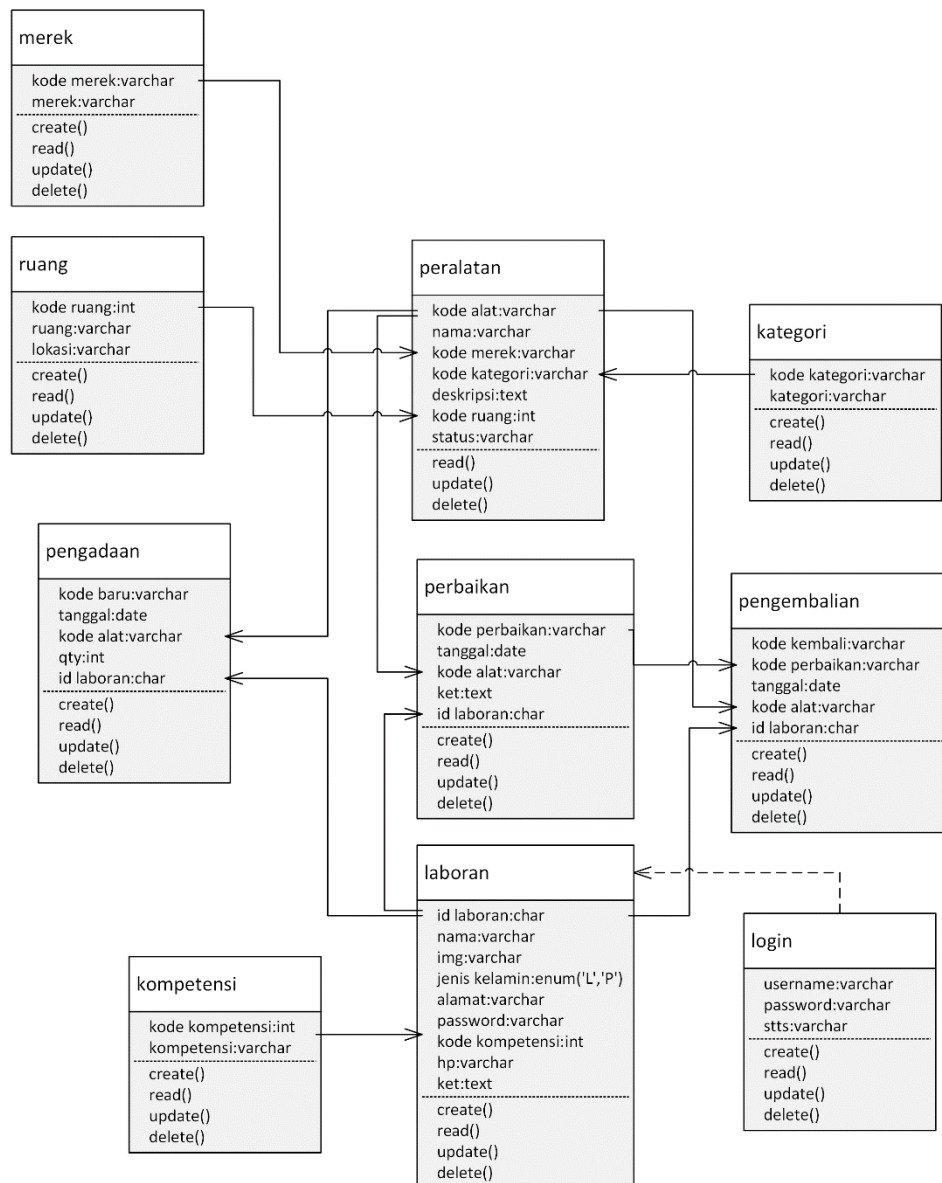


Gambar IV.11 Activity Diagram Laboran (Awaluddin dkk., 2020)

Pada activity diagram ini, menjelaskan alur aplikasi jika *user* mengakses menu laboran. Menu laboran berfungsi untuk mendata laboran yang mengelola laboratorium komputer yang dimana dengan adanya data mengenai laboran, dapat memiliki informasi mengenai siapa saja yang mengelola laboratorium komputer serta memiliki akses ke aplikasi aset laboratorium komputer. Pada menu laboran, memiliki 4 *method* yaitu *method create*, *read*, *update* dan *delete*. Menu laboran hanya dapat dikelola oleh admin.

### c. Class Diagram

Pada perancangan *class diagram* menjelaskan mengenai tabel database yang akan dibuat untuk aplikasi aset laboratorium komputer. Pada tabel ini akan menunjukkan atribut, *method* dan relasi penghubung di tiap tabel yang ada.

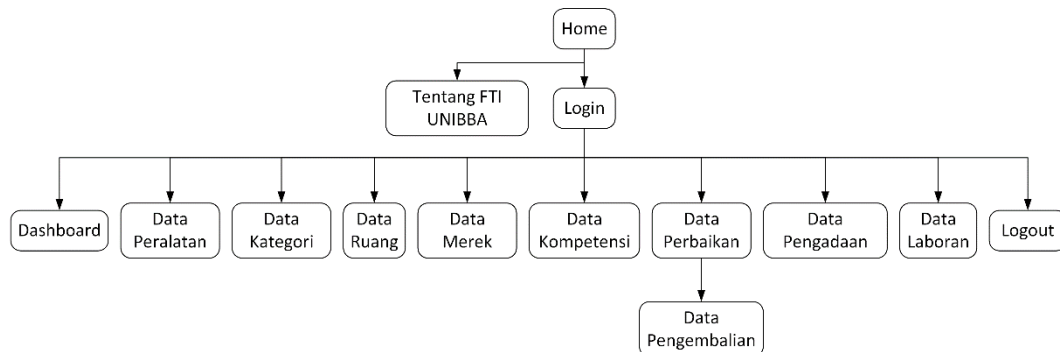


Gambar IV.12 Class Diagram (Nuraeni dkk., 2017)

### d. Perancangan Tampilan Antarmuka

Pada tahapan perancangan tampilan antarmuka aplikasi aset laboratorium komputer, pembuatan rancangannya menggunakan *tools* atau *software* Balsamiq. Perancangan desain tampilan antarmuka ditujukan agar tampilan website yang akan dibuat

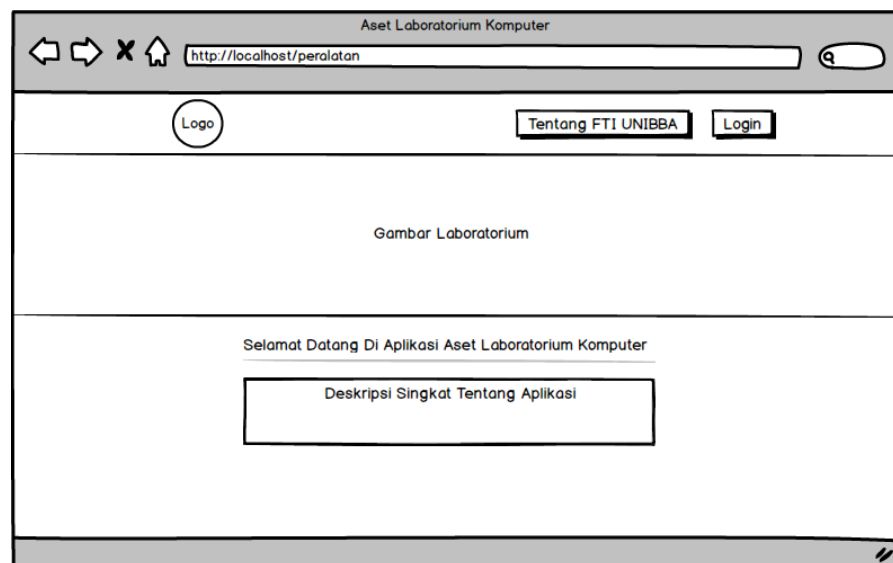
terancang dengan baik. Sebelum membuat rancangan tampilan desain, ada struktur yang mendeskripsikan alur dan jumlah dari desain tampilan antarmuka aplikasi aset laboratorium komputer.



**Gambar IV.13 Struktur Menu (Rahayu S., 2018)**

Untuk rancangan desain tampilan antarmuka aplikasi aset laboratroium komputer yaitu sebagai berikut:

#### 1) Tampilan Home



**Gambar IV.14 Desain Tampilan Antarmuka Home**

Berikut merupakan deskripsi dari rancangan mockup antarmuka home :

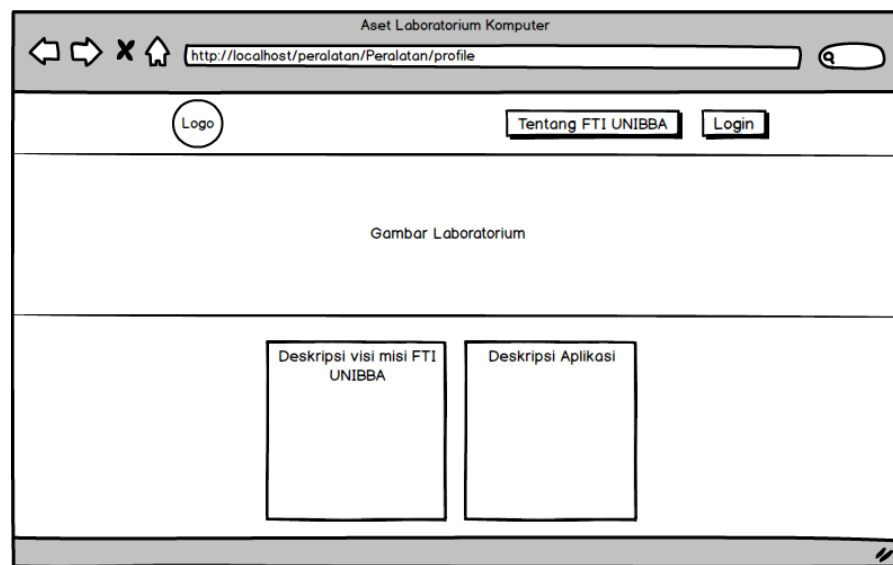
**Tabel IV.37 Deskripsi Mockup Antarmuka Home**

Komponen	Tipe	Fungsi
Logo	Tombol	1) Kembali ke halaman home 2) Logo FTI UNIBBA
Tentang FTI UNIBBA	Tombol	Ke halaman tentang FTI
Login	Tombol	Ke halaman <i>form</i> login

Gambar Laboratorium	Gambar	Gambar laboratorium FTI
Selamat Datang di Aplikasi Aset Laboratorium Komputer	Teks	Teks selamat datang
Deskripsi Singkat Tentang Aplikasi	Teks	Deskripsi singkat tentang aplikasi aset laboratorium komputer FTI UNIBBA

Sumber: (Purwiranda dkk., 2019)

## 2) Tampilan Tentang FTI UNIBBA



**Gambar IV.15 Desain Tampilan Antarmuka Tentang FTI UNIBBA**

Berikut merupakan deskripsi dari rancangan mockup antarmuka Tentang FTI UNIBBA:

**Tabel IV.38 Deskripsi Mockup Antarmuka Tentang FTI UNIBBA**

Komponen	Tipe	Fungsi
Logo	Tombol	Kembali ke halaman home dan logo FTI UNIBBA
Tentang FTI UNIBBA	Tombol	Ke halaman tentang FTI UNIBBA
Login	Tombol	Ke halaman <i>form</i> login
Gambar Laboratorium	Gambar	Gambar laboratorium FTI UNIBBA
Deskripsi visi misi FTI UNIBBA	Teks	Deskripsi tentang visi dan misi FTI
Deskripsi Aplikasi	Teks	Deskripsi lengkap tentang aplikasi aset laboratorium komputer FTI

### 3) Tampilan Login

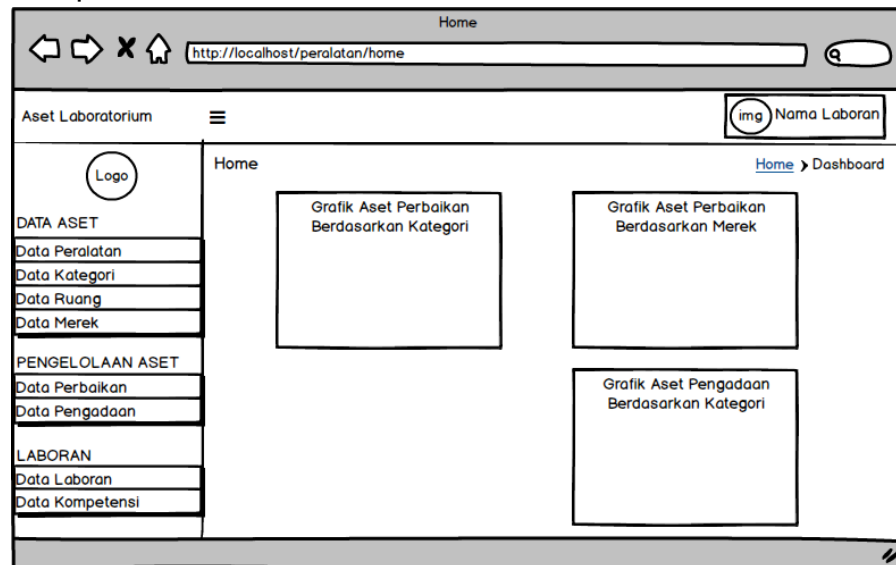
**Gambar IV.16 Desain Tampilan Antarmuka Login**

Berikut merupakan deskripsi dari rancangan mockup antarmuka login:

**Tabel IV.39 Deskripsi Mockup Antarmuka Login**

Komponen	Tipe	Fungsi
Aplikasi Aset Laboratorium Komputer FTI UNIBBA	Teks	Nama aplikasi
Masukkan Akun	Teks	Keterangan untuk memasukkan username dan password akun aplikasi aset laboratorium komputer FTI UNIBBA
Username	Input Teks	Username akun untuk login
Password	Input Teks	Password akun untuk login
Login	Tombol	1) Konfirmasi username dan password akun untuk login 2) Ke halaman menu pengelolaan data aset laboratorium komputer FTI UNIBBA
Kembali Ke Situs	Tombol	Kembali ke halaman home aplikasi aset laboratorium komputer FTI

## 4) Tampilan Dashboard



Gambar IV.17 Desain Tampilan Antarmuka Dashboard

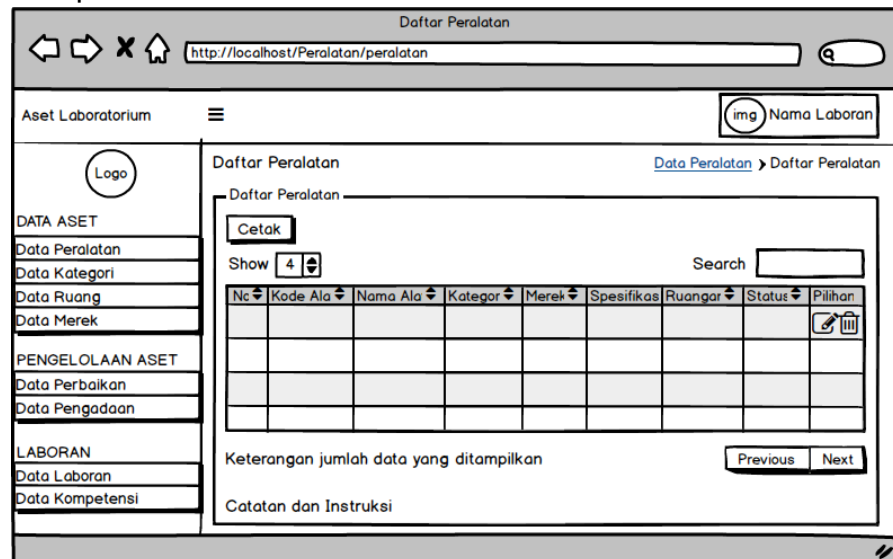
Berikut merupakan deskripsi dari rancangan mockup antarmuka dashboard:

Tabel IV.40 Deskripsi Mockup Antarmuka Dashboard

Komponen	Tipe	Fungsi
Aset Laboratorium FTI	Tombol	Ke halaman dashboard/home
≡	Tombol	Tampilkan/sembunyikan sidebar menu
Nama Laboran	Tombol	1) Informasi akun 2) Logout
Logo	Tombol	Membuka/memperbesar gambar logo
DATA ASET	Teks	Kategori menu data yang bersangkutan dengan data aset
PENGELOLAAN ASET	Teks	Kategori menu data yang bersangkutan dengan pengelolaan data aset
LABORAN	Teks	Kategori menu data yang bersangkutan dengan data laboran
Data Peralatan	Tombol	Mengelola data aset
Data Kategori	Tombol	Mengelola data kategori aset
Data Ruang	Tombol	Mengelola data ruangan laboratorium
Data Merek	Tombol	Mengelola data merek aset

Data Perbaikan	Tombol	Mengelola data perbaikan dan pengembalian aset
Data Pengadaan	Tombol	Mengelola data pengadaan aset
Data Laboran	Tombol	Mengelola data akun/laboran
Data Kompetensi	Tombol	Mengelola data kompetensi
Home	Teks	Keterangan menu data yang sedang dibuka/diakses
Home > Dashboard	Tombol	Kembali ke tampilan awal menu yang sedang di akses
Grafik Aset Perbaikan Berdasarkan Kategori	Grafik	Menampilkan data Aset Perbaikan Berdasarkan Kategori dalam bentuk grafik
Grafik Aset Perbaikan Berdasarkan Merek	Grafik	Menampilkan data Aset Perbaikan Berdasarkan Merek dalam bentuk grafik
Grafik Aset Pengadaan Berdasarkan Kategori	Grafik	Menampilkan data Aset Pengadaan Berdasarkan Kategori dalam bentuk grafik

#### 5) Tampilan Data Peralatan





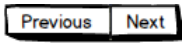


Gambar IV.18 Desain Tampilan Antarmuka Data Peralatan

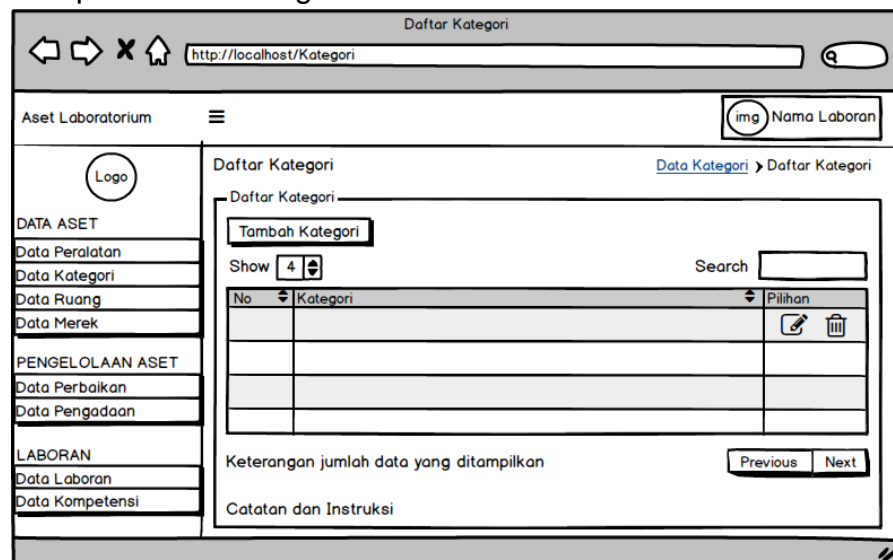
Berikut merupakan deskripsi dari rancangan mockup antarmuka data peralatan:

Tabel IV.41 Deskripsi Mockup Antarmuka Data Peralatan

Komponen	Tipe	Fungsi
Daftar Peralatan	Kotak	Kotak berisi data peralatan

Cetak	Tombol	Ke halaman cetak/export data peralatan
Show 	Tombol/Input Angka	Menentukan banyak data yang ditampilkan di tabel data peralatan
Search <input type="text"/>	Input Teks	Pencarian data yang ada di tabel data peralatan
	Tombol	Mengurutkan data yang ada di tabel data peralatan
	Tombol	Edit data peralatan
	Tombol	Hapus data peralatan
	Tombol	Berganti halaman tabel data peralatan
Keterangan jumlah data yang ditampilkan	Teks	Keterangan banyak halaman tabel yang memuat data peralatan
Catatan dan Instruksi	Teks	Keterangan cara untuk mengedit dan menghapus data peralatan

## 6) Tampilan Data Kategori



Gambar IV.19 Desain Tampilan Antarmuka Data Kategori

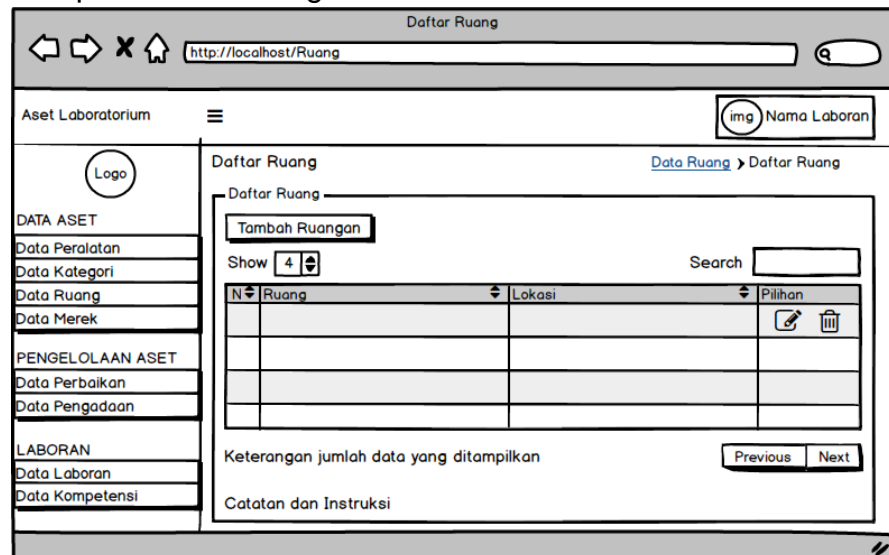
Berikut merupakan deskripsi dari rancangan mockup antarmuka data kategori:



Tabel IV.42 Deskripsi Mockup Antarmuka Data Kategori

Komponen	Tipe	Fungsi
Tambah Kategori	Tombol	Ke halaman <i>form</i> tambah data kategori

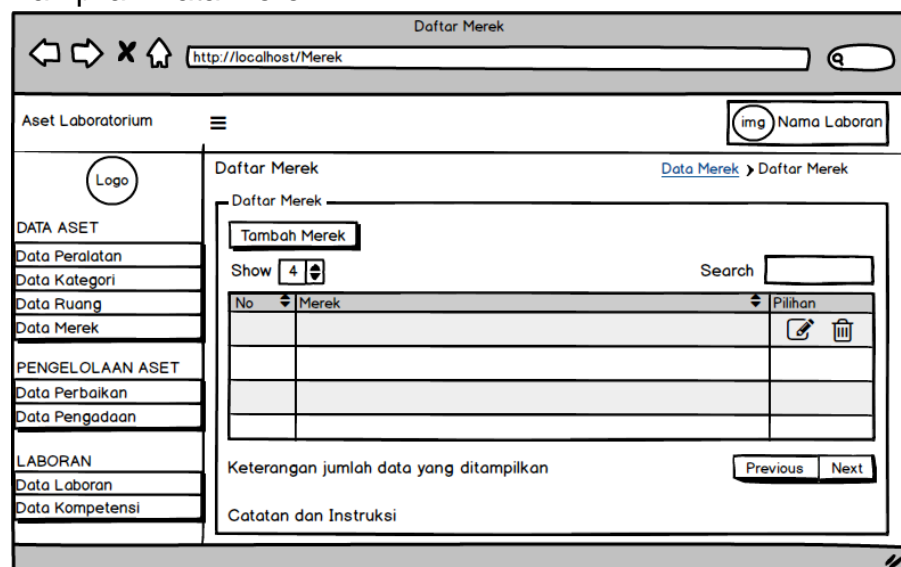
## 7) Tampilan Data Ruang



Gambar IV.20 Desain Tampilan Antarmuka Data Ruang

Deskripsi dari rancangan mockup antarmuka data kategori sama saja seperti deskripsi dari rancangan antarmuka sebelumnya. Yang membedakan hanyalah data yang dikelolanya.

## 8) Tampilan Data Merek

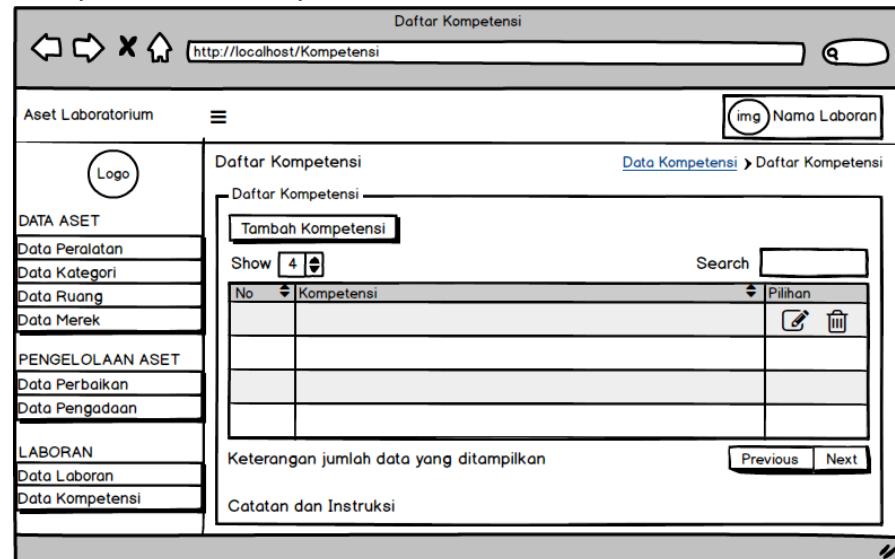


Gambar IV.21 Desain Tampilan Antarmuka Data Merek

Deskripsi dari rancangan mockup antarmuka data merek sama saja seperti deskripsi dari rancangan antarmuka

sebelumnya. Yang membedakan hanyalah data yang dikelolanya.

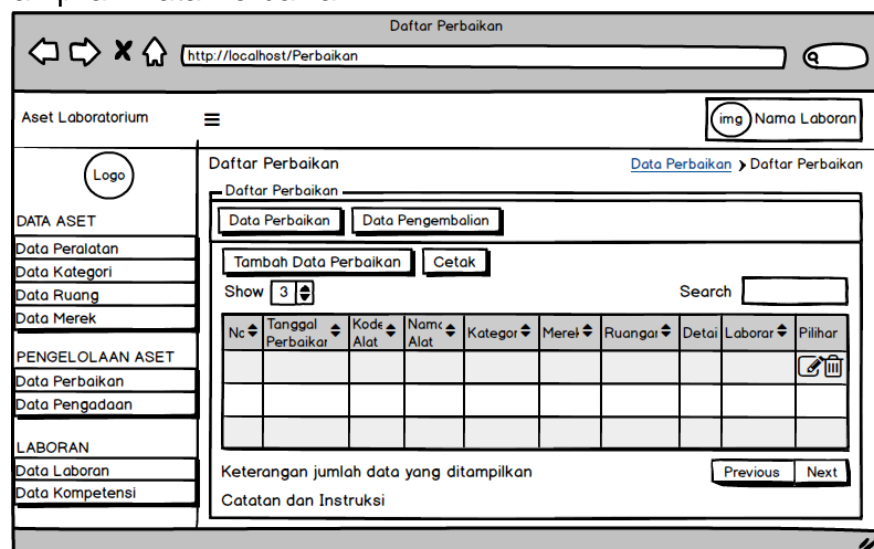
#### 9) Tampilan Data Kompetensi



Gambar IV.22 Desain Tampilan Antarmuka Data Kompetensi

Deskripsi dari rancangan mockup antarmuka data kompetensi sama saja seperti deskripsi dari rancangan antarmuka sebelumnya. Yang membedakan hanyalah data yang dikelolanya.

#### 10) Tampilan Data Perbaikan



Gambar IV.23 Desain Tampilan Antarmuka Data Perbaikan

Berikut merupakan deskripsi dari rancangan mockup antarmuka data perbaikan:

Tabel IV.43 Deskripsi Mockup Antarmuka Data Perbaikan

Komponen	Tipe	Fungsi
Data Perbaikan	Tombol	Ke halaman menu data perbaikan aset
Data pengembalian	Tombol	Ke halaman menu data pengembalian aset yang telah diperbaiki

## 11) Tampilan Data Pengembalian

Gambar IV.24 Desain Tampilan Antarmuka Data Pengembalian

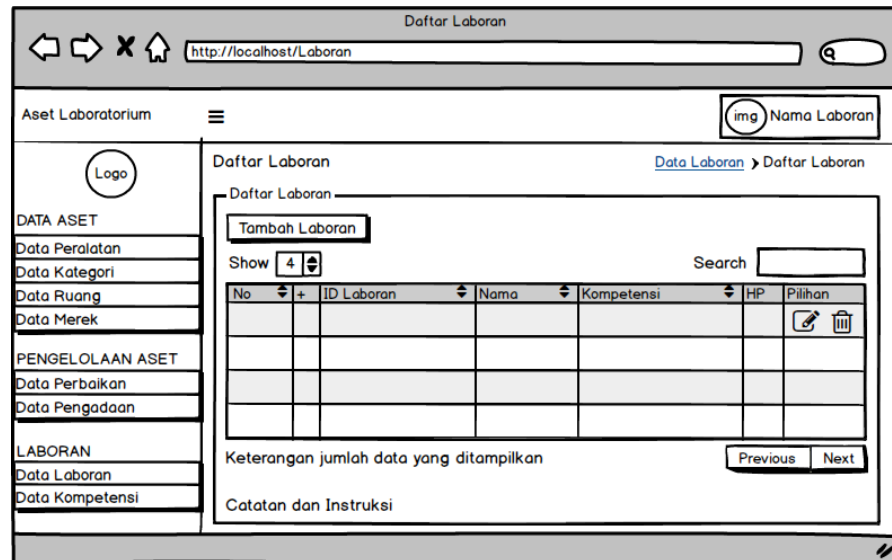
Deskripsi dari rancangan mockup antarmuka data pengembalian sama saja seperti deskripsi dari rancangan antarmuka sebelumnya. Yang membedakan hanyalah data yang dikelolanya.

## 12) Tampilan Data Pengadaan

Gambar IV.25 Desain Tampilan Antarmuka Data Pengadaan

Deskripsi dari rancangan mockup antarmuka data pengadaan sama saja seperti deskripsi dari rancangan antarmuka sebelumnya. Yang membedakan hanyalah data yang dikelolanya.

### 13) Tampilan Data Laboran



Gambar IV.26 Desain Tampilan Antarmuka Data Laboran

Deskripsi dari rancangan mockup antarmuka data Laboran sama saja seperti deskripsi dari rancangan antarmuka sebelumnya. Yang membedakan hanyalah data yang dikelolanya.

#### e. Perancangan Basis Data

Setelah melakukan rancangan untuk desain antarmuka, selanjutnya rancangan untuk basis data aplikasi aset laboratorium komputer. Pada perancangan basis data terdiri dari *field*, Tipe, *size*, indeks dan Deskripsi. Untuk indeks terdiri dari *Primary Key* (PK) dan *Foreign Key* (FK).

Tabel IV.44 Perancangan Data Peralatan

Field	Tipe	Size	Indeks	Deskripsi
kode_alat	VarChar	10	PK	Kode alat
nama	VarChar	30		Nama Alat
kode_merek	VarChar	10	FK	Kode Merek
kode_kategori	VarChar	10	FK	Kode Kategori

deskripsi	text			Deskripsi Spesifikasi
kode_ruang	int	11	FK	Kode ruangan
status	VarChar	20		kondisi alat

Tabel IV.45 Perancangan Data Merek

Field	Tipe	Size	Indeks	Deskripsi
kode_merek	VarChar	10	PK	Kode merek
merek	VarChar	20		Nama merek

Tabel IV.46 Perancangan Data Ruang

Field	Tipe	Size	Indeks	Deskripsi
kode_ruang	int	11	PK	Kode ruangan
ruang	VarChar	20		Nama ruangan
lokasi	VarChar	20		Lokasi ruangan

Tabel IV.47 Perancangan Data Kategori

Field	Tipe	Size	Indeks	Deskripsi
kode_kategori	VarChar	10	PK	Kode Kategori
kategori	VarChar	20		Nama kategori

Tabel IV.48 Perancangan Data Perbaikan

Field	Tipe	Size	Indeks	Deskripsi
kode_perbaikan	int	11	PK	Kode perbaikan
tanggal	date			Tanggal perbaikan
kode_alat	VarChar	10	FK	Kode alat
ket	text			Informasi perbaikan
id_laboran	char	10	FK	Id laboran

Tabel IV.49 Perancangan Data Pengembalian

Field	Tipe	Size	Indeks	Deskripsi
kode_kembali	int	11	PK	Kode pengembalian
kode_perbaikan	int	11	FK	Kode perbaikan

tanggal	date			Tanggal kembali
kode_alat	VarChar	10	FK	Kode alat
id_laboran	Char	10	FK	Id laboran

Tabel IV.50 Perancangan Data Pengadaan

Field	Tipe	Size	Indeks	Deskripsi
kode_baru	int	11	PK	Kode pengadaan
tanggal	date			Tanggal pengadaan
kode_alat	VarChar	10	FK	Kode alat
qty	int	10		Kuantitas pengadaan alat
id_laboran	char	10	FK	Id laboran

Tabel IV.51 Perancangan Data Laboran

Field	Tipe	Size	Indeks	Deskripsi
id_laboran	Char	10	PK	Id laboran
nama	VarChar	20		Nama Laboran
img	VarChar	50		Foto laboran
jenis_kelamin	Enum {L, P}			Jenis kelamin
alamat	text			Alamat laboran
password	VarChar	50		Password <i>user</i>
kode_kompetensi	VarChar	10	FK	Kode kompetensi
hp	VarChar	15		Nomor HP laboran
ket	text			Keterangan laboran

Tabel IV.52 Perancangan Data Kompetensi

Field	Tipe	Size	Indeks	Deskripsi
kode_kompetensi	VarChar	10	PK	Kode Kompetensi
kompetensi	VarChar	20		Nama kompetensi

Tabel IV.53 Perancangan Data Login

Field	Tipe	Size	Indeks	Deskripsi
username	VarChar	10	PK	Username
password	VarChar	35		Password pengguna
stts	VarChar	10		Jenis pengguna

### IV.2.3 Pelaporan Hasil Kerja Praktek

Proses pelaporan hasil kerja praktek dilakukan pada tahap akhir kerja praktek di Laboratorium Fakultas Teknologi Informasi Universitas Bale Bandung. Pelaporan hasil kerja praktek dilakukan dengan pembuatan laporan kerja praktek yang nantinya pelaporan yang telah dibuat akan dipresentasikan kepada penguji kampus.


## IV.3 Pencapaian Hasil

Adapun hasil yang telah dicapai setelah melaksanakan kerja praktek selama dua bulan di Laboratorium komputer Fakultas Teknologi Informasi Universitas Bale Bandung yaitu terciptanya aplikasi aset laboratorium komputer beserta laporannya.

### 1. Database MySQL

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	kode_alat 	varchar(10)	latin1_swedish_ci		Tidak	Tidak ada		
2	nama	varchar(30)	latin1_swedish_ci		Tidak	Tidak ada		
3	kode_merek 	varchar(10)	latin1_swedish_ci		Tidak	Tidak ada		
4	kode_kategori 	varchar(10)	latin1_swedish_ci		Tidak	Tidak ada		
5	deskripsi	text	latin1_swedish_ci		Tidak	Tidak ada		
6	kode_ruang 	int(10)			Tidak	Tidak ada		
7	status	varchar(20)	latin1_swedish_ci		Tidak	Tidak ada		

Gambar IV.27 Database Peralatan

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	username 	varchar(10)	latin1_swedish_ci		Tidak	Tidak ada		
2	password	varchar(35)	latin1_swedish_ci		Tidak	Tidak ada		
3	stts	varchar(10)	latin1_swedish_ci		Tidak	Tidak ada		

Gambar IV.28 Database Login

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	kode_baru 🔑	int(11)			Tidak	Tidak ada		
2	tanggal	date			Tidak	Tidak ada		
3	kode_alat 🔑	varchar(10)	latin1_swedish_ci		Tidak	Tidak ada		
4	qty	int(10)			Tidak	Tidak ada		
5	id_laboran 🔑	char(10)	latin1_swedish_ci		Tidak	Tidak ada		

Gambar IV.29 Database Pengadaan

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	kode_kembali 🔑	int(11)			Tidak	Tidak ada		
2	kode_perbaikan 🔑	int(11)			Tidak	Tidak ada		
3	tanggal	date			Tidak	Tidak ada		
4	kode_alat 🔑	varchar(10)	latin1_swedish_ci		Tidak	Tidak ada		
5	id_laboran 🔑	char(10)	latin1_swedish_ci		Tidak	Tidak ada		

Gambar IV.30 Database Pengembalian

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	kode_ruang 🔑	int(11)			Tidak	Tidak ada		AUTO_INCREMENT
2	ruang	varchar(20)	latin1_swedish_ci		Tidak	Tidak ada		
3	lokasi	varchar(20)	latin1_swedish_ci		Tidak	Tidak ada		

Gambar IV.31 Database Ruang

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	kode_perbaikan 🔑	int(11)			Tidak	Tidak ada		AUTO_INCREMENT
2	tanggal	date			Tidak	Tidak ada		
3	kode_alat 🔑	varchar(10)	latin1_swedish_ci		Tidak	Tidak ada		
4	ket	text	latin1_swedish_ci		Tidak	Tidak ada		
5	id_laboran 🔑	char(10)	latin1_swedish_ci		Tidak	Tidak ada		

Gambar IV.32 Database Perbaikan

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	id_laboran 🔑	char(10)	latin1_swedish_ci		Tidak	Tidak ada		
2	nama	varchar(20)	latin1_swedish_ci		Tidak	Tidak ada		
3	img	varchar(50)	latin1_swedish_ci		Tidak	Tidak ada		
4	jenis_kelamin	enum('L', 'P')	latin1_swedish_ci		Tidak	Tidak ada		
5	alamat	text	latin1_swedish_ci		Tidak	Tidak ada		
6	password	varchar(50)	latin1_swedish_ci		Tidak	Tidak ada		
7	kode_kompetensi 🔑	varchar(10)	latin1_swedish_ci		Tidak	Tidak ada		
8	hp	varchar(15)	latin1_swedish_ci		Tidak	Tidak ada		
9	ket	text	latin1_swedish_ci		Tidak	Tidak ada		

Gambar IV.33 Database Laboran



#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	kode_merek	varchar(10)	latin1_swedish_ci		Tidak	Tidak ada		
2	merek	varchar(20)	latin1_swedish_ci		Tidak	Tidak ada		

Gambar IV.34 Database Merek

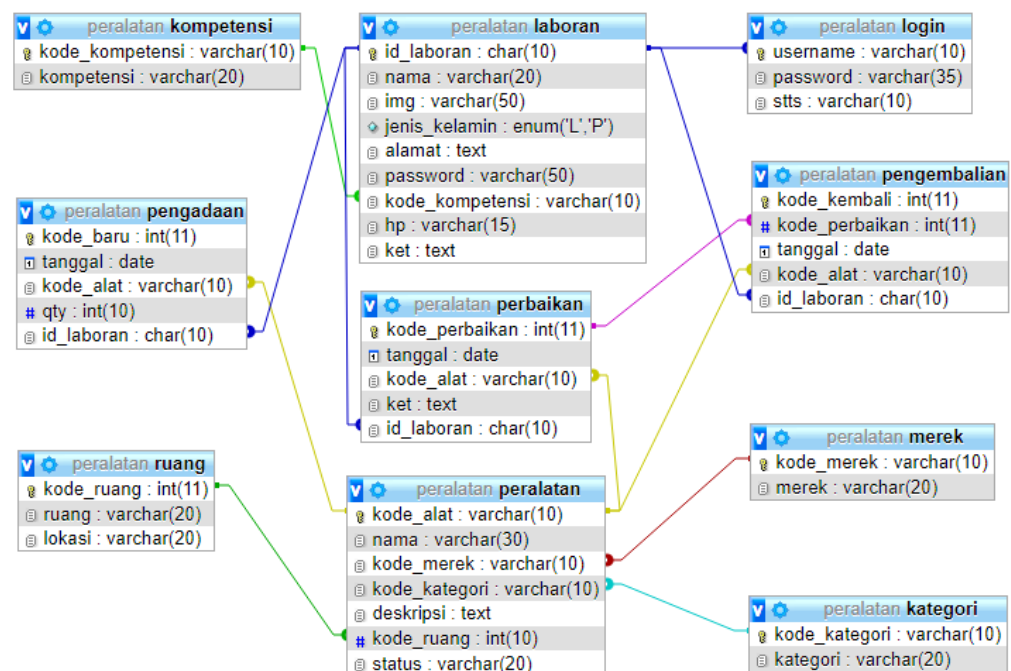
#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	kode_kategori	varchar(10)	latin1_swedish_ci		Tidak	Tidak ada		
2	kategori	varchar(20)	latin1_swedish_ci		Tidak	Tidak ada		

Gambar IV.35 Database Kategori

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra
1	kode_kompetensi	varchar(10)	latin1_swedish_ci		Tidak	Tidak ada		
2	kompetensi	varchar(20)	latin1_swedish_ci		Tidak	Tidak ada		

Gambar IV.36 Database Kompetensi

## 2. Relasi Database



Gambar IV.37 Relasi Database Aplikasi Aset Laboratorium Komputer

Seperti yang terlihat pada Gambar IV.37, data laboran, data perbaikan, data pengembalian, dan data peralatan merupakan bagian data inti pada aplikasi aset laboratorium komputer yang memiliki banyak relasi dengan data-data lainnya.

### 3. Tampilan Aplikasi



Selamat Datang Di Aplikasi Aset Laboratorium Komputer

Aplikasi Aset LAB KOMP FTI merupakan aplikasi yang digunakan untuk melakukan Pengelolaan Data Aset yang ada di Laboratorium Komputer FTI.

**Gambar IV.38 Tampilan Home**



Profil Fakultas Teknologi Informasi

#### Visi

Menjadi Fakultas Teknologi Informasi (FTI) yang unggul dibidang Teknik Informatika dan Sistem Informasi, yang mampu

#### Tentang Aplikasi Aset Laboratorium Komputer FTI UNIBBA

Fakultas Teknologi Informasi

**Gambar IV.39 Tampilan Tentang FTI UNIBBA**

Gambar IV.40 Tampilan Login

Gambar IV.41 Tampilan Dashboard

Gambar IV.42 Tampilan Kategori

**Aset Laboratorium**

**Daftar Ruang**

+Tambah Ruangan

Show 10 entries Search:

No	Ruang	Lokasi	Pilihan
No data available in table			
No	Ruang	Lokasi	Pilihan

Showing 0 to 0 of 0 entries Previous Next

Menampilkan daftar Ruangan, untuk mengedit dan menghapus data ruangan klik tombol pada kolom pilihan.

Gambar IV.43 Tampilan Ruang

**Aset Laboratorium**

**Daftar Merek**

+Tambah Merek

Show 10 entries Search:

No	Merek	Pilihan
No data available in table		
No	Merek	Pilihan

Showing 0 to 0 of 0 entries Previous Next

Menampilkan daftar Merek, mengedit dan menghapus Merek klik tombol pada kolom pilihan.

Gambar IV.44 Tampilan Merek

**Aset Laboratorium**

**Daftar Kompetensi**

+Tambah Kompetensi

Show 10 entries Search:

No	Kompetensi	Pilihan
No data available in table		
No	Kompetensi	Pilihan

Showing 0 to 0 of 0 entries Previous Next

Menampilkan daftar kompetensi, mengedit dan menghapus Kompetensi klik tombol pada kolom pilihan.

Gambar IV.45 Tampilan Kompetensi

## **BAB V**

### **PENUTUP**

#### **V.1 Kesimpulan dan Saran Mengenai Pelaksanaan**

Berdasarkan uraian pada bab-bab sebelumnya maka dapat disimpulkan:

##### **V.1.1 Kesimpulan Pelaksanaan Kerja Praktek**

1. Mahasiswa dapat mengaplikasikan ilmu yang diperoleh selama perkuliahan untuk menyelesaikan permasalahan di dunia kerja.
2. Mahasiswa dapat mengetahui ilmu dan keterampilan yang dibutuhkan untuk memasuki dunia kerja, seperti:
  - Keterampilan berkomunikasi dan bekerja sama dengan orang lain.
  - Ilmu dasar mengenai bidang spesifik di dunia kerja.
  - Keterampilan menganalisis permasalahan untuk dicari solusinya.
  - Ilmu pengetahuan umum.
  - Keterampilan mempelajari hal baru dalam waktu relatif singkat.
3. Mahasiswa menyadari pentingnya etos kerja yang baik, disiplin, dan tanggung jawab dalam menyelesaikan suatu pekerjaan.
4. Kerja praktek dapat melatih mahasiswa untuk bekerja sama dalam suatu tim, baik antar peserta kerja praktek maupun dengan karyawan lain di Laboratorium FTI UNIBBA.
5. Mahasiswa memperoleh tambahan ilmu yang tidak diperoleh di proses perkuliahan. Pada kerja praktek yang dilakukan di Laboratorium FTI UNIBBA, mahasiswa mendapatkan pengetahuan tambahan mengenai:
  - Cakupan pekerjaan di ruang laboratorium komputer FTI UNIBBA pada bagian pengelolaan aset, seperti pengelolaan data-data mengenai aset laboratorium komputer FTI UNIBBA.
  - Perancangan antarmuka aplikasi berbasis web yang *user-friendly* dalam waktu singkat.

##### **V.1.2 Saran Pelaksanaan Kerja Praktek**

Adapun saran mengenai pelaksanaan kerja praktek antara lain:

1. Perlu adanya kemampuan mahasiswa untuk menggabungkan seluruh ilmu yang pernah didapat di perkuliahan dalam proses pembangunan perangkat lunak.
2. Perlu adanya bimbingan secara lebih intensif bagi mahasiswa kerja praktek.

## **V.2 Kesimpulan dan Saran Mengenai Aplikasi Aset Laboratorium Komputer**

Berikut merupakan kesimpulan dan saran mengenai aplikasi aset laboratorium komputer untuk laboratorium FTI UNIBBA:

### **V.2.1 Kesimpulan Aplikasi Aset Laboratorium Komputer**

Setelah melalui proses pembuatan aplikasi aset laboratorium komputer, kesimpulan yang didapat sebagai berikut:

1. Aplikasi aset laboratorium komputer merupakan hasil dari pelaksanaan kerja praktek selama di laboratorium komputer FTI UNIBBA.
2. Dengan memanfaatkan aplikasi yang telah dibuat, telah membuka peluang untuk mengelola data aset laboratorium komputer lebih cepat, akurat dan tepat.
3. Dengan memanfaatkan aplikasi yang telah dibuat, dapat membantu memudahkan untuk melakukan pembuatan proposal usulan perbaikan dan pengadaan aset laboratorium ke pihak yayasan maupun rektorat Universitas Bale Bandung.
4. Pengembangan aplikasi aset laboratorium komputer dilakukan sesuai dengan *software development lifecycle* yang terarah dengan digunakannya metodologi *waterfall*. Metodologi *waterfall* memungkinkan pengembang perangkat lunak untuk mengembangkan perangkat lunak yang dibuat dilakukan secara berurutan pada setiap tahapan-tahapannya.

### **V.2.2 Saran Aplikasi Aset Laboratorium Komputer**

Berdasarkan hasil pengembangan aplikasi aset laboratorium komputer, saran yang diajukan adalah sebagai berikut:

1. Perlu adanya optimasi secara lebih lanjut, misalnya seperti perbaikan fitur, dan peningkatan antarmuka (UI dan UX).
2. Perlu melengkapi fungsionalitas aplikasi pada bagian menu dashboard, peralatan, perbaikan, pengembalian serta menu pengadaan aset.
3. Perlu menyesuaikan penamaan setiap variabel dan juga URL yang digunakan pada aplikasi.
4. Perlu adanya penambahan fitur pada aplikasi aset laboratorium komputer agar fungsinya semakin lengkap.

## DAFTAR PUSTAKA

- Alkodri, A. A. (2018). Rancang Bangun Sistem Informasi Manajemen Laboratorium Komputer Berbasis Website: Studi Kasus STMIK ATMA LUHUR. Konferensi Nasional Sistem Informasi (KNSI) 2018.
- Awaluddin MI, Arifin RW, Setiyadi D. 2020. Implementasi Framework Laravel pada Sistem Informasi Pengelolaan Aset Laboratorium Komputer. Bina Insani ICT Journal. 7(2): 187-197.
- Fitriani, B., Angraini, T., & Putra, Y. H. G. (2018, July). Pemodelan Use Case Diagram Sistem Informasi Inventaris Laboratorium Teknik Mesin. In Proceeding Seminar Nasional Sistem Informasi dan Teknologi Informasi (Vol. 1, No. 1, pp. 626-631).
- Gustopa. (2021, 12 12). Kelebihan dan Kekurangan VS Code. Retrieved from plimbi.com: <https://www.plimbi.com/article/176010/kelebihan-dan-kekurangan-vs-code>
- Idcloudhost, T. (2021, 12 12). Mengetahui Fungsi, Kelebihan dan Kekurangan Framework CodeIgniter. Retrieved from idcloudhost.com: <https://idcloudhost.com/panduan/mengetahui-fungsi-kelebihan-dan-kekurangan-codeigniter/>
- Kusnadi, E., Papuas, A., & Tindi, A. (2018). Aplikasi Peminjaman Alat Praktikum Di Jurusan Teknik Komputer Dan Komunikasi Menggunakan Barcode. Jurnal Ilmiah Behongang, 1(1), 1-6.
- MOKOGINTA, N. (2012). sistem informasi manajemen laboratorium komputer pada jurusan teknik informatika. Skripsi, 1(531307073).
- Nuraeni, F., Mufizar, T., & Suparman, N. G. H. (2017, October). Sistem Informasi Inventori Barang Laboratorium Komputer Berbasis Android di STMIK Tasikmalaya. In Seminar Nasional Informatika (SNIf) (Vol. 1, No. 1, pp. 339-343).
- Purwiranda, P. P., Gumilang, S. F., & Witarsyah, D. (2019). Pembangunan Dashboard Market Place Borongajayuk Menggunakan Metode Iterative Incremental. eProceedings of Engineering, 6(2).



- Rahayu, S. (2018). Perancangan Aplikasi Barbershop Online. *Jurnal Algoritma*, 15(1), 29-36.
- Saputra, Y., M. (2021, 12 12). Pengertian Bootstrap. Retrieved from ruangguru.co: <https://ruangguru.co/pengertian-bootstrap/>
- Setiawan, W. N., & Nada, N. Q. (2020). Perancangan Sistem Informasi Inventory Laboratorium Komputer Universitas Pgri Semarang Berbasis Web. In *Science and Engineering National Seminar* (Vol. 5, No. 1, pp. 818-824).
- THenny, T. H. P. T. (2018). Aplikasi Sistem Inventaris Laboratorium Komputer. *Infokam*, 14(1).
- Wicaksono, T. A., & Fauzi, M. A. (2015). System Inventory Control Pada Laboratorium Komputer Smk Muhammadiyah Kajen Berbasis Web Dengan Framework Codeigniter. *Jurnal Surya Informatika: Membangun Informasi dan Profesionalisme*, 1(1), 25-29.
- Yanuarti, E. (2017). Desain Aplikasi Pengelolaan Laboratorium Komputer. *Jurnal Sisfokom (Sistem Informasi Dan Komputer)*, 6(1), 60-66.

## **LAMPIRAN A.**

### **TOR (TERM OF REFERENCE)**

Sebelum melaksanakan kerja praktek, penulis melakukan beberapa metode penelitian yaitu observasi, wawancara, dan studi pustaka. Setelah itu, mengamati dan mempelajari lokasi kerja praktek yang telah ditentukan dan disetujui oleh instansi tempat kerja praktek. Setelah kepala instansi menyetujui penulis melakukan kerja praktek tersebut, kepala instansi menjelaskan bahwa dari instansi memiliki tugas yang harus dikerjakan oleh penulis selama kerja praktek yaitu mempermudah pengelolaan dan akses informasi data aset laboratorium komputer serta mempermudah dalam pengajuan proposal aset laboratorium komputer ke pihak Yayasan dan Rektorat Universitas Bale Bandung.

Bandung, .....

Disetujui Oleh:

Peserta Kerja Praktek

Pembimbing Lapangan

Andika Dwiputra

NIM: 301180004

Mochamad Ridwan, S.T.

NIK: 04104808007