



**RANCANG BANGUN APLIKASI PENCATATAN KEUANGAN BERBASIS
WEBSITE PADA PESANTREN MAHASISWA MAHAD THAYBAH**

KERJA PRAKTIK

**Program Studi
S1 Sistem Informasi**

Oleh :

**MOCHAMMAD APRILANDI NANU DWI FIRMANSYAH
17410100053**

**FAKULTAS TEKNOLOGI DAN INFORMATIKA
UNIVERSITAS DINAMIKA
2020**

**RANCANG BANGUN APLIKASI PENCATATAN KEUANGAN
BERBASIS *WEBSITE* PADA PESANTREN MAHASISWA MAHAD
THAYBAH**

Diajukan sebagian salah satu syarat untuk menyelesaikan
Program Sarjana



Disusun Oleh:

**Nama : MOCHAMMAD APRILANDI
NANU DWI FIRMANSYAH**

NIM : 17410100053

Program : S1 (Strata Satu)

Jurusan : Sistem Informasi

**FAKULTAS TEKNOLOGI DAN INFORMATIKA
UNIVERSITAS DINAMIKA**

2020

Work Hard AND Smart don't just do one thing...



UNIVERSITAS
Dinamika



UNIVERSITAS
Dinamika

*Kupersembahkan karya ini kepada Ibu dan Ayah Tercinta,
Serta keluarga besar dan teman-temanku.*

LEMBAR PENGESAHAN

**RANCANG BANGUN APLIKASI PENCATATAN KEUANGAN BERBASIS
WEBSITE PADA PESANTREN MAHASISWA MAHAD THAYBAH**

Laporan Kerja Praktik oleh
Mochammad Aprilandi Nanu Dwi Firmansyah
Nim : 17410100053
Telah diperiksa, diuji dan disetujui

Surabaya, 20 Juli 2020

Disetujui :

Pembimbing

**Sulisti
owati**

Sulistiowati, S.Si., M.M.
NIDN. 0719016801

Penyelia



Mengetahui

Ketua Program Studi SI Sistem Informasi

**Anjik
Sukmaaji**

Dr. Anjik Sukmaaji, S.Kom., M.Eng.
NIDN. 0731057301

Digitally signed
by Anjik
Sukmaaji
Date: 2020.07.26
21:11:14 +07'00'

SURAT PERNYATAAN

PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH

Sebagai mahasiswa Universitas Dinamika, saya :

Nama : Mochammad Aprilandi Nanu Dwi Firmansyah
NIM : 17410100053
Program Studi : S1 Sistem Informasi
Fakultas : Fakultas Teknologi dan Informatika
Jenis Karya : Laporan Kerja Praktik
Judul Karya : **RANCANG BANGUN APLIKASI PENCATATAN
KEUANGAN BERBASIS WEBSITE PADA
PESANTREN MAHASISWA MAHAD THAYBAH**

Menyatakan dengan sesungguhnya bahwa:

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, saya menyetujui memberikan kepada Universitas Dinamika Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalti Free Right*) atas seluruh isi/ sebagian karya ilmiah saya tersebut di atas untuk disimpan, dialihmediakan dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama saya sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta
2. Karya tersebut di atas adalah karya asli saya, bukan plagiat baik sebagian maupun keseluruhan. Kutipan, karya atau pendapat orang lain yang ada dalam karya ilmiah ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiat pada karya ilmiah ini, maka saya bersedia untuk menerima pencabutan terhadap gelar kesarjanaan yang telah diberikan kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Surabaya, 20 Juli 2020

Yang menyatakan



Mochammad Aprilandi Nanu Dwi Firmansyah
NIM : 17410100053

ABSTRAK

Vertice adalah salah satu *Software House* yang berlokasi di Surabaya. Pada saat ini Vertice mendapatkan proyek untuk pembuatan aplikasi berbasis *website* untuk mengelola pencatatan keuangan pada Pesantren Mahasiswa Mahad Thaybah. Kegiatan yang dilakukan oleh bagian keuangan pada Pesantren Mahasiswa Mahad Thaybah yaitu pencatatan pada saat terjadinya transaksi kas masuk dan kas keluar yang disertai dengan tanda bukti ke dalam buku besar. Kas masuk meliputi iuran bulanan santri, pemasukan dari donatur dan yayasan. Kas keluar meliputi pembelian perlengkapan, pembelian bahan baku makanan, penyewaan olahraga, kegiatan kajian/buka Bersama, dan lain-lain. Setiap harinya dari catatan di buku besar di inputkan ke dalam Program Microsoft Excel, sedangkan pada akhir bulan dilakukan pembuatan laporan. Permasalahan yang dihadapi oleh Pesantren Mahasiswa Mahad Thaybah yaitu karyawan melakukan 2 (dua) kali kerja dalam pencatatan transaksi keuangan dari mencatat di buku besar dan menginputkan data dari buku besar tersebut ke Program Microsoft Excel, kesulitan dalam mencari bon atau bukti transaksi dalam pembuatan laporan, pembuatan laporan memerlukan waktu 3 (tiga) jam.

Solusi yang ditawarkan yakni pembuatan rancang bangun aplikasi pencatatan keuangan berbasis *website* dengan *framework* Laravel meliputi pengelolaan master kategori keuangan, master biaya, pencatatan iuran, pencatatan transaksi disertai dengan upload bukti transaksi, serta pembuatan laporan.

Kata kunci: Laravel, *website*, Keuangan, Pesantren Mahasiswa Mahad Thaybah.

KATA PENGANTAR

Puji syukur kehadiran Allah SWT atas segala rahmat dan nikmat yang diberikan-Nya sehingga penulis dapat menyelesaikan Laporan Kerja Praktik yang berjudul “Rancang Bangun Aplikasi Pencatatan Keuangan Berbasis Website Pada Pesantren Mahasiswa Mahad Thaybah” secara tepat waktu.

Pelaksanaan kerja praktik dan penyelesaian laporan kerja praktik ini tidak terlepas dari bantuan berbagai pihak yang telah memberikan banyak masukan, nasihat, saran, kritik dan dukungan moral maupun materil kepada penulis. Oleh karena itu penulis menyampaikan rasa terima kasih kepada:

1. Ayah, ibu, kakak serta keluarga besar tercinta yang selalu memberikan motivasi dan doa untuk penulis dalam pelaksanaan kerja praktik ini.
2. Bapak Prof. Dr. Budi Jatmiko, M.Pd. selaku rektor Universitas Dinamika yang memberikan kesempatan secara resmi untuk melakukan kerja praktik.
3. Ibu Sulistiowati, S.Si., M.M selaku dosen pembimbing penulis, yang telah membimbing dengan sabar dan memberikan dukungan dalam proses penyelesaian laporan kerja praktik.
4. Bapak Aris Triwibowo selaku CEO Vertice yang telah memberikan kesempatan untuk melaksanakan kerja praktik kepada penulis.
5. Teman-teman tercinta yang memberikan motivasi untuk penulis dalam melaksanakan kerja praktik ini.
6. Pihak-pihak lain yang tidak disebutkan satu-persatu yang telah memberikan bantuan dan dukungan kepada penulis dalam pelaksanaan kerja praktik ini.

Semoga Allah SWT memberikan balasan yang setimpal kepada semua

pihak yang telah memberikan bantuan, bimbingan, nasehat serta motivasi dalam proses pelaksanaan kerja praktik ini. Penulis menyadari bahwa laporan kerja praktik ini terdapat banyak kekurangan, Semoga laporan kerja praktik ini dapat diterima dan bermanfaat bagi penulis dan semua pihak.

Surabaya, 20 Juli 2020

Penulis



UNIVERSITAS
Dinamika

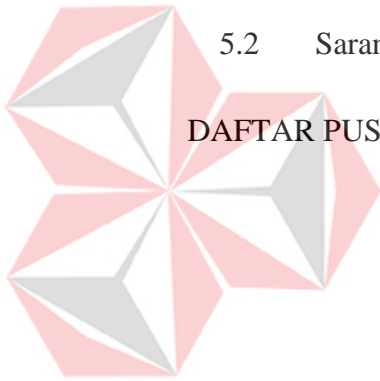
DAFTAR ISI

	Halaman
ABSTRAK	vii
KATA PENGANTAR	viii
DAFTAR ISI.....	x
DAFTAR TABEL	xiv
DAFTAR GAMBAR	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan	2
1.5 Manfaat	2
1.6 Sistematika Penulisan	3
BAB II GAMBARAN UMUM.....	4
2.1 Sejarah Vertice	4
2.2 Profile Perusahaan.....	5
2.3 Logo Vertice.....	5
2.4 Visi	6
2.5 Misi	6
2.6 Struktur Perusahaan	6
2.7 Tugas Dan Wewenang / Tanggung Jawab	7

2.8	CEO.....	7
2.8.1	System Analyst.....	8
2.8.2	Project Manager	8
2.8.3	Content Production	8
2.8.4	Network Architect	8
2.8.5	Designer	9
2.8.6	Programmer.....	9
BAB III LANDASAN TEORI.....		10
3.1	Pondok Pesantren	10
3.2	Akuntansi Keuangan	10
3.3	Website	11
3.4	Database	11
3.5	HTML	11
3.6	PHP	11
3.7	MySQL.....	12
3.8	Web Server	12
3.9	Apache.....	12
3.10	XAMPP	12
3.11	Laravel.....	13
3.12	Use case Diagram	13
3.13	Activity Diagram	13
3.14	Flow Of Event	13
3.15	Sequence Diagram	14
3.16	Class Diagram	14
3.17	User Interface.....	14

BAB IV DESKRIPSI PEKERJAAN	15
4.1 Analisis Sistem.....	15
4.1.1 Kebutuhan Fungsional	15
4.1.2 Kebutuhan Nonfungsional	16
4.2 Perancangan Sistem	16
4.2.1 <i>Use Case Diagram</i>	16
4.2.2 <i>Activity Diagram</i>	17
4.2.3 <i>Flow Of Event</i>	33
4.2.4 <i>Sequence Diagram</i>	47
4.2.5 <i>Class Diagram</i>	57
4.3 Desain User Interface	58
4.3.1 <i>User Interface Login</i>	58
4.3.2 <i>User Interface Menu Utama</i>	59
4.3.3 <i>User Interface Logout</i>	60
4.3.4 <i>User Interface Menu Utama Kategori Keuangan</i>	61
4.3.5 <i>User Interface Tambah Kategori Keuangan</i>	62
4.3.6 <i>User Interface Ubah Kategori Keuangan</i>	63
4.3.7 <i>User Interface Hapus Kategori Keuangan</i>	64
4.3.8 <i>User Interface Menu Utama Biaya</i>	65
4.3.9 <i>User Interface Tambah Biaya</i>	66
4.3.10 <i>User Interface Ubah Biaya</i>	68
4.3.11 <i>User Interface Hapus Biaya</i>	69
4.3.12 <i>User Interface Menu Utama Iuran</i>	70
4.3.13 <i>User Interface Tambah Iuran</i>	71
4.3.14 <i>User Interface Ubah Iuran</i>	72

4.3.15	<i>User Interface</i> Hapus Iuran.....	73
4.3.16	<i>User Interface</i> Menu Utama Transaksi	74
4.3.17	<i>User Interface</i> Tambah Transaksi.....	75
4.3.18	<i>User Interface</i> Detail Transaksi	76
4.3.19	<i>User Interface</i> Ubah Transaksi	77
4.3.20	<i>User Interface</i> Hapus Transaksi.....	78
4.3.21	<i>User Interface</i> Menu Utama Laporan	79
4.3.22	<i>User Interface</i> Menu Detail Laporan	80
BAB V PENUTUP.....		82
5.1	Kesimpulan	82
5.2	Saran.....	82
DAFTAR PUSTAKA		83



UNIVERSITAS
Dinamika

DAFTAR TABEL

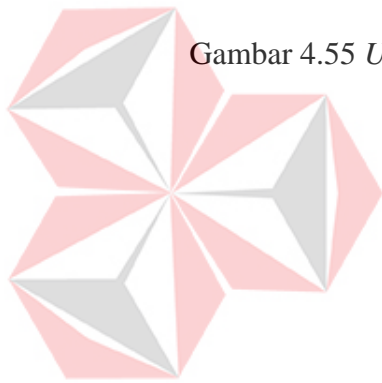
	Halaman
Table 4-1 <i>Flow of Event Insert</i> Master Kategori Keuangan	35
Table 4-2 <i>Flow of Event Update</i> Master Kategori Keuangan.....	36
Table 4-3 <i>Flow of Event Delete</i> Master Transaksi.....	37
Table 4-4 <i>Flow of Event Insert</i> Master Biaya	38
Table 4-5 <i>Flow of Event Update</i> Master Biaya.....	39
Table 4-6 <i>Flow of Event Delete</i> Master Biaya.....	40
Table 4-7 <i>Flow of Event Insert</i> Iuran	41
Table 4-8 <i>Flow of Event Update</i> Iuran.....	42
Table 4-9 <i>Flow of Event Delete</i> Iuran.....	43
Table 4-10 <i>Flow of Event Insert</i> Transaksi	44
Table 4-11 <i>Flow of Event Update</i> Transaksi.....	45
Table 4-12 <i>Flow of Event Delete</i> Transaksi.....	46

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Peta Vertice	5
Gambar 4.1 Usecase Diagram Modul Keuangan	16
Gambar 4.2 <i>Activity Diagram Login</i>	17
Gambar 4.3 <i>Activity Diagram Add Kategori Keuangan</i>	18
Gambar 4.4 <i>Activity Diagram Edit Kategori Keuangan</i>	19
Gambar 4.5 <i>Activity Diagram Delete Kategori Keuangan</i>	20
Gambar 4.6 <i>Activity Diagram Add Biaya</i>	21
Gambar 4.7 <i>Activity Diagram Edit Biaya</i>	22
Gambar 4.8 <i>Activity Diagram Delete Biaya</i>	23
Gambar 4.9 <i>Activity Diagram Add Iuran</i>	24
Gambar 4.10 <i>Activity Diagram Edit Iuran</i>	25
Gambar 4.11 <i>Activity Diagram Delete Iuran</i>	27
Gambar 4.12 <i>Activity Diagram Add Transaksi</i>	28
Gambar 4.13 <i>Activity Diagram Edit Transaksi</i>	30
Gambar 4.14 <i>Activity Diagram Delete Transaksi</i>	32
Gambar 4.15 <i>Sequence Diagram Login</i>	47
Gambar 4.16 <i>Sequence Diagram Kategori Keuangan</i>	48
Gambar 4.17 <i>Sequence Diagram Biaya</i>	50
Gambar 4.18 <i>Sequence Diagram Iuran</i>	52
Gambar 4.19 <i>Sequence Diagram Transaksi</i>	54
Gambar 4.20 <i>Sequence Diagram Laporan</i>	56

Gambar 4.21 <i>Class Diagram</i> Modul Keuangan.....	57
Gambar 4.22 <i>User Interface Login</i>	58
Gambar 4.23 <i>User Interface Menu Utama</i>	59
Gambar 4.24 <i>User Interface Logout</i>	60
Gambar 4.25 <i>User Interface</i> tombol <i>logout</i> yang berada di menu <i>profile</i>	60
Gambar 4.26 <i>User Interface</i> tombol <i>logout</i> yang berada di <i>sidebar</i>	60
Gambar 4.27 <i>User Interface Menu Utama Kategori Keuangan</i>	61
Gambar 4.28 <i>User Interface Tombol Tambah Data Kategori Keuangan</i>	62
Gambar 4.29 <i>User Interface Pop Up Form Tambah Kategori Keuangan</i>	62
Gambar 4.30 <i>User Interface data kategori keuangan</i>	63
Gambar 4.31 <i>User Interface Pop Up Form Ubah Kategori Keuangan</i>	63
Gambar 4.32 <i>User Interface Konfirmasi Hapus Kategori Keuangan</i>	64
Gambar 4.33 <i>User Interface Menu Utama Biaya</i>	65
Gambar 4.34 <i>User Interface Tombol Tambah Data Biaya</i>	66
Gambar 4.35 <i>User Interface Pop Up Form Tambah Biaya</i>	67
Gambar 4.36 <i>User Interface Data Biaya</i>	68
Gambar 4.37 <i>User Interface Pop Up Form Ubah Biaya</i>	68
Gambar 4.38 <i>User Interface Konfirmasi Hapus Biaya</i>	69
Gambar 4.39 <i>User Interface Menu Utama Iuran</i>	70
Gambar 4.40 <i>User Interface Tombol Tambah Data Iuran</i>	71
Gambar 4.41 <i>User Interface Form Tambah Iuran</i>	71
Gambar 4.42 <i>User Interface Data Iuran</i>	72
Gambar 4.43 <i>User Interface Form Ubah Iuran</i>	72
Gambar 4.44 <i>User Interface Konfirmasi Hapus Iuran</i>	73

Gambar 4.45 <i>User Interface</i> Menu Utama Transaksi	74
Gambar 4.46 <i>User Interface</i> Tombol Tambah Data Transaksi.....	75
Gambar 4.47 <i>User Interface</i> Form Tambah Transaksi	75
Gambar 4.48 <i>User Interface</i> Data Transaksi.....	76
Gambar 4.49 <i>User Interface</i> Pop Up Detail Transaksi	76
Gambar 4.50 <i>User Interface</i> Form Ubah Transaksi	77
Gambar 4.51 <i>User Interface</i> Konfirmasi Hapus Transaksi.....	78
Gambar 4.52 <i>User Interface</i> Menu Utama Laporan	79
Gambar 4.53 <i>User Interface</i> Data Laporan yang Telah Di Rekap.....	80
Gambar 4.54 <i>User Interface</i> Detail Laporan.....	80
Gambar 4.55 <i>User Interface</i> Foto Bukti Transaksi.....	81



UNIVERSITAS
Dinamika

BAB I

PENDAHULUAN

1.1 Latar Belakang

Vertice merupakan perusahaan yang bergerak di bidang teknologi informasi dan komunikasi (*Software House*), yang beralamatkan di Jl. Ketintang baru Gang 4b no 40, Surabaya, Jawa Timur, 76115. Vertice melayani beberapa perusahaan untuk membantu pengelolaan Teknologi informasi dan Komunikasi seperti Pesantren Mahasiswa Mahad Thaybah. Di pesantren tersebut, Vertice mendapat proyek untuk membuat aplikasi pencatatan keuangan berbasis website.

Alur proses bisnis pengelolaan keuangan yang dilakukan oleh Pesantren Mahasiswa Mahad Thaybah yaitu bagian keuangan mencatat setiap kali ada kas masuk dan keluar yang terjadi di dalam Pesantren Mahasiswa Mahad Thaybah disertai dengan bukti seperti nota atau kwitansi, dicatat ke dalam buku besar. Selanjutnya dari data yang ada didalam buku besar setiap harinya di inputkan oleh bagian keuangan ke dalam program Microsoft Excel. Kas masuk meliputi iuran bulanan santri, pemasukan dari donatur dan yayasan. Kas keluar meliputi pembelian perlengkapan, pembelian bahan baku makanan, penyewaan olahraga, kegiatan kajian/buka Bersama, dan lain-lain. Terdapat juga pembuatan laporan keuangan yang merupakan rekap-rekap dari semua transaksi perbulan berdasarkan arsip buku besar tersebut yang kemudian di lapokan kepada ketua santri dan ketua yayasan.

Pada proses tersebut ditemukan 3 (tiga) permasalahan utama yang muncul dari proses pencatatan keuangan ini. Permasalahan pertama, pencatatan yang masih secara manual dengan menulis di buku besar yang setelah itu dimasukkan ke *software* Microsoft Excel satu per satu yang menimbulkan karyawan menjadi melakukan 2 (dua) kali kerja. Permasalahan kedua yaitu kesulitan dalam mencari bon atau bukti transaksi dalam pembuatan laporan. Permasalahan ketiga, pembuatan laporan memerlukan waktu 3 (tiga) jam untuk merangkum semua pencatatan yang berada di aplikasi excel. Pihak pesantren juga mengatakan ingin memiliki sistem informasi tentang keuangan yang diakses secara internal yang memudahkan pihak pesantren dalam memasukan dan mengolah datanya.

Dengan penulis membuat rancang bangun aplikasi pencatatan keuangan berbasis *website* pada Pesantren Mahasiswa Mahad Thaybah, membuat aplikasi ini sesuai dengan kebutuhan Pesantren Mahasiswa Mahad Thaybah.

1.2 Rumusan Masalah

Bagaimana merancang dan membangun aplikasi pencatatan keuangan berbasis *website* pada Pesantren Mahasiswa Mahad Thaybah Surabaya.

1.3 Batasan Masalah

Pada pembuatan aplikasi pencatatan keuangan ini terdapat beberapa batasan-batasan yaitu:

- a. Perancangan aplikasi pencatatan keuangan pada Pesantren Mahasiswa Mahad Thaybah menggunakan bahasa pemrograman PHP dan *database* yang digunakan adalah MySQL.
- b. Perancangan aplikasi pada Pesantren Mahasiswa Mahad Thaybah ini menggunakan *Framework* Laravel.
- c. Aplikasi menghasilkan laporan yang merupakan rekap dari transaksi.

1.4 Tujuan

Menghasilkan Rancang Bangun Aplikasi Pencatatan Keuangan pada Pesantren Mahasiswa Mahad Thaybah berbasis *website* untuk mempermudah pengelolaan keuangan, sehingga tidak lagi dilakukan secara manual.

1.5 Manfaat

Manfaat yang diharapkan dari pembuatan rancang bangun aplikasi ini adalah :

- a. Memudahkan pihak Pesantren Dapat memproses pencatatan dan pengelolaan data keuangan.
- b. Membantu dalam mengurangi waktu dalam pembuatan laporan keuangan.
- c. Meminimalisir kesalahan dalam proses pencatatan data, pencarian data, dan penyimpanan data.
- d. Pencatatan pemasukan dan pengeluaran keuangan.

1.6 Sistematika Penulisan

Sistematika penulisan yang digunakan dalam penulisan laporan kerja praktik ini yaitu:

BAB I : PENDAHULUAN

Bab ini menjelaskan tentang latar belakang, rumusan masalah, batasan masalah, tujuan, dan manfaat yang didapat dari kerja praktik ini.

BAB II : GAMBARAN UMUM

Bab ini menjelaskan tentang gambaran umum dari perusahaan seperti sejarah perusahaan, *profile* perusahaan, logo, visi dan misi, struktur perusahaan serta tugas dan wewenang dari masing-masing bagian di dalam perusahaan.

BAB III : LANDASAN TEORI

Bab ini menjelaskan tentang teori-teori yang berkaitan dengan laporan kerja praktik ini.

BAB IV : DESKRIPSI PEKERJAAN

Bab ini menjelaskan tentang hasil kerja praktik yang telah dilakukan di Vertice.

BAB V : PENUTUP

Bab ini menjelaskan tentang kesimpulan yang didapat dari pembuatan aplikasi yang telah dibuat serta saran yang bertujuan untuk pengembangan maupun perbaikan aplikasi dimasa yang akan datang.

BAB II

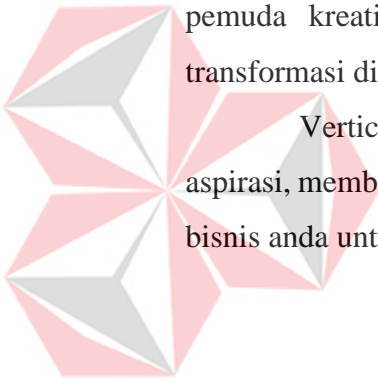
GAMBARAN UMUM

2.1 Sejarah Vertice

Vertice Software Development berdiri pada tahun 2018. Awal berdirinya Vertice ini diawali dengan adanya mantan lulusan Stikom Surabaya yang menginginkan untuk mendirikan sebuah Software House. Dengan pengalamannya di Bidang teknologi informasi, kini vertice telah cukup dikenal dikalangan masyarakat dengan banyaknya perusahaan-perusahaan yang ingin dibuatkan sistem informasinya seperti PT. Tamba Waras, CV Tibra Baswara Dan lain-lain.

Jadi, Vertice Merupakan sebuah Software House yang terdiri dari pemuda-pemuda kreatif, dengan latar belakang pendidikan IT dan percaya bahwa transformasi digital adalah kunci pertumbuhan bisnis anda.

Vertice mampu memahami permasalahan dengan cepat, menampung aspirasi, memberikan solusi efektif dan kreatif yang akan berdampak pada kesiapan bisnis anda untuk bersaing pada era revolusi industri keempat (Industri 4.0).



UNIVERSITAS
Dinamika

2.2 Profile Perusahaan

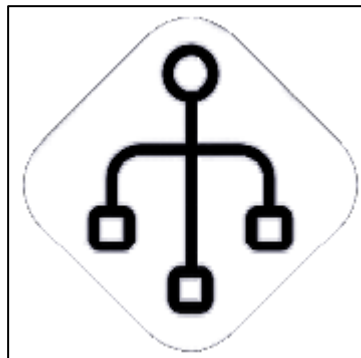


Gambar 2.1 Peta Vertice



Nama Instansi : Vertice Software Development
Alamat : Jl. Ketintang baru Gang 4b no 40, Surabaya.
No. Telepon : (+62)85615222985
No. Fax : -
Website : <https://vertice.id/>
Email : vertice15@gmail.com

2.3 Logo Vertice



Gambar 2.2 Logo Vertice

2.4 Visi

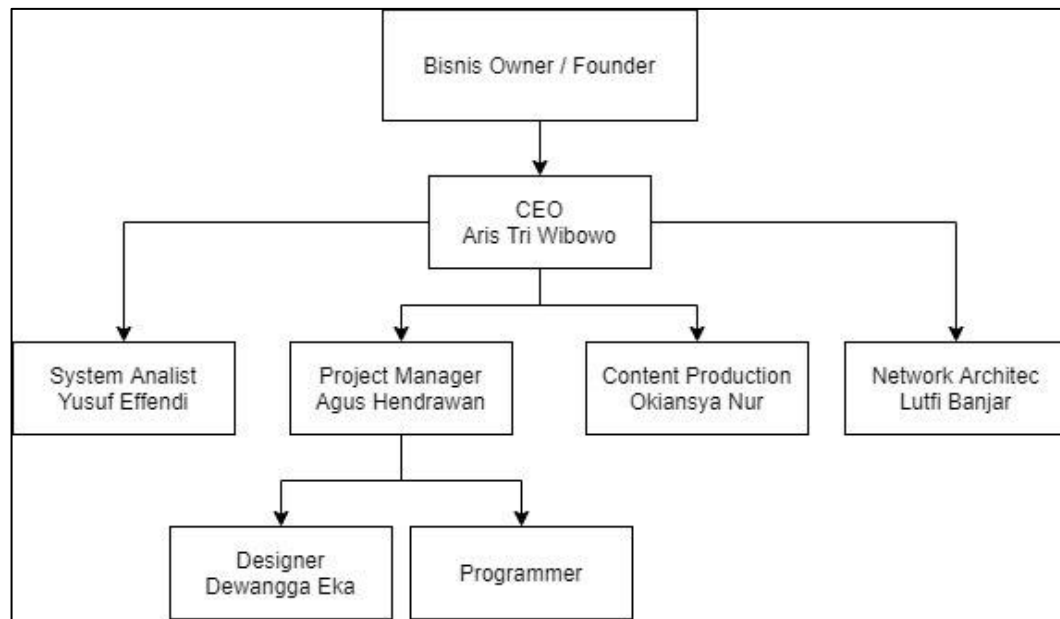
Menjadi persusahaan teknologi informasi yang handal dan *concern* dalam menghasilkan produk digital sebagai solusi terbaik, profesional, *responsive*, integritas dan *innovative*. Serta Menjadikan setiap *project digital* yang dikerjakan memiliki keamanan yang tinggi

2.5 Misi

- a. Mengembangkan produk industri teknologi informasi yang berkualitas dan kompetitif.
- b. Mengedepankan profesionalitas yang tinggi dalam menghasilkan produk yang berkualitas.
- c. Memberikan layanan yang terbaik untuk konsumen.
- d. Meningkatkan *benefit* dan *value* bagi konsumen dan *stakeholder*.
- e. Mengoptimalkan penggunaan teknologi yang reliable, secure dan menguntungkan.

2.6 Struktur Perusahaan

Dari data profile Vertice Tahun 2019 dapat diambil suatu gambaran struktur perusahaan untuk mendukung segala aktifitas pada Vertice sehingga dapat mencapai tujuan yang sudah ditetapkan. Adapun strukturnya sebagai berikut :



Gambar 2.3 Struktur Perusahaan Vertice

2.7 Tugas Dan Wewenang / Tanggung Jawab

Uraian tugas, wewenang dan tanggung jawab dari masing-masing jabatan yang ada di Vertice adalah sebagai berikut :

2.8 CEO

- Merencanakan, mengelola, dan menganalisis segala aktivitas fungsional bisnis seperti operasional, sumber daya manusia, keuangan, dan pemasaran
- Merencanakan dan mengelola proses penganggaran, lalu mengamati dan menganalisis apabila ada kejanggalan dalam prakteknya
- Mengelola perusahaan sesuai dengan tujuan strategis perusahaan dengan keefektivan dan biaya seefisien mungkin
- Merencanakan dan mengelola kinerja pada sumber daya manusia agar sumber daya manusia yang berkompeten teridentifikasi dan dapat ditempatkan pada posisi yang sesuai sehingga dapat memaksimalkan kinerja perusahaan
- Membuat kebijakan, prosedur, dan standar pada perusahaan

2.8.1 System Analyst

- a. Melakukan analisa terhadap sistem/aplikasi yang ada beserta referensi dan dokumen penunjang sistem/aplikasi.
- b. Melakukan analisa kebutuhan sistem.
- c. Menyusun model konseptual dan spesifikasi kebutuhan fungsional.
- d. Membuat perancangan aplikasi dengan menggunakan pemodelan *Unified Modeling Language* (UML).
- e. Membantu *Project Manager/Leader/Ketua* Tim dalam merancang desain sistem/program aplikasi yang efektif dan efisien untuk menciptakan aplikasi yang baik

2.8.2 Project Manager

- a. Memimpin perencanaan dan pelaksanaan proyek.
- b. Mendefinisikan ruang lingkup proyek, tujuan dan penyampaian.
- c. Menyusun dan mengkoordinasikan staff proyek.
- d. Mengelola anggaran dan alokasi sumber daya proyek.
- e. Perencanaan dan penjadwalan proyek.

2.8.3 Content Production

- a. Mengumpulkan ide serta data
- b. Melakukan riset untuk membuat konsep yang akan dijadikan sebuah konten
- c. Menghasilkan konten yang sesuai dengan identitas dan branding yang diinginkan untuk memenuhi tujuan yang disepakati dari sebuah konten

2.8.4 Network Architect

- a. Bertanggung jawab untuk melaksanakan, menjaga, mengembangkan, serta merancang jaringan komunikasi dalam suatu perusahaan atau antara perusahaan.
- b. Menguji dan mengevaluasi hardware dan software untuk menentukan efisiensi, reliabilitas, dan kompatibilitas dengan sistem yang ada, dan membuat rekomendasi pembelian.
- c. Memantau kinerja sistem dan menyediakan langkah-langkah keamanan, tips

dan pemeliharaan yang diperlukan.

2.8.5 Designer

- a. Berpikir kreatif untuk menghasilkan ide-ide dan konsep-konsep baru dan mengembangkan desain interaktif
- b. Menggunakan inovasi untuk mendefinisikan kembali desain dalam keterbatasan biaya dan waktu

2.8.6 Programmer

- a. Menemukan urutan instruksi yang akan mengotomatisasi dalam melakukan tugas tertentu atau memecahkan masalah yang diberikan.
- b. Menyusun dan menulis dokumentasi pengembangan program dan revisi berikutnya, memasukkan komentar dalam kode instruksi sehingga orang lain dapat memahami program yang dikembangkan.
- c. Melakukan percobaan menjalankan program dan aplikasi software untuk memastikan bahwa program menghasilkan informasi yang dikehendaki dan bahwa instruksi sudah benar.
- d. Menganalisis, meninjau, dan menulis ulang program, menggunakan grafik dan diagram alur kerja, dan menerapkan pengetahuan tentang kemampuan komputer, materi pelajaran, dan logika simbolik.
- e. Memperbaiki kesalahan dengan membuat perubahan yang sesuai dan memeriksa kembali program untuk memastikan bahwa hasil yang diinginkan telah berhasil.

BAB III

LANDASAN TEORI

Dalam merancang dan membangun aplikasi ini, memerlukan sebuah teori-teori untuk membantu dalam menyelesaikan penelitian ini dan sistem yang dibuat.

3.1 Pondok Pesantren

Definisi pondok pesantren adalah sebuah lembaga pendidikan Islam, di mana para santri dan kiai tinggal bersama dalam satu lingkungan asrama (komplek). Para santri yang belajar di pondok pesantren tidak hanya dituntut menguasai ilmu-ilmu yang diajarkan oleh kiai atau ustadz, namun sekaligus mengamalkannya dalam kehidupan sehari-hari. Jadi istilah pondok pesantren berasal dari dua kata, yaitu pondok dan pesantren. Pondok adalah tempat mondok, sedangkan pesantren berasal dari kata santri. Jadi pondok pesantren adalah tempat mencari ilmu yang anak didiknya diasramakan (Mukhlis, 2018).

3.2 Akuntansi Keuangan

Akuntansi keuangan adalah sebuah proses yang berakhir pada pembuatan laporan keuangan menyangkut perusahaan secara keseluruhan untuk digunakan baik pihak-pihak internal maupun pihak eksternal (Kieso, Weygandt, & Warfield, 2008).

Akuntansi keuangan berorientasi pada pelaporan pihak eksternal. Beragamnya pihak eksternal dengan tujuan spesifik bagi masing-masing pihak membuat pihak penyusun laporan keuangan menggunakan prinsip dan asumsi-asumsi dalam penyusunan laporan keuangan. Untuk itu diperlukan standar akuntansi yang dijadikan pedoman baik oleh penyusun maupun oleh pembaca laporan keuangan. Laporan yang dihasilkan dari akuntansi keuangan berupa laporan keuangan untuk tujuan umum (*general purpose financial statement*) (Martani, Siregar, Wardhani, Farahmita, & Tanujaya, 2012).

3.3 Website

Website adalah sebuah kumpulan halaman pada suatu domain di internet yang dibuat dengan tujuan tertentu dan saling berhubungan serta dapat diakses secara luas melalui halaman depan (*home page*) menggunakan sebuah browser menggunakan URL *website* (Firmansyah & Pitriani, 2017).

Jenis *website* (*web*) dapat dikategorikan menjadi dua yaitu *web* statis dan *web* dinamis. *Web* Statis adalah *web* yang menampilkan informasi-informasi yang sifatnya statis (tetap) (Firmansyah, Maulana, & Fatin, 2020). Jadi *website* dapat diartikan sebagai keseluruhan halaman-halaman web yang terdapat dari sebuah domain yang mengandung informasi (Prayitno & Safitri, 2015).

3.4 Database

Database merupakan suatu kesatuan yang dibentuk dari gabungan tabel dan file, di mana setiap tabel terdiri dari *record* yang disusun atas *field-field* yang ada di dalamnya (Agustini & Kurniawan, 2019).

3.5 HTML

HTML (*Hyper Text Markup Language*) sebenarnya bukan sebuah bahasa pemrograman, karena HTML digunakan untuk *mark up* (penanda) terhadap suatu dokumen teks. Simbol *mark up* yang digunakan oleh HTML ditandai dengan tanda lebih kecil (<) dan tanda lebih besar (>). Kedua tanda ini disebut *tag*. *Tag* yang digunakan sebagai tanda penutup diberi karakter garis miring (</...>) (Binarso, Sarwoko, & Bahtiar, 2012).

3.6 PHP

PHP (*Hypertext Preprocessor*) adalah sebuah bahasa pemrograman *web* berbasis *server* (*server-side scripting*) yang mampu memarsing kode PHP dari kode *web* dengan ekstensi .php, sehingga menghasilkan tampilan website yang dinamis di sisi *client* (*browser*). PHP adalah bahasa script yang sangat cocok untuk pengembangan web dan dapat dimasukkan ke dalam HTML (Agustini & Kurniawan, 2019).

3.7 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (DBMS) yang *multithread*, dan *multi-user*. MySQL adalah implementasi dari sistem manajemen basis data relasional (RDBMS). MySQL dibuat oleh TcX dan telah dipercaya mengelola sistem dengan 40 buah *database* berisi 10.000 tabel dan 500 di antaranya memiliki 7 juta baris. Pada saat ini MySQL merupakan *database server* yang sangat terkenal di dunia, semua itu tak lain karena bahasa dasar yang digunakan untuk mengakses *database* yaitu SQL. SQL (*Structured Query Language*) pertama kali diterapkan pada sebuah proyek riset pada laboratorium riset San Jose, IBM yang bernama system R. Kemudian SQL juga dikembangkan oleh Oracle, Informix dan Sybase. Dengan menggunakan SQL, proses pengaksesan *database* lebih *user-friendly* dibandingkan dengan yang lain, misalnya dBase atau Clipper karena mereka masih menggunakan perintah-perintah pemrograman murni (Vonny, 2018).

3.8 Web Server

Web Server adalah sebuah *software* yang memberikan layanan berbasis data dan berfungsi menerima permintaan dari HTTP atau HTTPS pada klien yang dikenal dan biasanya kita kenal dengan nama *web browser* dan untuk mengirimkan kembali yang hasilnya dalam bentuk beberapa halaman *web* dan pada umumnya akan berbentuk dokumen HTML (Budiman & Nugraha, 2019).

3.9 Apache

Apache adalah perangkat lunak berbasis *web server* yang gratis dan bersifat *open source*. *Server* ini telah menjadi platform bagi 46% *website* di seluruh dunia. Apache dibuat dan diatur oleh *Apache Software Foundation* dan sudah cukup lama beroperasi serta dapat diandalkan. Apache memiliki nama asli yaitu Apache HTTP *Server* yang berfungsi bagi *developer* sebuah situs *web* untuk menciptakan konten-konten di situs *web* miliknya (Minokaura, et al., 2020).

3.10 XAMPP

XAMPP merupakan singkatan X (empat *system* operasi apapun), Apache,

MySQL, PHP, Perl. XAMPP merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dalam paketnya sudah terdapat Apache (*web server*), MySQL (*database*), PHP (*server side scripting*), Perl, FTP *server*, phpMyAdmin dan berbagai pustaka bantu lainnya (Sari, 2019).

3.11 Laravel

Laravel adalah sebuah *framework* PHP yang dirilis di bawah lisensi MIT, dibangun dengan konsep MVC(*Model View Controller*). Laravel adalah pengembangan *website* berbasis MVC yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, dan untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas dan menghemat waktu (Hermanto, Yusman, & Nagara, 2019).

3.12 Use case Diagram

Use case diagram merupakan pemodelan untuk kelakuan sistem informasi yang akan dibuat. *Use case* bekerja dengan mendeskripsikan tipikal interaksi antara *user* sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sistem itu dipakai (Putra & Andriani, 2019).

3.13 Activity Diagram

Activity diagram merupakan diagram yang menggambarkan *workflow* atau aktivitas dari sebuah sistem yang ada pada perangkat lunak. (Putra & Andriani, 2019)

3.14 Flow Of Event

Elemen ini adalah elemen terakhir dalam penyusunan *Use case description* yang dimana elemen ini menjelaskan tentang proses bisnis dari tiap *use case* yang ada (Tegarden, Dennis, & Wixom, 2013). *Flow of event* sendiri dibagi menjadi tiga *steps* antara lain:

a. *Normal flow*

Berisikan tentang *steps* atau langkah langkah umum dari proses

bisnis yang ada dari tiap use case.

b. *Sub-Flow*

Berisikan tentang proses bisnis yang dipecah dari *normal flow* yang terlalu kompleks sehingga menjadi sebuah proses lain yang masih berhubungan dengan *normal flow*-nya

c. *Alternative/Exceptional flow*

Beriskan informasi tentang kendala-kendala yang menyebabkan proses bisnis tidak dapat berjalan secara baik.

3.15 *Sequence Diagram*

Menggambarkan interaksi antara sejumlah objek dalam urutan waktu. Kegunannya untuk menunjukan rangkaian pesan yang dikirim antara objek juga interaksi antar objek yang terjadi pada titik tertentu dalam eksekusi sistem (Isa & Hartawan, 2017).

3.16 *Class Diagram*

Class diagram merupakan gambaran struktur sistem dari segi pendefinisian kelas- kelas yang akan dibuat untuk membangun sistem. *Class diagram* terdiri dari atribut dan operasi dengan tujuan pembuat pembuat program dapat membuat hubungan antara dokumentasi perancangan dan perangkat lunak sesuai. (Putra & Andriani, 2019).

3.17 *User Interface*

Antarmuka pengguna atau *user interface* (UI) adalah suatu istilah yang digunakan untuk menggambarkan tampilan dari mesin atau komputer yang berinteraksi langsung dengan pengguna (Ghiffary, Susanto, & Herdiyanti, 2018).

BAB IV

DESKRIPSI PEKERJAAN

Deskripsi pekerjaan yang dibahas tentang perancangan sistem yang dilakukan pada saat kerja praktik yang terdiri atas *requirement analysis, system design, implementation, integration & testing, operation dan testing*. langkah-langkah tersebut mengacu pada metode SDLC (*System Development Life Cycle*) yang disebut *waterfall*.

4.1 Analisis Sistem

Pengembangan aplikasi yang dilakukan memerlukan analisis sistem yang tepat dengan proses bisnis yang ada. Proses dimulai dari bagian keuangan mencatat setiap transaksi yang terjadi pada Pesantren Mahasiswa Mahad Thaybah setiap terjadinya transaksi kas masuk dan keluar. Kas masuk didapat dari pembayaran iuran oleh santri, dan donasi dari donator atau yayasan. Kas keluar merupakan biaya-biaya yang dikeluarkan oleh pihak Pesantren Mahasiswa Mahad Thaybah. Setelah itu semua catatan tersebut diberikan ke sekertaris untuk dipindah ke *software Microsoft Excel* yang kemudian dilaporkan ke ketua santri dan ketua yayasan.

Sehingga untuk memenuhi kebutuhan tersebut dilakukan beberapa yang diantaranya :

4.1.1 Kebutuhan Fungsional

Analisa kebutuhan fungsional kerja praktik pada CV Vertice sebagai berikut :

1. Merancang *basis data* yang mampu menyimpan data sesuai kebutuhan Pesantren Mahasiswa Mahad Thaybah
2. Sistem dapat mengelola data inventaris (*view, input, dan edit*).

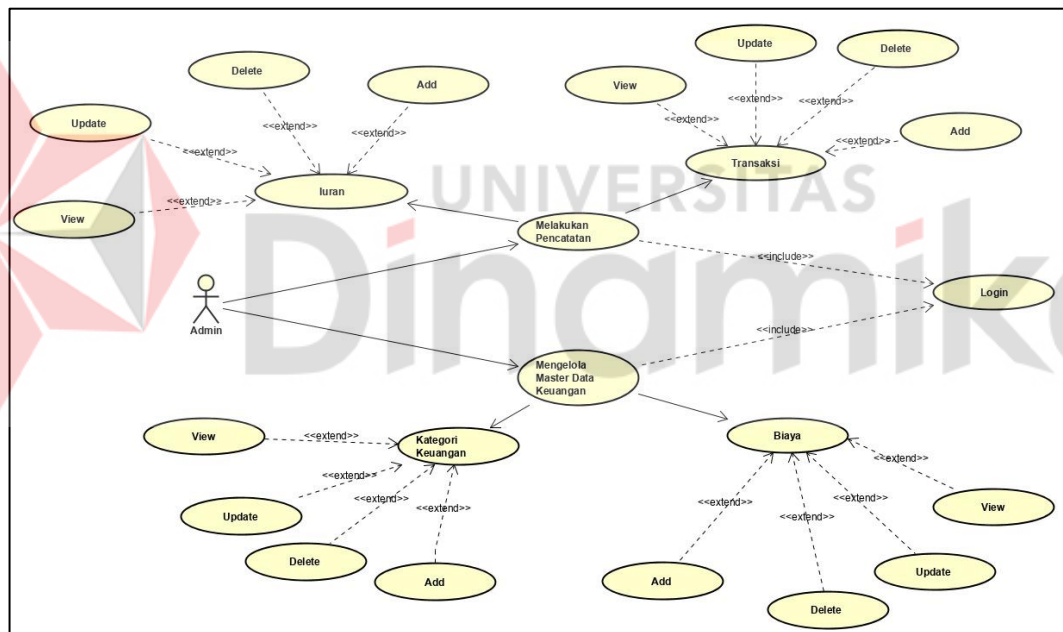
4.1.2 Kebutuhan Nonfungsional

Analisis kebutuhan non fungsional yang digunakan yaitu keamanan. Untuk masuk ke dalam aplikasi, *user* harus *login* menggunakan *username* dan *password* tertentu

4.2 Perancangan Sistem

Berikut ini merupakan *use case* Aplikasi pencatatan keuangan pada pesantren mahasiswa mahad thaybah. Pada *use case* ini terdapat 1 aktor yaitu *Admin*. Terdapat 5 proses bisnis yang terdapat pada *use case* tersebut. Berikut diantaranya :

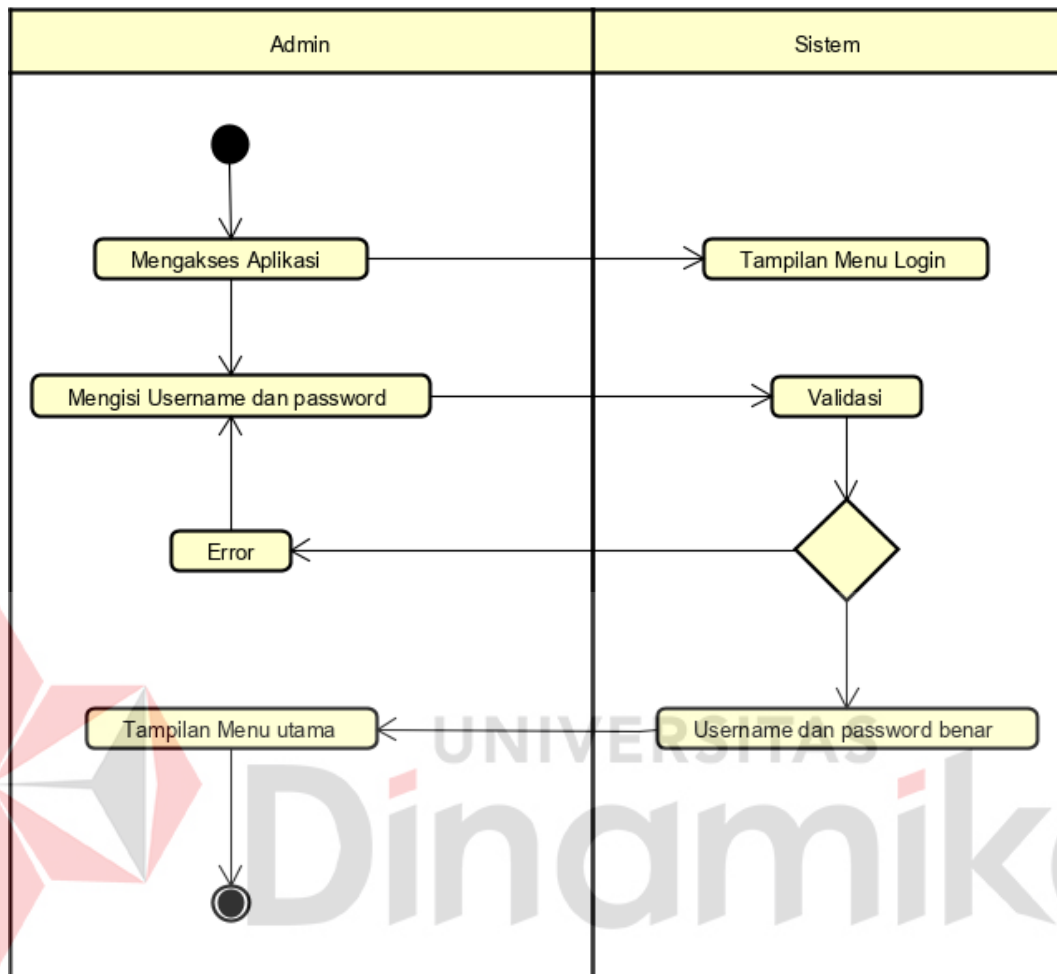
4.2.1 Use Case Diagram



Gambar 4.1 Usecase Diagram Modul Keuangan

4.2.2 Activity Diagram

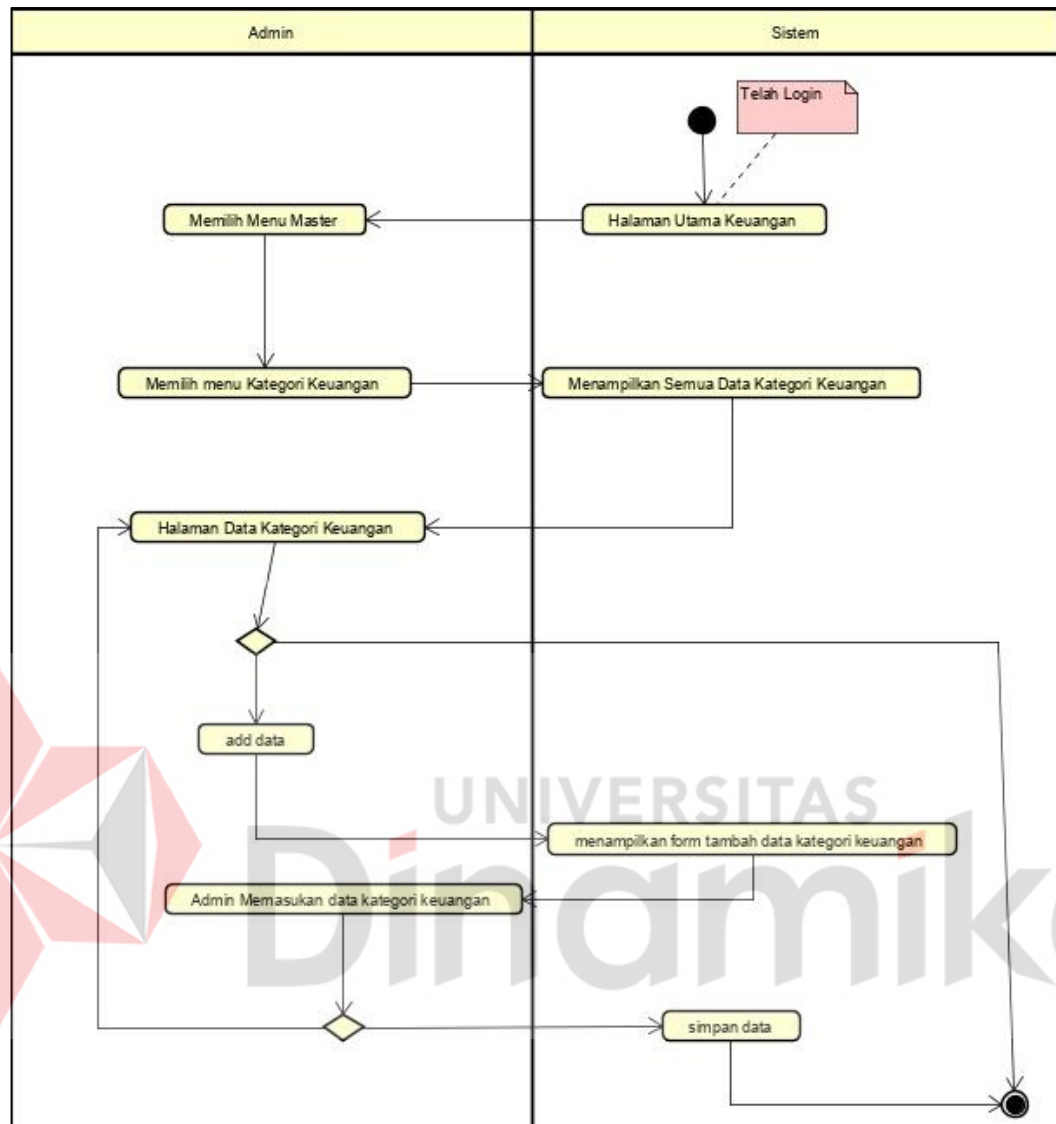
A Activity Diagram Login



Gambar 4.2 Activity Diagram Login

Pada gambar 4.2 menggambarkan proses aktivitas *login*. Diagram Tersebut menjelaskan bahwa pihak admin diharuskan memasukkan *username* dan *password* untuk masuk ke halaman utama.

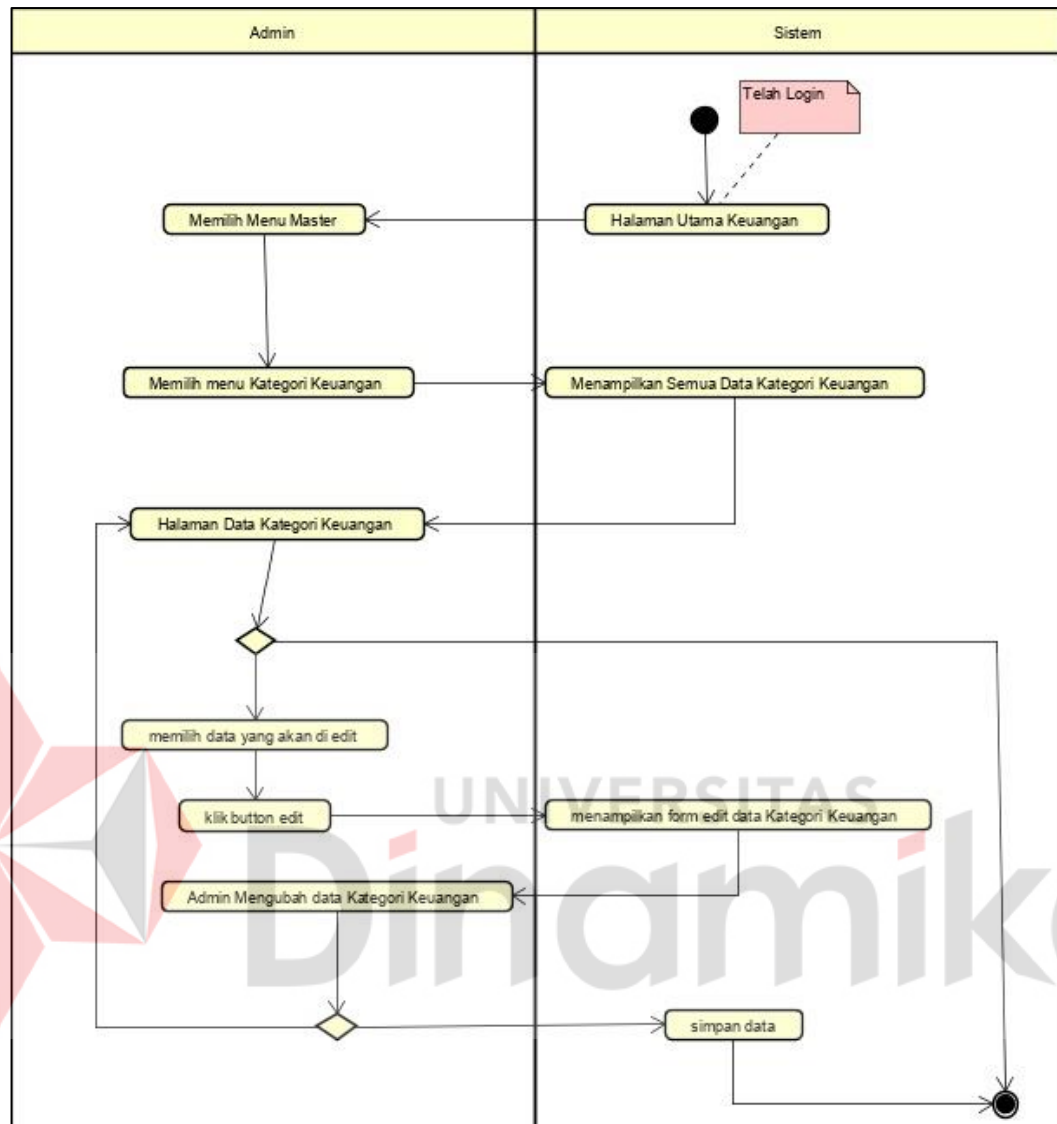
B Activity Diagram add Kategori Keuangan



Gambar 4.3 Activity Diagram Add Kategori Keuangan

Pada gambar diatas, Menggambarkan proses aktivitas menambah data kategori keuangan. Diagram tersebut menjelaskan bahwa setelah *admin login* akan masuk ke halaman utama. Untuk menambah kategori keuangan, *admin* diharuskan memilih *menu* kategori keuangan dan mengklik *button* tambah, setelah itu *admin* diharuskan mengisi *form* tambah dan menyimpannya.

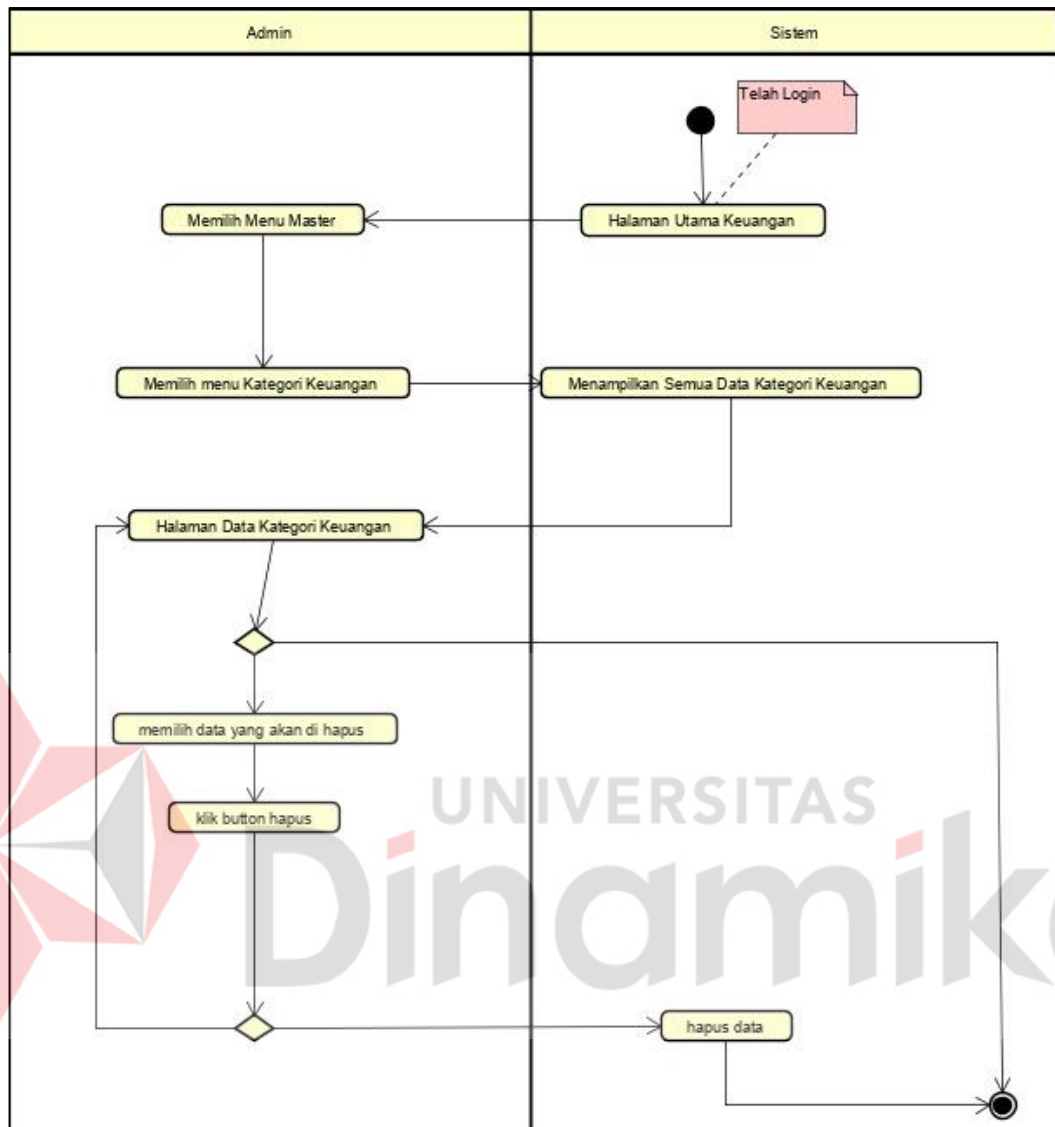
C Activity Diagram Edit Kategori Keuangan



Gambar 4.4 Activity Diagram Edit Kategori Keuangan

Pada gambar diatas, Menggambarkan proses aktivitas mengubah data kategori keuangan. Diagram tersebut menjelaskan bahwa setelah *admin login* akan masuk ke halaman utama. Untuk mengubah kategori keuangan, *admin* diharuskan memilih *menu* kategori keuangan dan mengklik *button edit* pada data yang ingin di *update*, setelah itu *admin* diharuskan menubah data yang diinginkan dan menyimpannya.

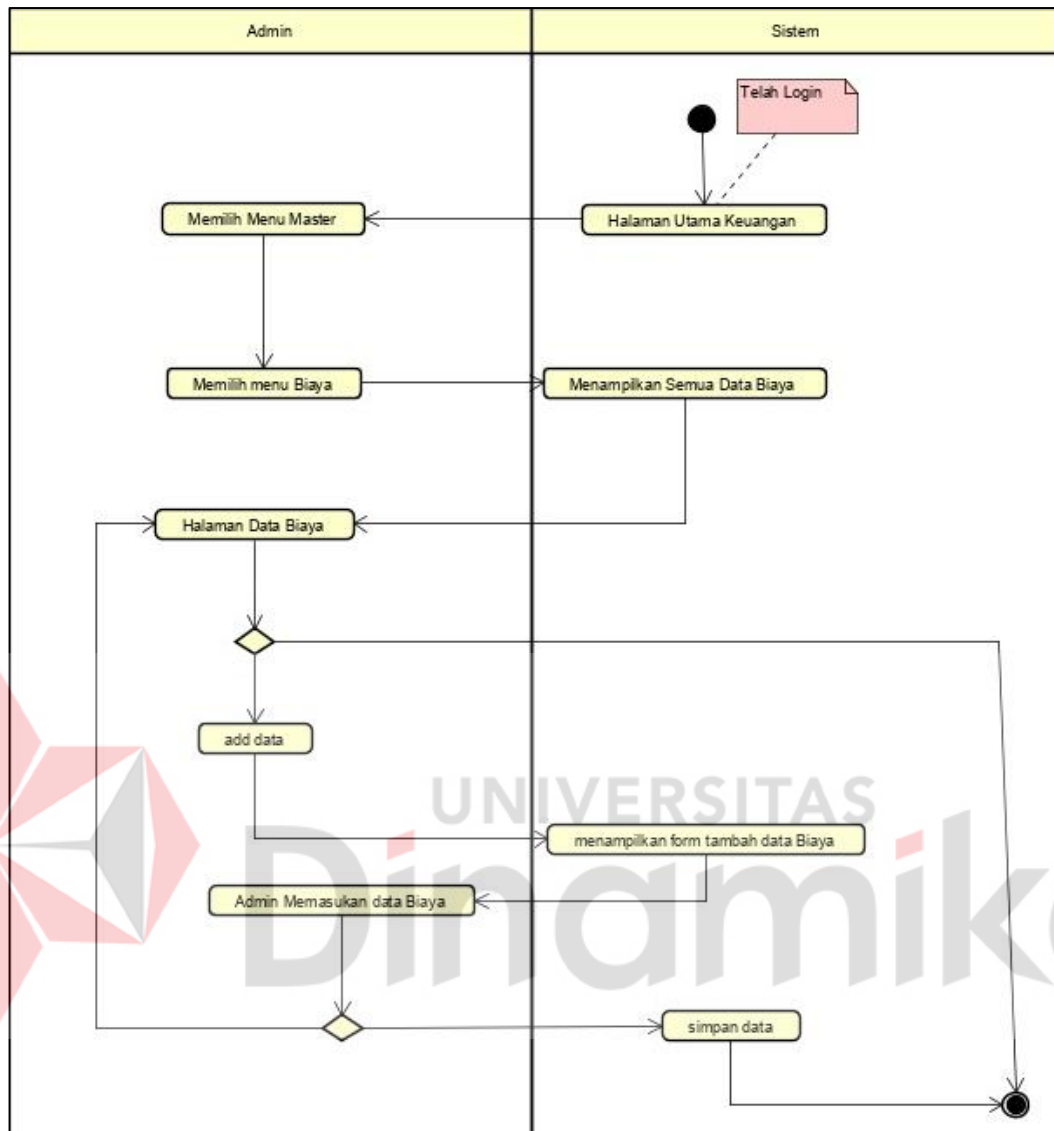
D Activity Diagram Delete Kategori Keuangan



Gambar 4.5 Activity Diagram Delete Kategori Keuangan

Pada gambar diatas, Menggambarkan proses aktivitas menghapus data kategori keuangan. Diagram tersebut menjelaskan bahwa setelah *admin login* akan masuk ke halaman utama. Untuk menghapus data kategori keuangan, *admin* diharuskan memilih *menu* kategori keuangan dan mengklik *button* hapus di data yang ingin dihapus, setelah itu *admin* diharuskan mengkonfirmasi penghapusan.

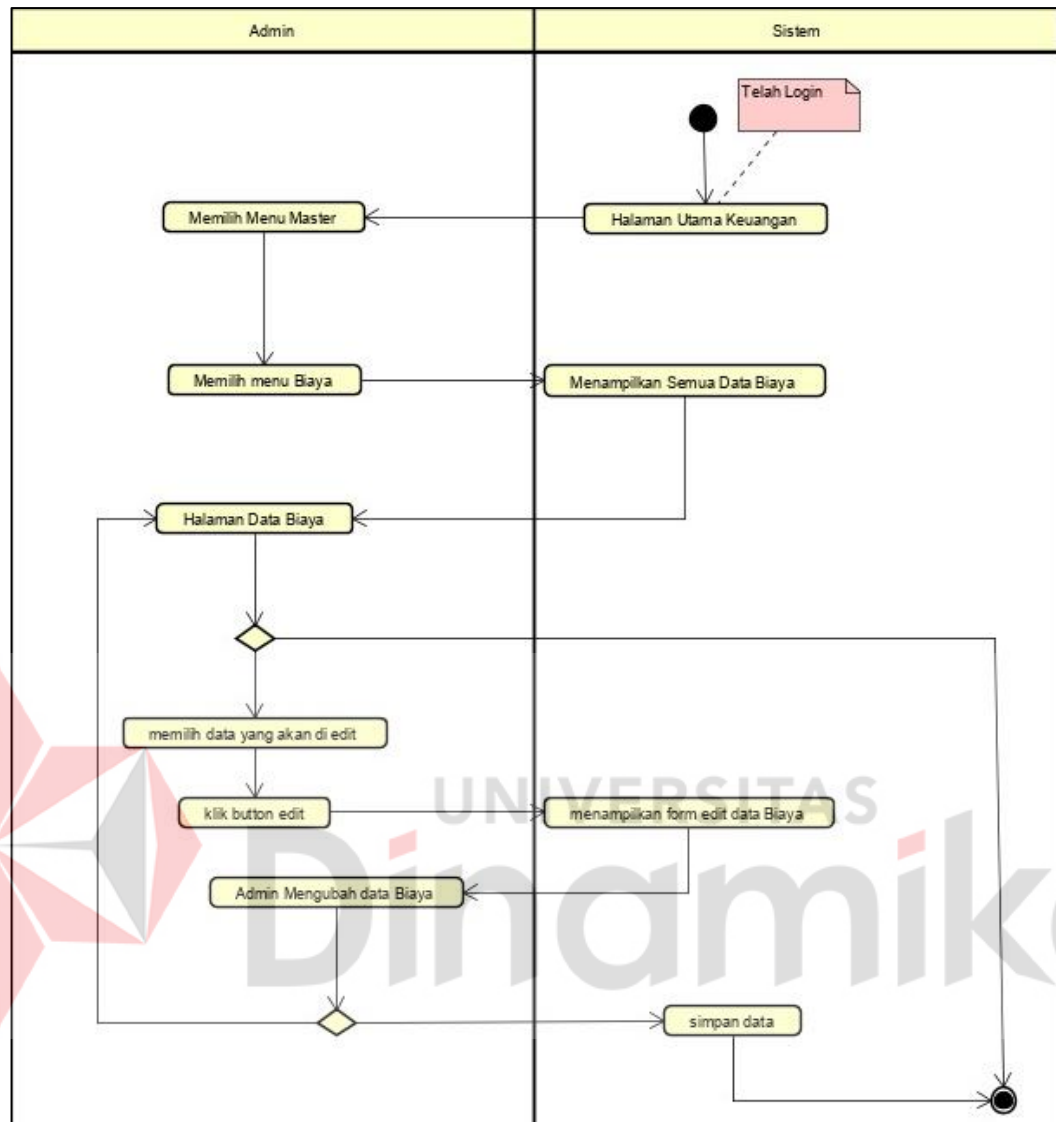
E Activity Diagram Add Biaya



Gambar 4.6 Activity Diagram Add Biaya

Pada gambar diatas, Menggambarkan proses aktivitas menambah data biaya. Diagram tersebut menjelaskan bahwa setelah *admin login* akan masuk ke halaman utama. Untuk menambah biaya, *admin* diharuskan memilih *menu* biaya dan mengklik *button* tambah, setelah itu *admin* diharuskan mengisi *form* tambah dan menyimpannya.

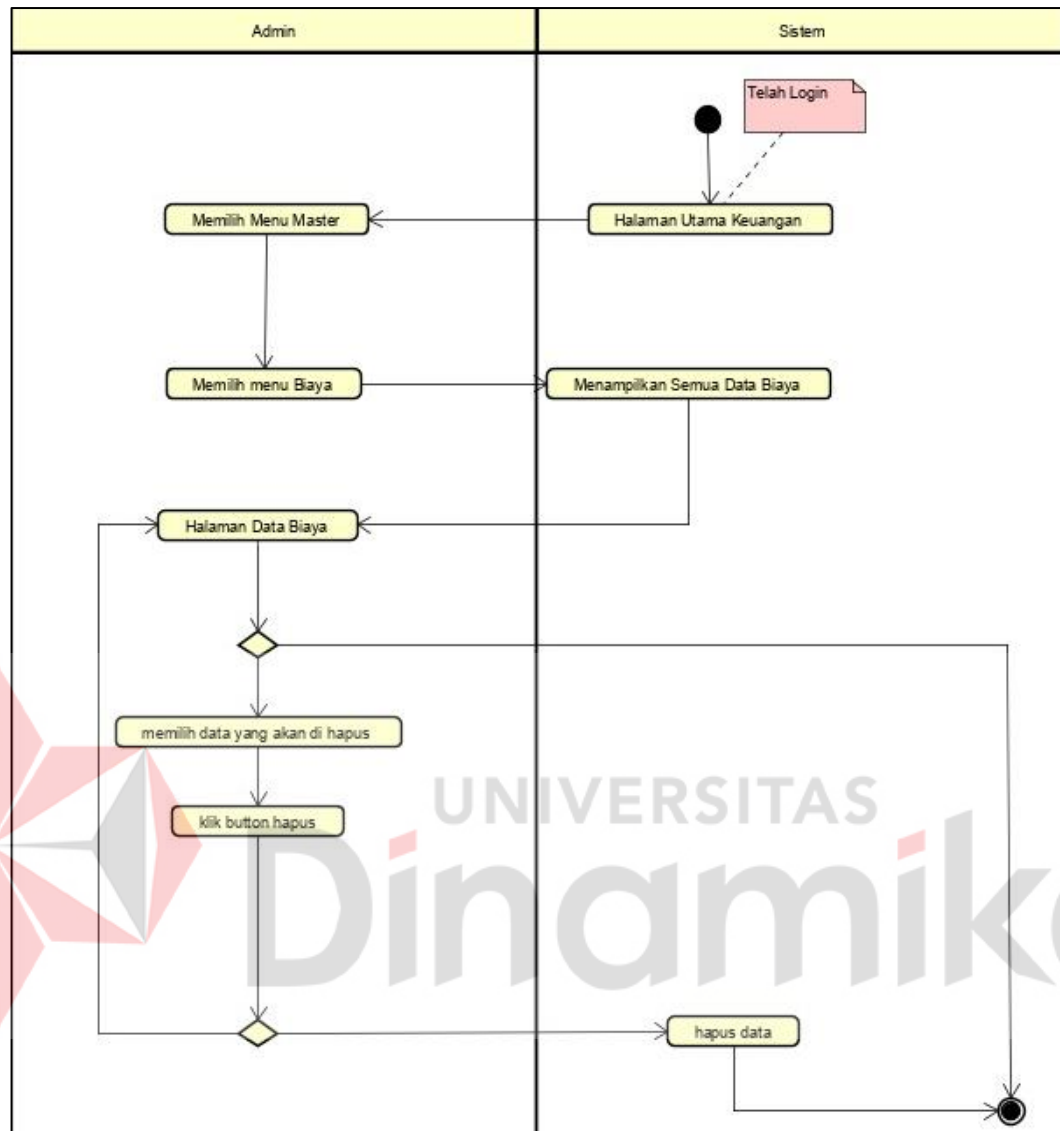
F Activity Diagram Edit Biaya



Gambar 4.7 Activity Diagram Edit Biaya

Pada gambar diatas, Menggambarkan proses aktivitas mengubah data biaya. Diagram tersebut menjelaskan bahwa setelah *admin login* akan masuk ke halaman utama. Untuk mengubah biaya, *admin* diharuskan memilih *menu* biaya dan mengklik *button edit* pada data yang ingin di *update*, setelah itu *admin* diharuskan menubah data yang diinginkan dan menyimpannya.

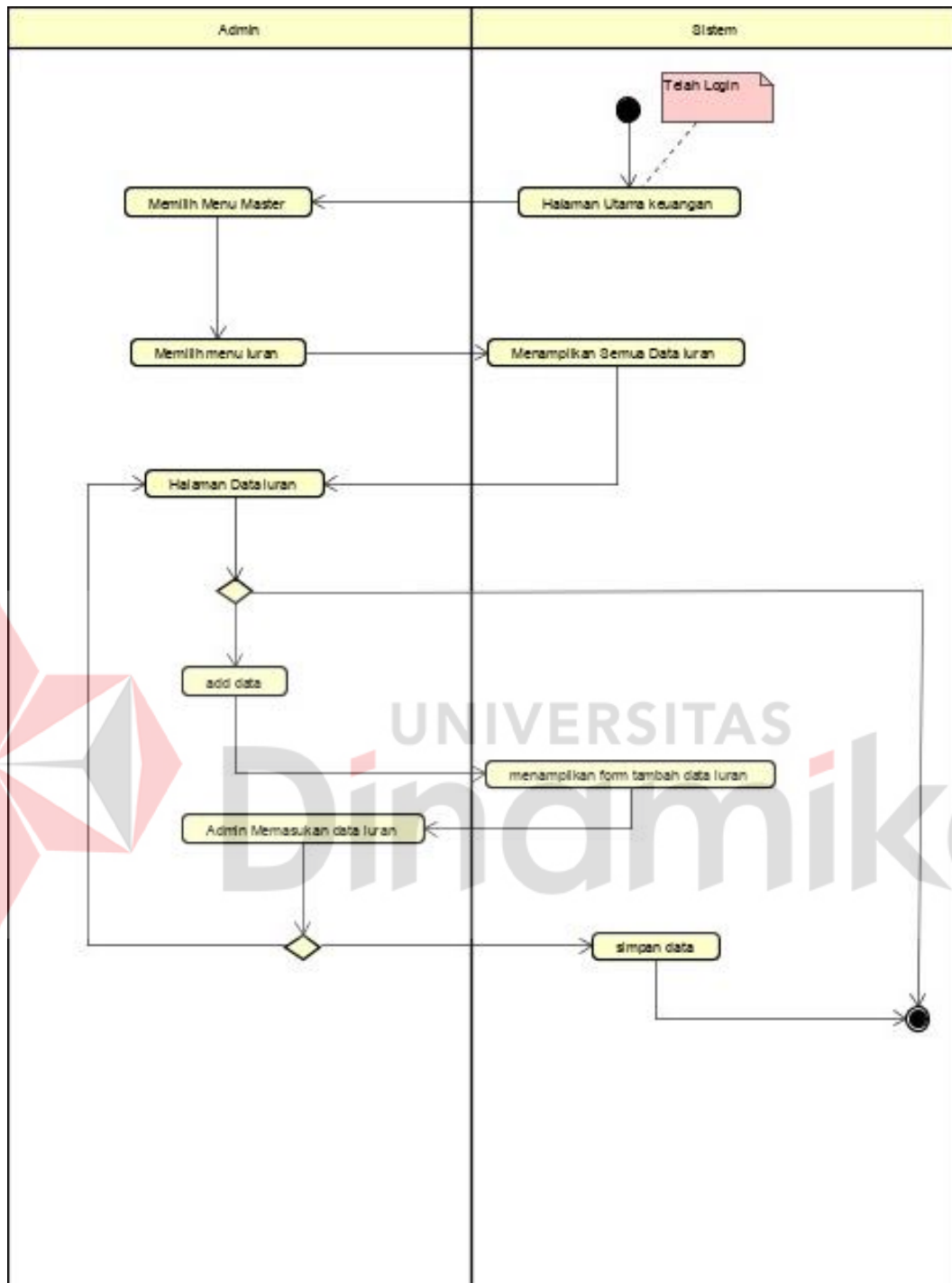
G Activity Diagram Delete Biaya



Gambar 4.8 Activity Diagram Delete Biaya

Pada gambar diatas Menggambarkan proses menghapus data biaya. Diagram tersebut menjelaskan bahwa setelah *admin login* akan masuk ke halaman utama. Setelah itu, *admin* diharuskan memilih *menu biaya* dan mengklik *button hapus* di data yang ingin dihapus, setelah itu *admin* diharuskan mengkonfirmasi penghapusan.

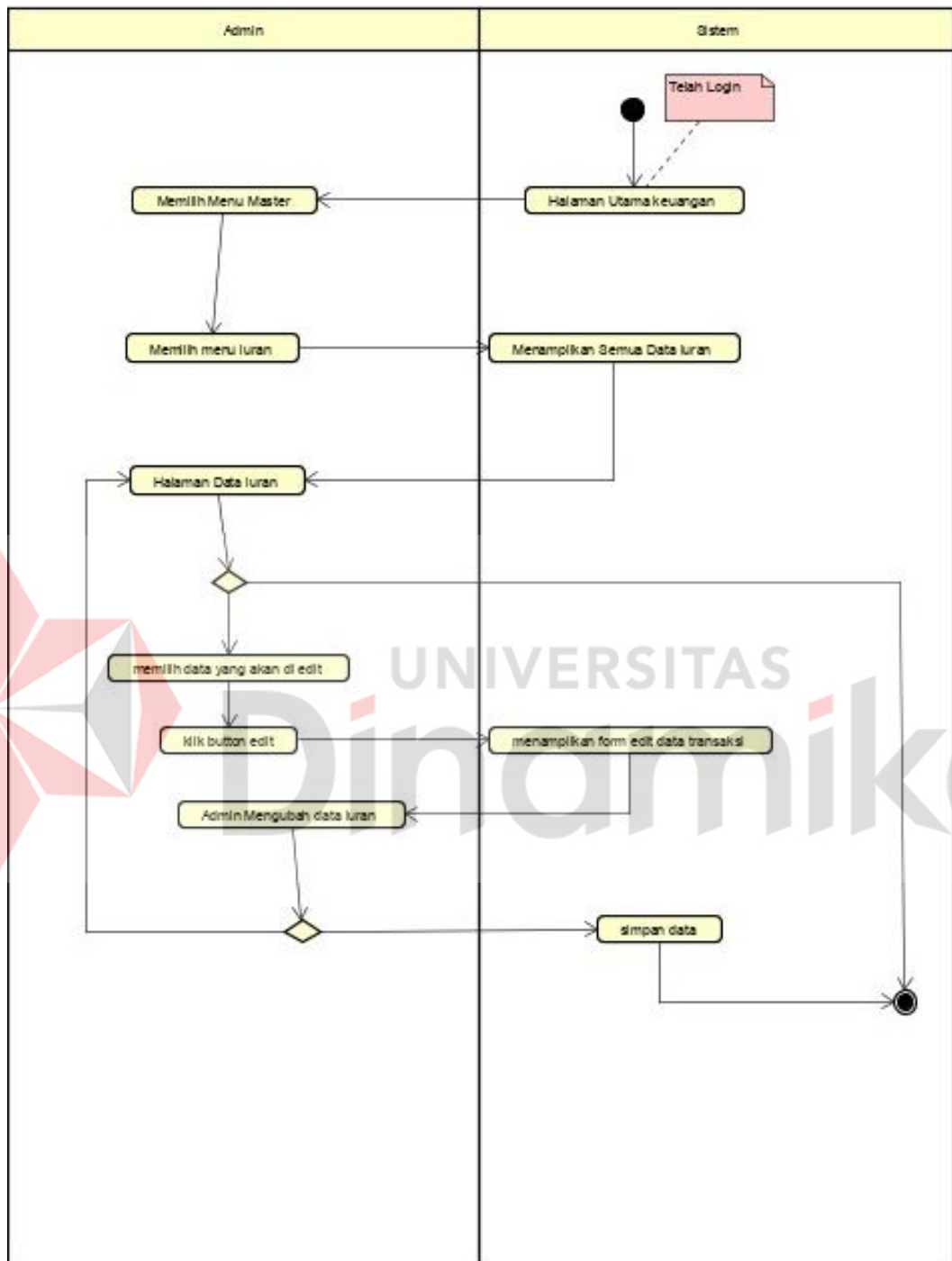
H Activity Diagram Add Iuran



Gambar 4.9 Activity Diagram Add Iuran

Pada gambar diatas, Menggambarkan proses menambah data iuran. Diagram tersebut menjelaskan bahwa setelah *admin login* akan masuk ke halaman utama. Setelah itu, *admin* diharuskan memilih *menu* iuran dan mengklik *button* tambah, setelah itu *admin* diharuskan mengisi *form* dan menyimpannya.

I Activity Diagram Edit Iuran



Gambar 4.10 Activity Diagram Edit Iuran

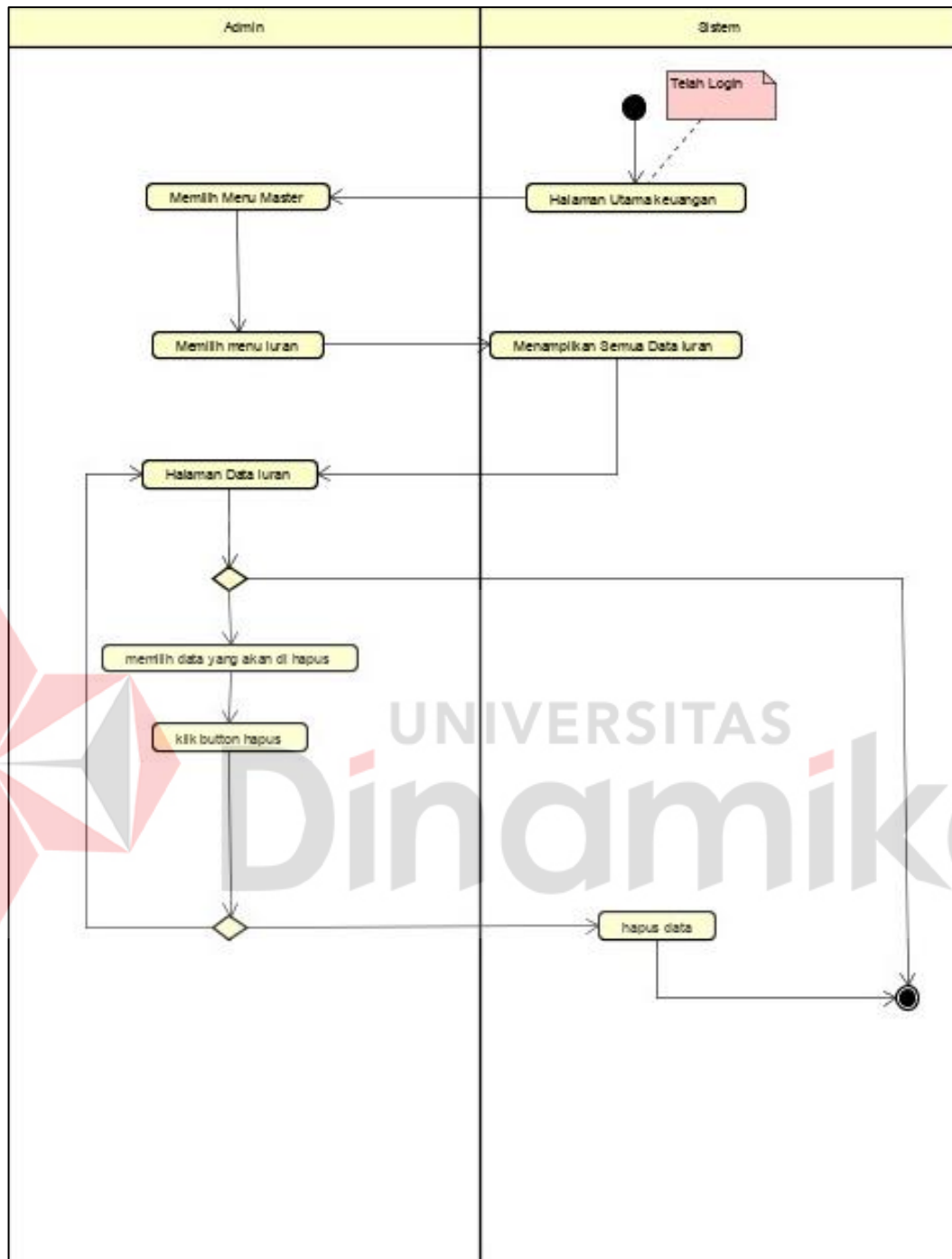
Pada gambar diatas, Menggambarkan proses aktivitas mengubah data iuran. Diagram tersebut menjelaskan bahwa setelah *admin login* akan masuk ke halaman utama. Untuk mengubah iuran, *admin* diharuskan memilih *menu* iuran dan

mengklik *button edit* pada data yang ingin di *update*, setelah itu *admin* diharuskan menubah data yang diinginkan dan menyimpannya.



UNIVERSITAS
Dinamika

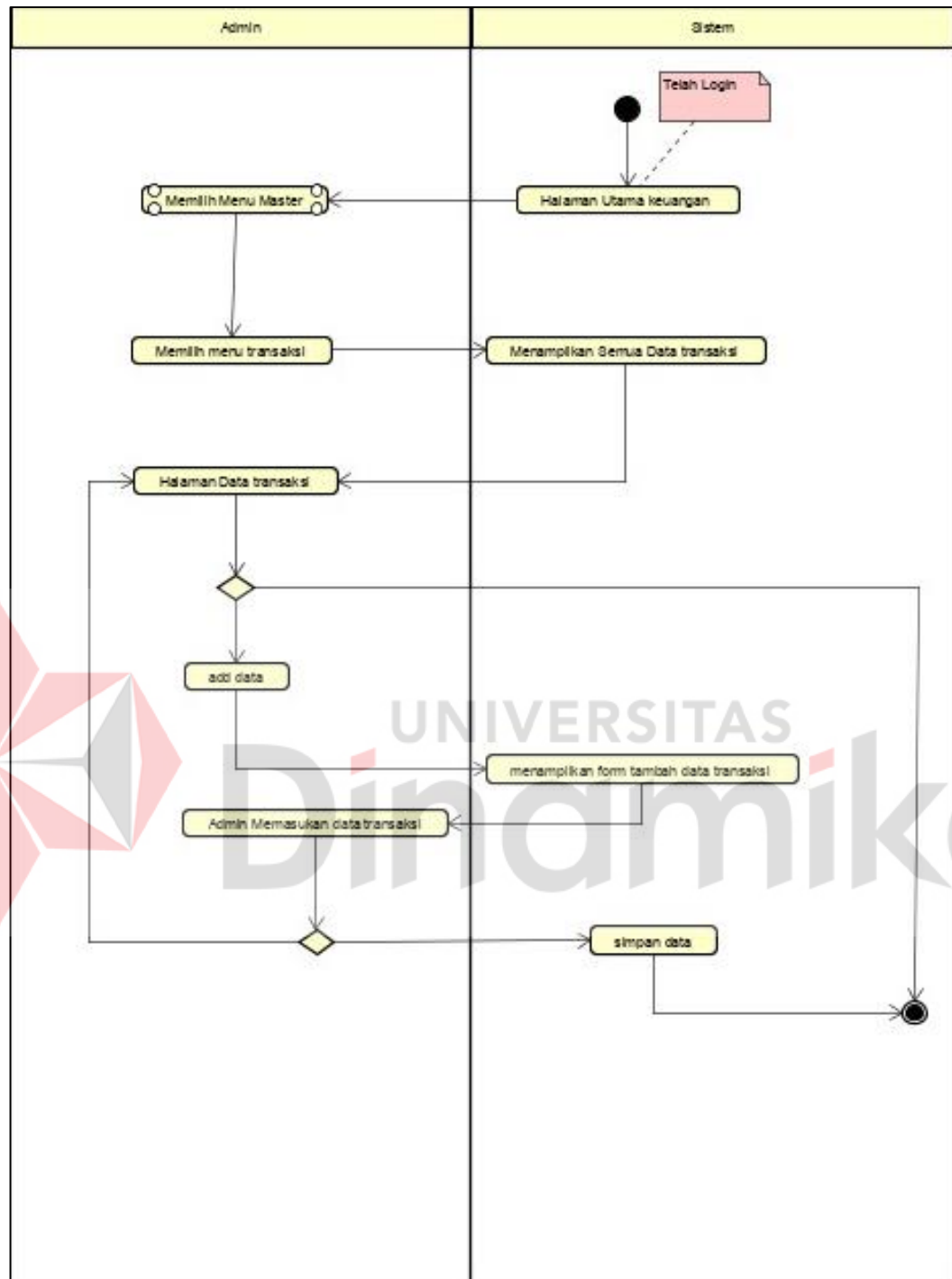
J Activity Diagram Delete Iuran



Gambar 4.11 Activity Diagram Delete Iuran

Pada gambar diatas Menggambarkan proses menghapus data biaya. Diagram tersebut menjelaskan bahwa setelah *admin login* akan masuk ke halaman utama. Setelah itu, *admin* diharuskan memilih *menu iuran* dan mengklik *button hapus* pada data, setelah itu *admin* diharuskan mengkonfirmasi penghapusan.

K Activity Diagram Add Transaksi



Gambar 4.12 Activity Diagram Add Transaksi

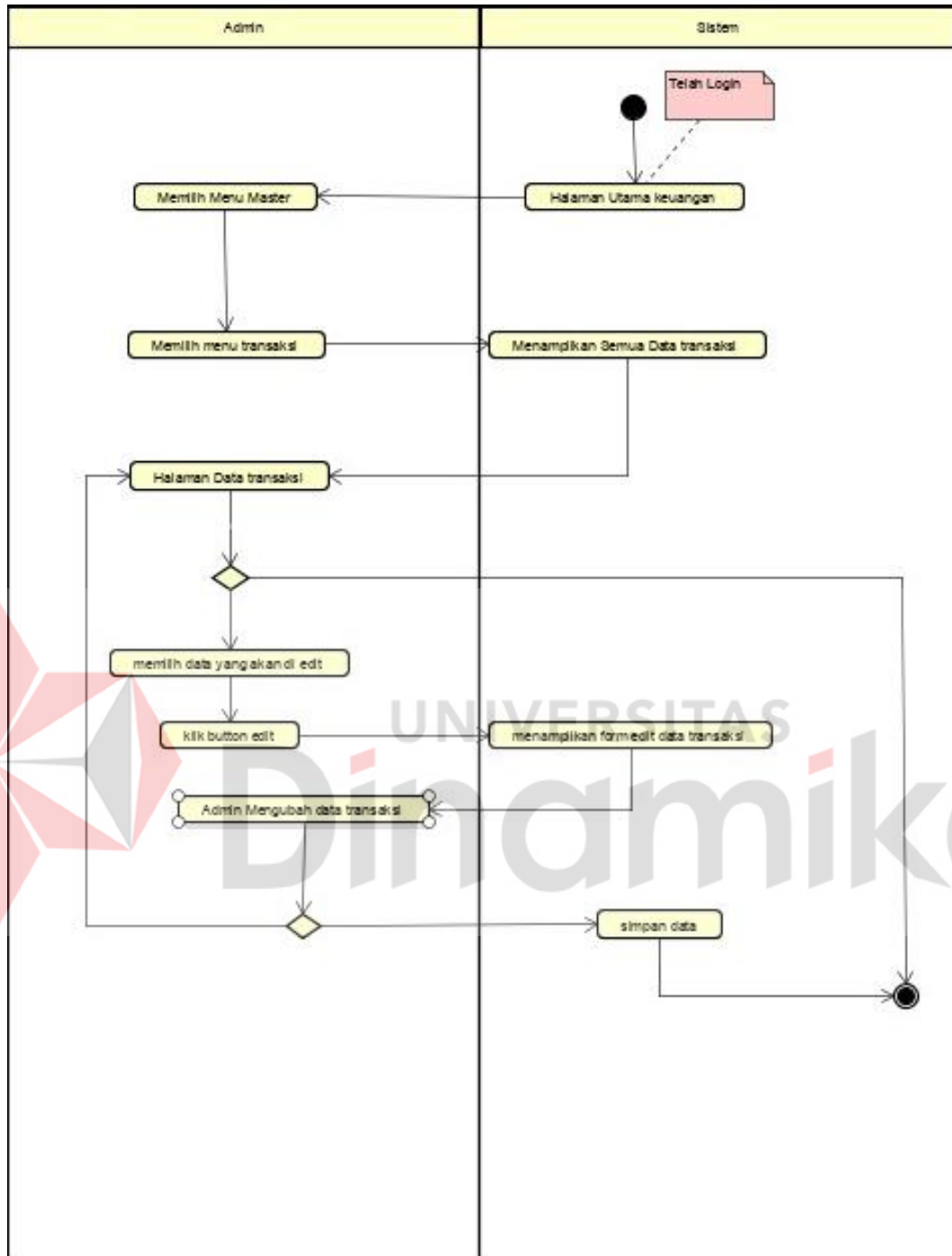
Pada gambar diatas, Menggambarkan proses aktivitas menambah data transaksi. Diagram tersebut menjelaskan bahwa setelah *admin login* akan masuk ke halaman utama. Untuk menambah transaksi, *admin* diharuskan memilih *menu*

transaksi dan mengklik *button* tambah, setelah itu *admin* diharuskan mengisi *form* tambah dan menyimpannya.



UNIVERSITAS
Dinamika

L Activity Diagram Edit Transaksi



Gambar 4.13 Activity Diagram Edit Transaksi

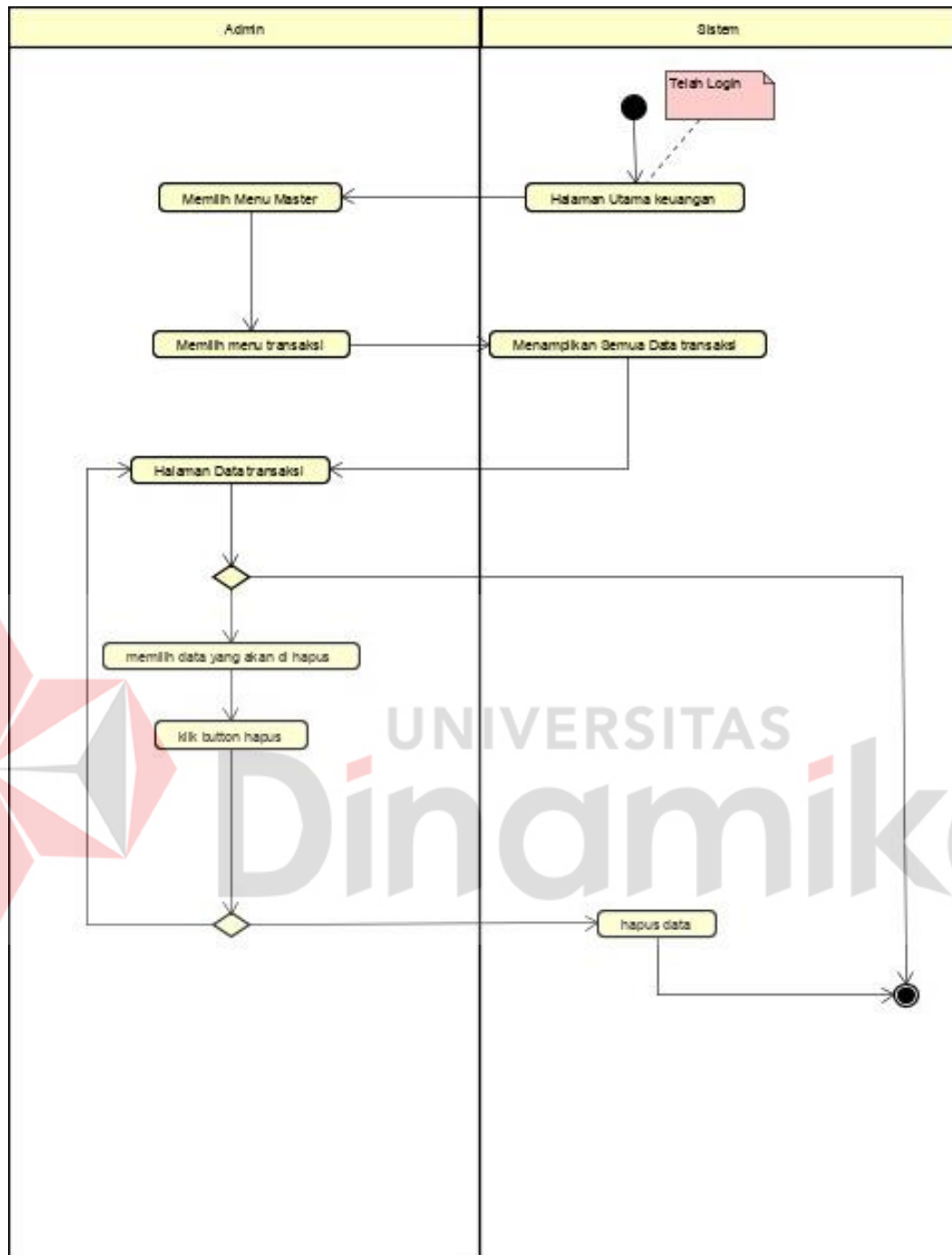
Pada gambar diatas, Menggambarkan proses aktivitas mengubah data transaksi. Diagram tersebut menjelaskan bahwa setelah *admin login* akan masuk ke halaman utama. Untuk mengubah transaksi, *admin* diharuskan memilih *menu* transaksi dan mengklik *button edit* pada data yang ingin di *update*, setelah itu *admin*

diharuskan menubah data yang diinginkan dan menyimpannya.



UNIVERSITAS
Dinamika

M Activity Diagram Delete Transaksi



Gambar 4.14 Activity Diagram Delete Transaksi

Pada gambar diatas Menggambarkan proses aktivitas menghapus data biaya. Diagram tersebut menjelaskan bahwa setelah *admin login* akan masuk ke halaman utama. Untuk menghapus data transaksi, *admin* diharuskan memilih *menu transaksi* dan mengklik *button hapus* di data yang ingin dihapus, setelah itu *admin* diharuskan mengkonfirmasi penghapusan.

4.2.3 Flow Of Event

Berikut merupakan *flow of event* dari aplikasi pencatatan keuangan pada pesantren mahasiswa mahad thaybah

A Flow Of Event Login

Table 4. 1 Flow of Event Login

Nama	Usecase Login	
Deskripsi	Proses ini adalah sebuah kegiatan untuk masuk ke dalam sistem	
Prasyarat	Sudah harus memiliki akun berupa <i>username</i> dan <i>password</i> untuk bisa <i>login</i>	
Post condition	1. <i>Login</i> berhasil 2. Masuk ke halaman utama	
Aktor	User	
Utama	Aksi Aktor	Reaksi Sistem
	1. Aktor membuka aplikasi	2. Sistem menampilkan halaman <i>login</i>
	3. Aktor memasukkan <i>username</i> dan <i>password</i>	4. Sistem mengecek apakah data yang dimasukkan valid (A.4) 5. Sistem mengecek apakah <i>username</i> dan <i>password</i> ada di <i>database</i> master <i>user</i> (A.5) (e.5) 6. Sistem menampilkan menu utama
Alternatif	(A.4) Aktor menginputkan ulang data	(A.4) Jika data tidak valid, maka sistem akan menampilkan pesan <i>error</i> lalu kembali ke menu <i>login</i>
	(A.5) Aktor menginputkan ulang data	(A.5) <i>username</i> dan <i>password</i> tidak terdaftar yang dimana sistem akan menampilkan pesan <i>error</i> dan kemudian kembali ke menu <i>login</i>

Kesalahan	(E.5) Terjadi jaringan <i>down</i> sehingga <i>database</i> tidak bisa diakses
Kondisi	Sukses : Aktor masuk kehalaman utama
Akhir	Gagal : Aktor tidak dapat masuk ke halaman utama <ul style="list-style-type: none"> 1) Data yang dimasukkan tidak valid 2) <i>Username / password</i> tidak terdaftar

Tabel 4.1 *Flow of event* ini menjelaskan bagaimana alur login pada aplikasi pencatatan keuangan.

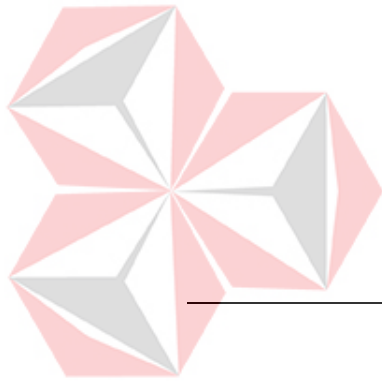


UNIVERSITAS
Dinamika

B *Flow of Event* Pengelolaan Master Kategori Keuangan

Table 4-1 *Flow of Event Insert* Master Kategori Keuangan

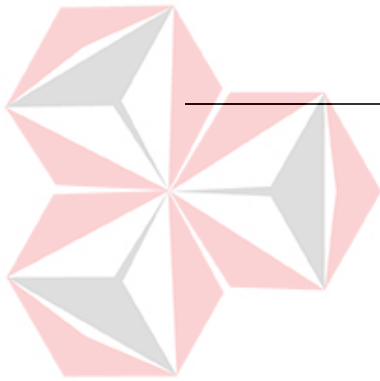
Deskripsi	<i>Usecase</i> Pengelolaan Master Kategori Keuangan	
Precondition	<i>Usecase Login</i>	
Aktor	Admin	
Alur	Aksi Aktor	Aksi Sistem
Utama	Pengelolaan <i>Insert</i>	
	1. Aktor memilih tampilan master kategori keuangan.	2. Sistem menampilkan data-data kategori keuangan.
	3. Aktor mengklik tombol tambah data.	6. Sistem menyimpan data.
	4. Aktor memasukkan data kategori keuangan.	7. Sistem menampilkan kembali data-data kategori keuangan. (E.6)
	5. Aktor mengklik tombol simpan.	



UNIVERSITAS
Dinamika

Table 4-2 *Flow of Event Update* Master Kategori Keuangan

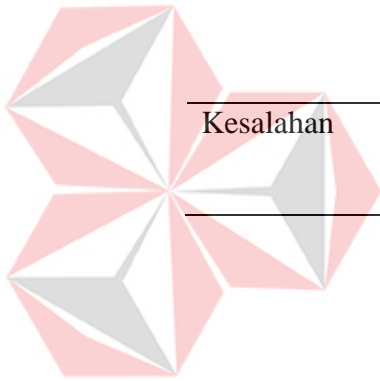
Pengelolaan <i>Update</i>	
1. Aktor memilih tampilan master kategori keuangan.	2. Sistem menampilkan data-data kategori keuangan.
3. Aktor memilih data kategori keuangan yang akan dirubah.	4. Sistem menampilkan data kategori keuangan yang dipilih aktor dan dapat dirubah.
5. Aktor merubah data.	7. Sistem menyimpan perubahan data.
6. Aktor mengklik tombol simpan.	8. Sistem menampilkan kembali data-data kategori keuangan.
(E.12)	



UNIVERSITAS
Dinamika

Table 4-3 *Flow of Event Delete Master Transaksi*

Pengelolaan <i>Delete</i>			
	1. Aktor memilih tampilan master kategori keuangan	2. Sistem menampilkan data-data kategori keuangan	
	3. Aktor memilih data kategori keuangan yang akan dihapus.	4. Sistem menampilkan notifikasi konfirmasi untuk penghapusan.	
	5. Aktor mengkonfirmasi penghapusan.	6. Sistem menghapus data. 7. Sistem menampilkan kembali data-data kategori keuangan.	
Alternative	(A.I.4) Aktor tidak mengkonfirmasi penghapusan.	(A.I.4) Sistem batal menghapus, maka data tidak jadi terhapus.	
Kesalahan		(E.6) Sistem <i>down</i> tidak bisa mengakses <i>database</i> .	

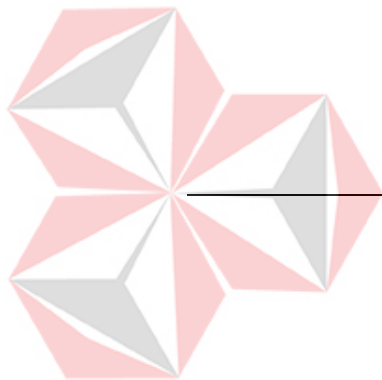


UNIVERSITAS
Dinamika

C *Flow of Event* Pengelolaan Master Biaya

Table 4-4 *Flow of Event Insert* Master Biaya

Deskripsi	<i>Usecase</i> Pengelolaan Master Biaya	
Precondition	<i>Usecase Login</i>	
Aktor	Admin	
Alur	Aksi Aktor	Aksi Sistem
Utama	Pengelolaan <i>Insert</i>	
	1. Aktor memilih tampilan master biaya.	2. Sistem menampilkan data-data biaya.
	3. Aktor mengklik tombol tambah data.	6. Sistem menyimpan data.
	4. Aktor memasukkan data biaya.	7. Sistem menampilkan kembali tampilan data-data biaya. (E.6)
	5. Aktor mengklik tombol simpan.	



UNIVERSITAS
Dinamika

Table 4-5 *Flow of Event Update Master Biaya*

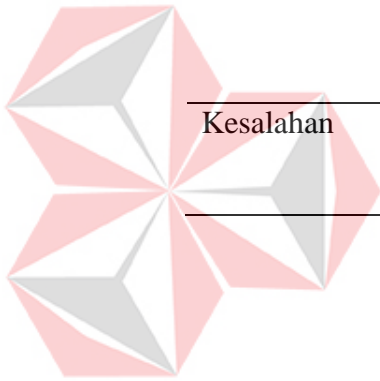
Pengelolaan <i>Update</i>	
1. Aktor memilih tampilan master biaya.	2. Sistem menampilkan data-data biaya.
3. Aktor memilih data biaya yang akan dirubah.	4. Sistem menampilkan data biaya yang dipilih aktor dan dapat dirubah.
5. Aktor merubah data	7. Sistem menyimpan perubahan data.
6. Aktor mengklik tombol simpan.	8. Sistem menampilkan kembali data-data biaya. (E.12)



UNIVERSITAS
Dinamika

Table 4-6 *Flow of Event Delete Master Biaya*

Pengelolaan <i>Delete</i>			
	1. Aktor memilih tampilan master biaya.	2. Sistem menampilkan data-data biaya.	
	3. Aktor memilih data biaya yang akan dihapus.	4. Sistem menampilkan notifikasi konfirmasi untuk penghapusan.	
	5. Aktor mengkonfirmasi penghapusan	6. Sistem menghapus data. 7. Sistem menampilkan kembali data-data biaya.	
Alternative	(A.I.4) Aktor tidak mengkonfirmasi penghapusan.	(A.I.4) Sistem batal menghapus, maka data tidak jadi terhapus.	
Kesalahan		(E.6) Sistem <i>down</i> tidak bisa mengakses <i>database</i> .	

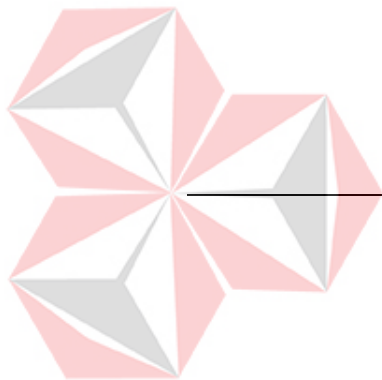


UNIVERSITAS
Dindamika

D *Flow of Event* Pengelolaan Iuran

Table 4-7 *Flow of Event Insert* Iuran

Deskripsi	<i>Usecase</i> Pengelolaan Iuran	
Precondition	<i>Usecase Login</i>	
Aktor	Admin	
Alur	Aksi Aktor	Aksi Sistem
Utama	Pengelolaan <i>Insert</i>	
	1. Aktor memilih tampilan iuran.	2. Sistem menampilkan data-data iuran.
	3. Aktor mengklik tombol tambah data.	6. Sistem menyimpan data.
	4. Aktor menginputkan data iuran.	7. Sistem menampilkan kembali data-data iuran. (E.6)
	5. Aktor mengklik tombol simpan.	



UNIVERSITAS
Dinamika

Table 4-8 *Flow of Event Update Iuran*

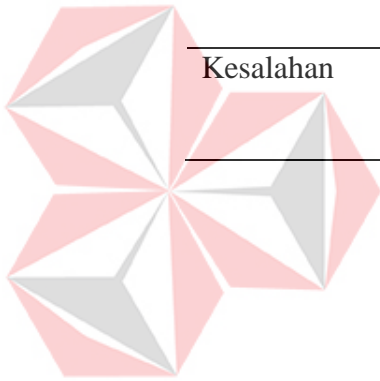
Pengelolaan <i>Update</i>	
1. Aktor memilih tampilan iuran.	2. Sistem menampilkan data-data iuran.
3. Aktor memilih data iuran yang akan dirubah.	4. Sistem menampilkan data iuran yang dipilih aktor dan dapat dirubah.
5. Aktor merubah data.	7. Sistem menyimpan data.
6. Aktor mengklik tombol simpan	8. Sistem menampilkan kembali data-data iuran. (E.12)



UNIVERSITAS
Dinamika

Table 4-9 *Flow of Event Delete Iuran*

Pengelolaan <i>Delete</i>			
	1. Aktor memilih tampilan iuran.	2. Sistem menampilkan data-data iuran.	
	3. Aktor memilih data iuran yang akan dihapus.	4. Sistem menampilkan notifikasi konfirmasi untuk penghapusan.	
	5. Aktor mengkonfirmasi penghapusan.	6. Sistem menghapus data. 7. Sistem menampilkan kembali data-data iuran.	
Alternative	(A.I.4) Aktor tidak mengkonfirmasi penghapusan.	(A.I.4) Sistem batal menghapus, maka data tidak jadi terhapus.	
Kesalahan		(E.6) Sistem <i>down</i> tidak bisa mengakses <i>database</i> .	

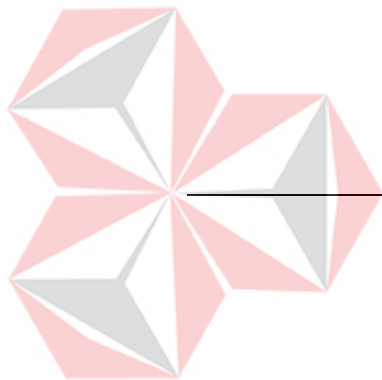


UNIVERSITAS
Dinamika

E *Flow of Event* Pengelolaan Transaksi

Table 4-10 *Flow of Event Insert* Transaksi

Deskripsi	<i>Usecase</i> Pengelolaan Transaksi	
Precondition	<i>Usecase Login</i>	
Aktor	Admin	
Alur	Aksi Aktor	Aksi Sistem
Utama	Pengelolaan Insert	
	1. Aktor memilih tampilan transaksi.	2. Sistem menampilkan data-data transaksi.
	3. Aktor memilih tombol tambah data.	6. Sistem menyimpan data.
	4. Aktor menginputkan data transaksi.	7. Sistem menampilkan kembali data-data transaksi. (E.6)
	5. Aktor mengklik tombol simpan.	



UNIVERSITAS
Dinamika

Table 4-11 *Flow of Event Update Transaksi*

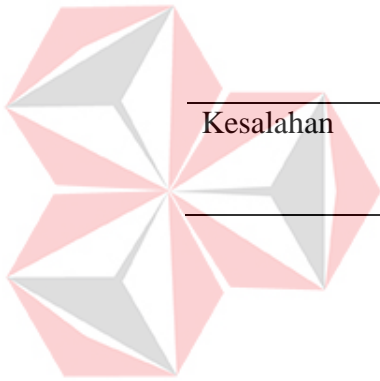
Pengelolaan <i>Update</i>	
1. Aktor memilih tampilan transaksi.	2. Sistem menampilkan data-data transaksi.
3. Aktor memilih data transaksi yang akan dirubah.	4. Sistem menampilkan data iuran yang dipilih aktor dan dapat dirubah.
5. Aktor merubah data.	7. Sistem menyimpan data.
6. Aktor mengklik tombol simpan.	8. Sistem menampilkan kembali data-data transaksi. (E.12)



UNIVERSITAS
Dinamika

Table 4-12 *Flow of Event Delete* Transaksi

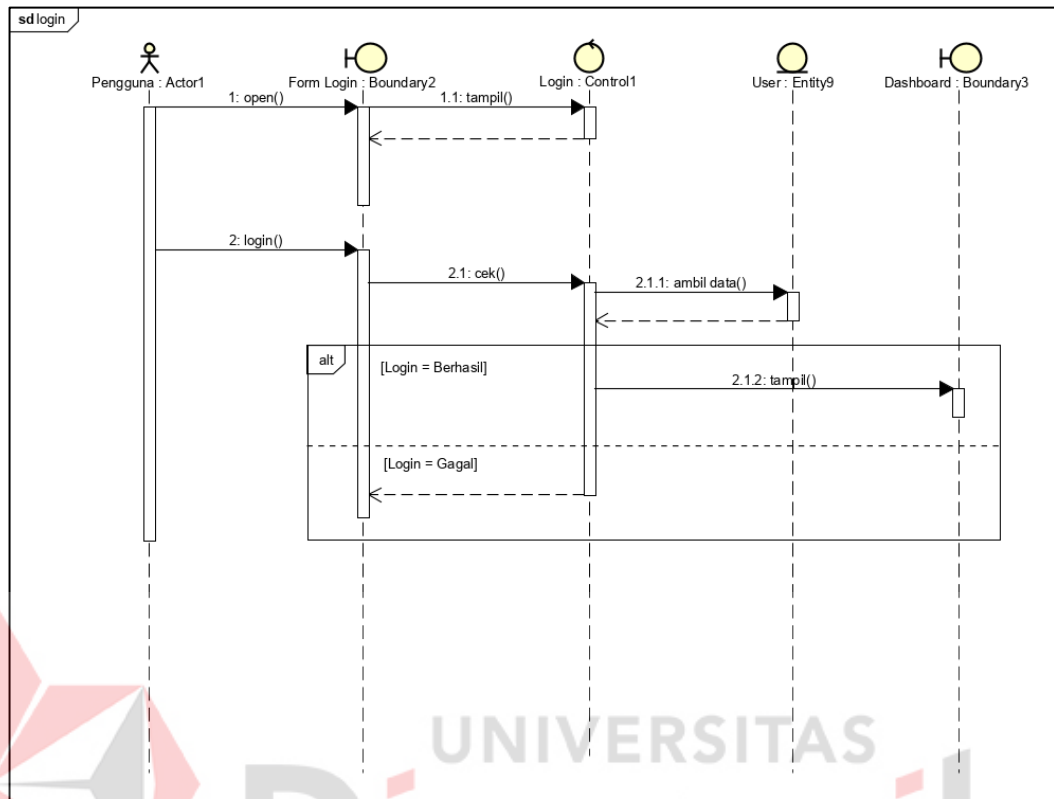
Pengelolaan <i>Delete</i>			
	1. Aktor memilih tampilan transaksi.	2. Sistem menampilkan data-data transaksi.	
	3. Aktor memilih data transaksi yang akan dihapus.	4. Sistem menampilkan notifikasi konfirmasi untuk penghapusan.	
	5. Aktor mengkonfirmasi penghapusan.	6. Sistem menghapus data. 7. Sistem menampilkan kembali data-data transaksi.	
Alternative	(A.I.4) Aktor tidak mengkonfirmasi penghapusan	(A.I.4) Sistem batal menghapus, maka data tidak jadi terhapus	
Kesalahan		(E.6) Sistem <i>down</i> tidak bisa mengakses <i>database</i>	



UNIVERSITAS
Dindamika

4.2.4 Sequence Diagram

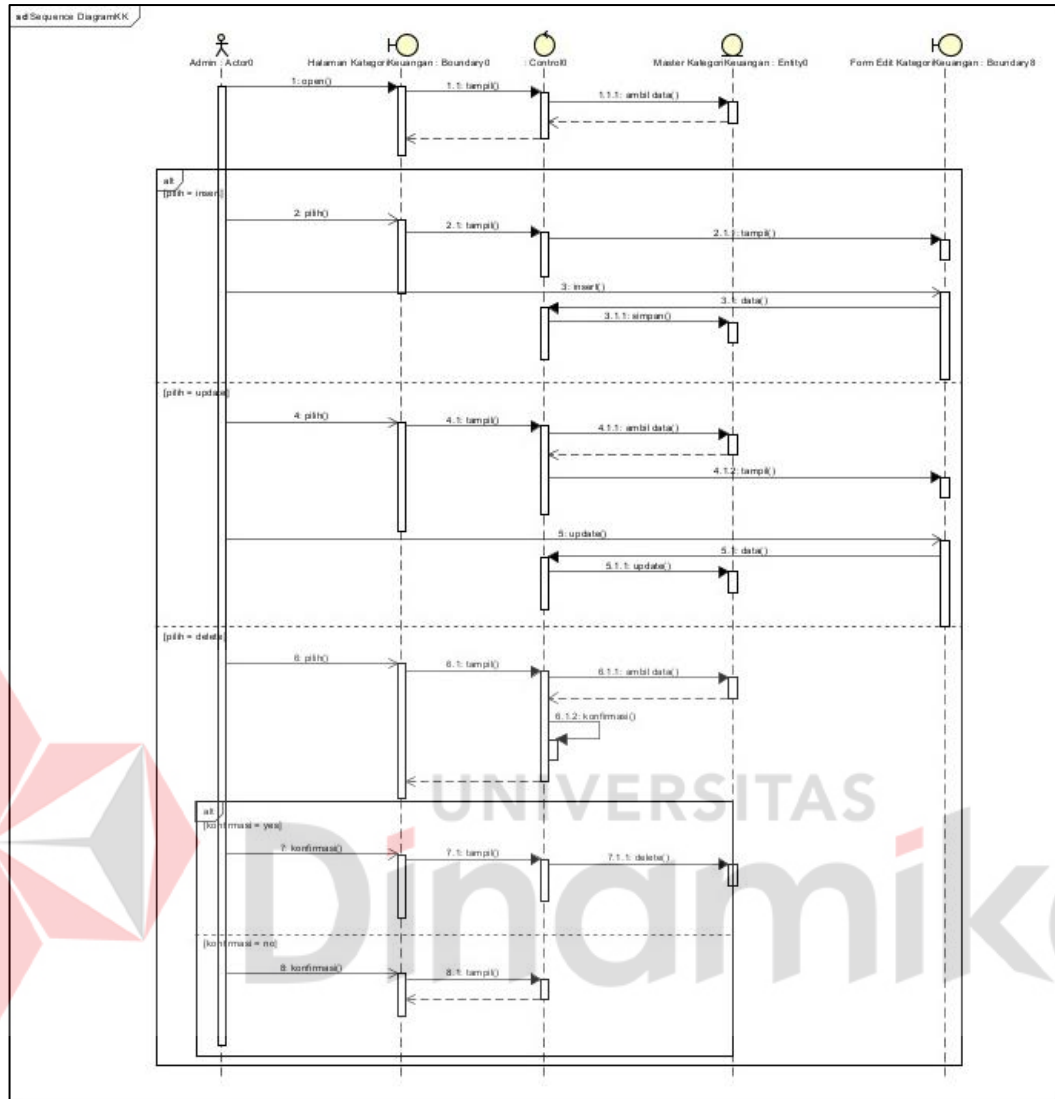
A Sequence Diagram Login



Gambar 4.15 Sequence Diagram Login

Pengguna mengakses *form login*, *form login* mengirim *request* tampil pada *controller*, setelah itu *controller* menampilkan halaman *form login* pada pengguna tersebut. Pengguna melakukan *login* dengan memasukkan *username* serta *password*, setelah itu *controller* melakukan pengecekan *username* dan *password* dari inputan pengguna tersebut dengan data yang diambil dari tabel *User*. Jika *login* berhasil maka *controller* akan menampilkan halaman *dashboard* pada pengguna, jika *login* gagal maka *controller* menampilkan kembali halaman *form login* dengan pesan *error*.

B Sequence Diagram Kategori Keuangan



Gambar 4.16 Sequence Diagram Kategori Keuangan

Pengguna mengakses halaman kategori keuangan, halaman kategori keuangan mengirim *request* tampil pada *controller*, setelah itu *controller* mengambil data kategori keuangan dari tabel Master KategoriKeuangan dan menampilkan halaman kategori keuangan dengan data kategori keuangan yang berbentuk tabel pada pengguna tersebut.

Jika pengguna melakukan *insert* data maka *controller* akan menampilkan *form edit* kategori keuangan yang berbentuk sebagai *pop up* untuk menambahkan data, setelah itu pengguna memasukkan data-data kategori keuangan dan menyimpan data tersebut, maka *controller* akan menyimpan data-data yang telah

dimasukkan dari pengguna tersebut ke tabel Master KategoriKeuangan.

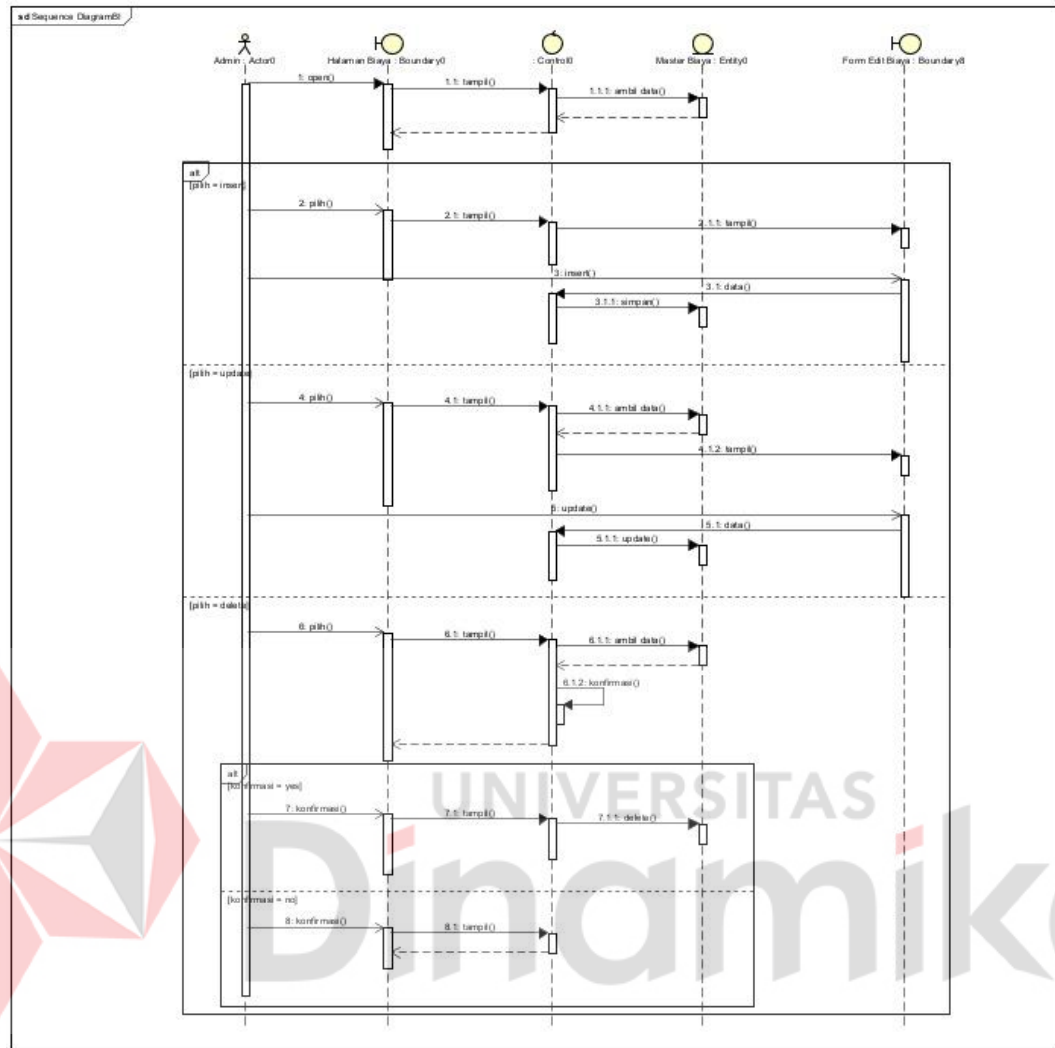
Jika pengguna melakukan *update* data maka *controller* akan menampilkan *form edit* kategori keuangan beserta data-data yang akan dirubah yang berbentuk sebagai *pop up* untuk mengubah data, setelah itu pengguna merubah data-data kategori keuangan dan menyimpan data tersebut, maka *controller* akan menyimpan perubahan data-data yang telah dimasukkan dari pengguna tersebut ke tabel Master KategoriKeuangan.

Jika pengguna melakukan *delete* data maka *controller* akan menampilkan konfirmasi penghapusan data kategori keuangan tersebut yang berbentuk sebagai *pop up*, jika pengguna mengkonfirmasi iya maka *controller* akan menghapus data kategori keuangan tersebut dari tabel Master KategoriKeuangan dan menampilkan kembali halaman kategori keuangan, jika pengguna mengkonfirmasi tidak maka *pop up* konfirmasi penghapusan tersebut ditutup.



UNIVERSITAS
Dinamika

C Sequence Diagram Biaya



Gambar 4.17 Sequence Diagram Biaya

Pengguna mengakses halaman biaya, halaman biaya mengirim *request* tampil pada *controller*, setelah itu *controller* mengambil data biaya dari tabel Master Biaya dan menampilkan halaman biaya dengan data biaya yang berbentuk tabel pada pengguna tersebut.

Jika pengguna melakukan *insert* data maka *controller* akan menampilkan *form edit* biaya yang berbentuk sebagai *pop up* untuk menambahkan data, setelah itu pengguna memasukkan data-data biaya dan menyimpan data tersebut, maka *controller* akan menyimpan data-data yang telah dimasukkan dari pengguna tersebut ke tabel Master Biaya.

Jika pengguna melakukan *update* data maka *controller* akan menampilkan

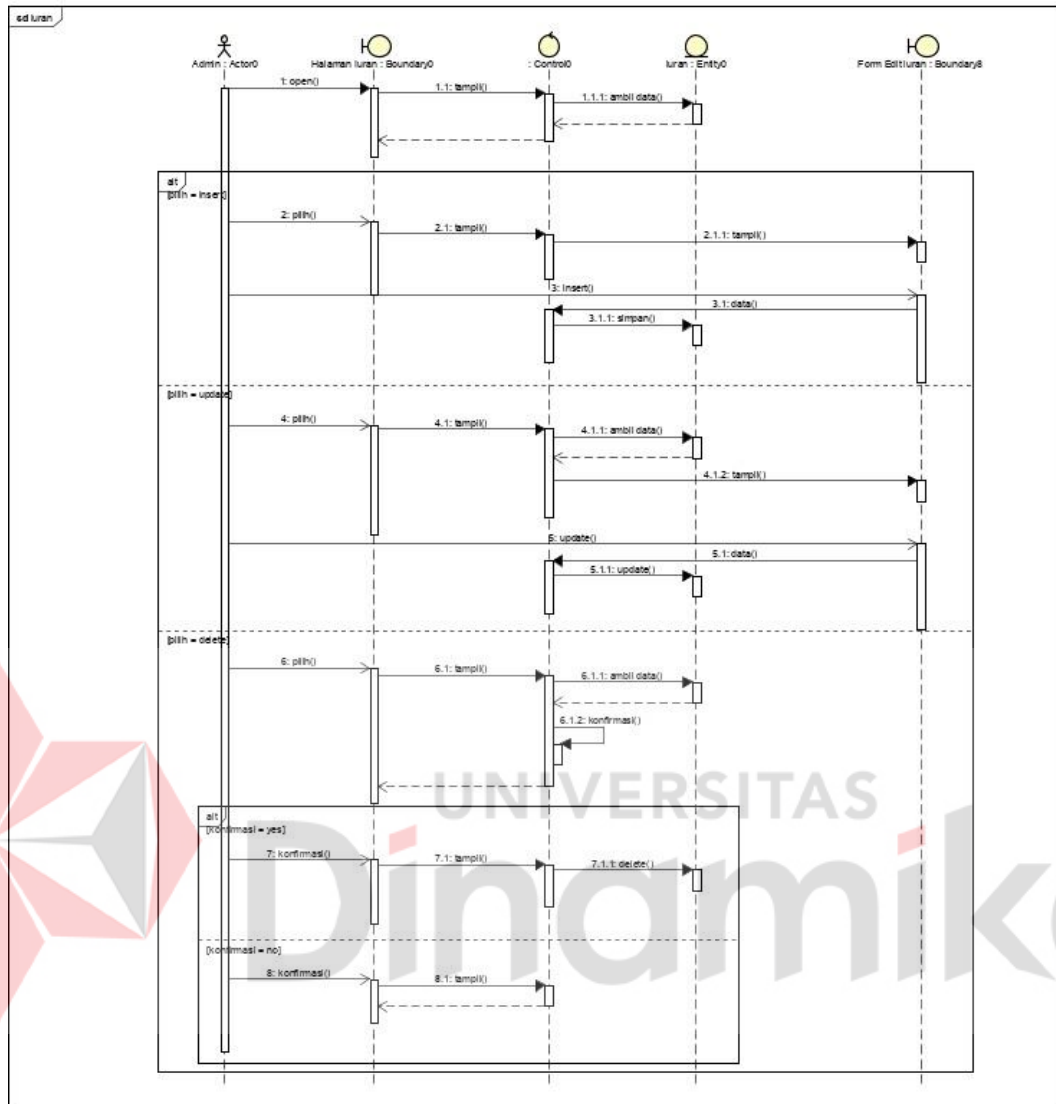
form edit biaya beserta data-data yang akan dirubah yang berbentuk sebagai *pop up* untuk mengubah data, setelah itu pengguna merubah data-data biaya dan menyimpan data tersebut, maka *controller* akan menyimpan perubahan data-data yang telah dimasukkan dari pengguna tersebut ke tabel Master Biaya.

Jika pengguna melakukan *delete* data maka *controller* akan menampilkan konfirmasi penghapusan data biaya tersebut yang berbentuk sebagai *pop up*, jika pengguna mengkonfirmasi iya maka *controller* akan menghapus data biaya tersebut dari tabel Master Biaya dan menampilkan kembali halaman biaya, jika pengguna mengkonfirmasi tidak maka *pop up* konfirmasi penghapusan tersebut ditutup.



UNIVERSITAS
Dinamika

D Sequence Diagram Iuran



Gambar 4.18 Sequence Diagram Iuran

Pengguna mengakses halaman iuran, halaman iuran mengirim *request* tampil pada *controller*, setelah itu *controller* mengambil data iuran dari tabel Iuran dan menampilkan halaman iuran dengan data iuran yang berbentuk tabel pada pengguna tersebut.

Jika pengguna melakukan *insert* data maka *controller* akan menampilkan halaman *form edit* iuran untuk menambahkan data, setelah itu pengguna memasukkan data-data iuran dan menyimpan data tersebut, maka *controller* akan menyimpan data-data yang telah dimasukkan dari pengguna tersebut ke tabel Iuran.

Jika pengguna melakukan *update* data maka *controller* akan menampilkan

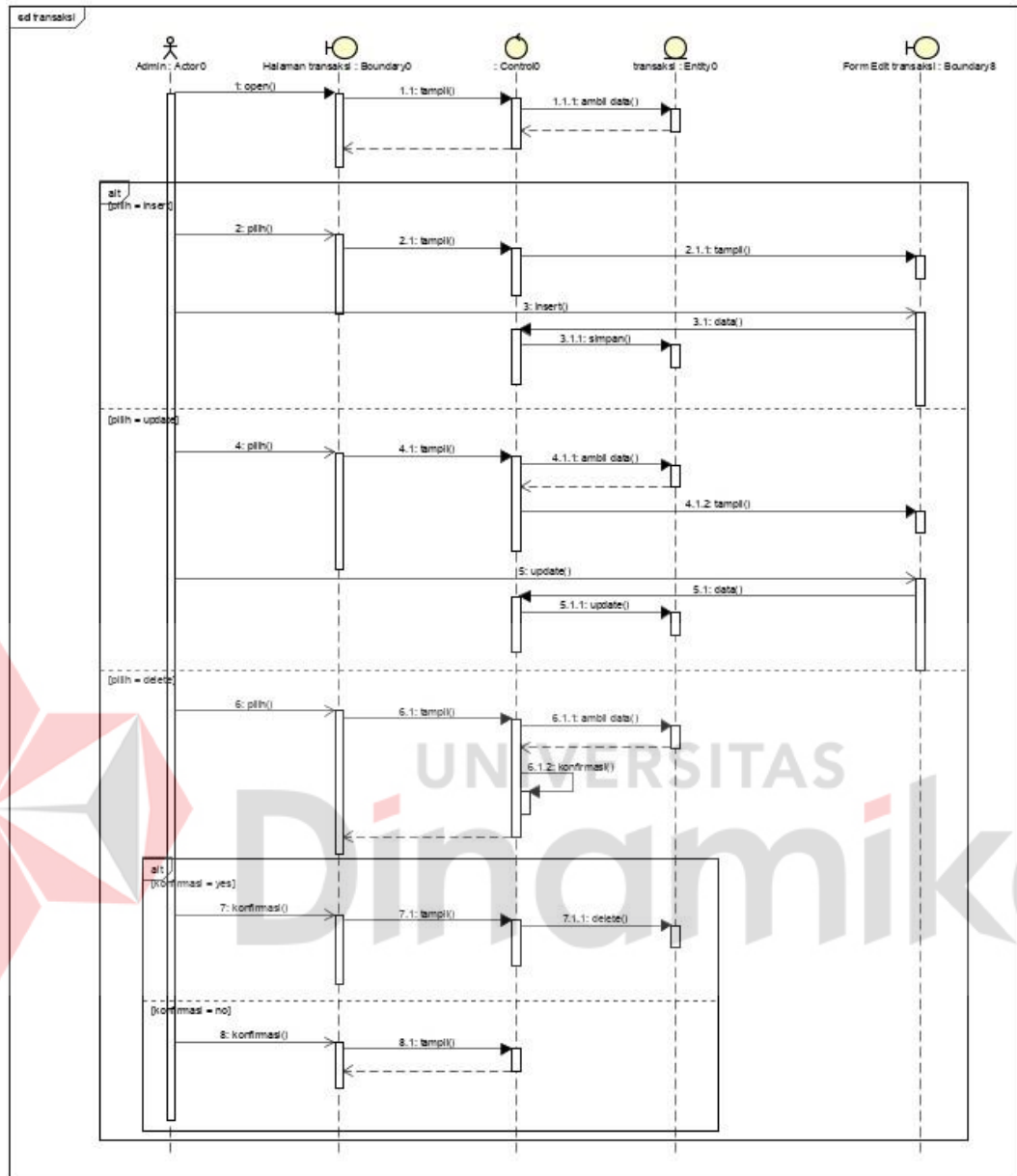
halaman *form edit* iuran beserta data-data yang akan dirubah untuk mengubah data, setelah itu pengguna merubah data-data iuran dan menyimpan data tersebut, maka *controller* akan menyimpan perubahan data-data yang telah dimasukkan dari pengguna tersebut ke tabel Iuran.

Jika pengguna melakukan *delete* data maka *controller* akan menampilkan konfirmasi penghapusan data iuran tersebut yang berbentuk sebagai *pop up*, jika pengguna mengkonfirmasi iya maka *controller* akan menghapus data iuran tersebut dari tabel Iuran dan menampilkan kembali halaman iuran, jika pengguna mengkonfirmasi tidak maka *pop up* konfirmasi penghapusan tersebut ditutup.



UNIVERSITAS
Dinamika

E Sequence Diagram Transaksi



Gambar 4.19 Sequence Diagram Transaksi

Pengguna mengakses halaman transaksi, halaman transaksi mengirim *request* tampil pada *controller*, setelah itu *controller* mengambil data transaksi dari tabel Transaksi dan menampilkan halaman transaksi dengan data transaksi yang berbentuk tabel pada pengguna tersebut.

Jika pengguna melakukan *insert* data maka *controller* akan menampilkan halaman *form edit* transaksi untuk menambahkan data, setelah itu pengguna memasukkan data-data transaksi dan menyimpan data tersebut, maka *controller*

akan menyimpan data-data yang telah dimasukkan dari pengguna tersebut ke tabel Transaksi.

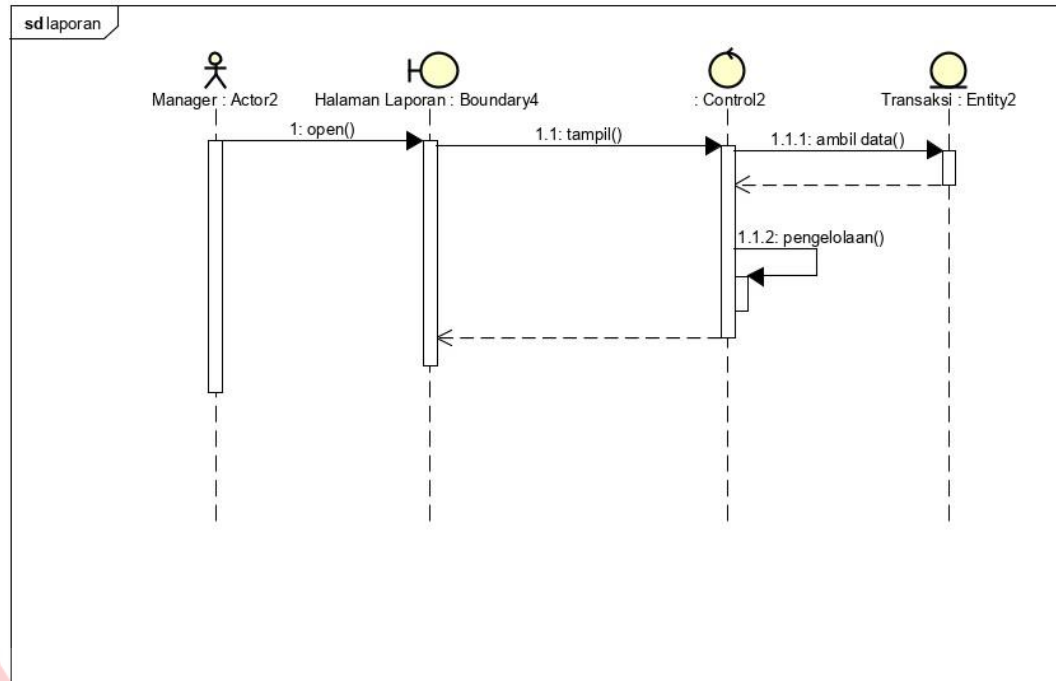
Jika pengguna melakukan *update* data maka *controller* akan menampilkan halaman *form edit* transaksi beserta data-data yang akan dirubah untuk mengubah data, setelah itu pengguna merubah data-data transaksi dan menyimpan data tersebut, maka *controller* akan menyimpan perubahan data-data yang telah dimasukkan dari pengguna tersebut ke tabel Transaksi.

Jika pengguna melakukan *delete* data maka *controller* akan menampilkan konfirmasi penghapusan data transaksi tersebut yang berbentuk sebagai *pop up*, jika pengguna mengkonfirmasi iya maka *controller* akan menghapus data transaksi tersebut dari tabel Transaksi dan menampilkan kembali halaman transaksi, jika pengguna mengkonfirmasi tidak maka *pop up* konfirmasi penghapusan tersebut ditutup.



UNIVERSITAS
Dinamika

F Sequence Diagram Laporan

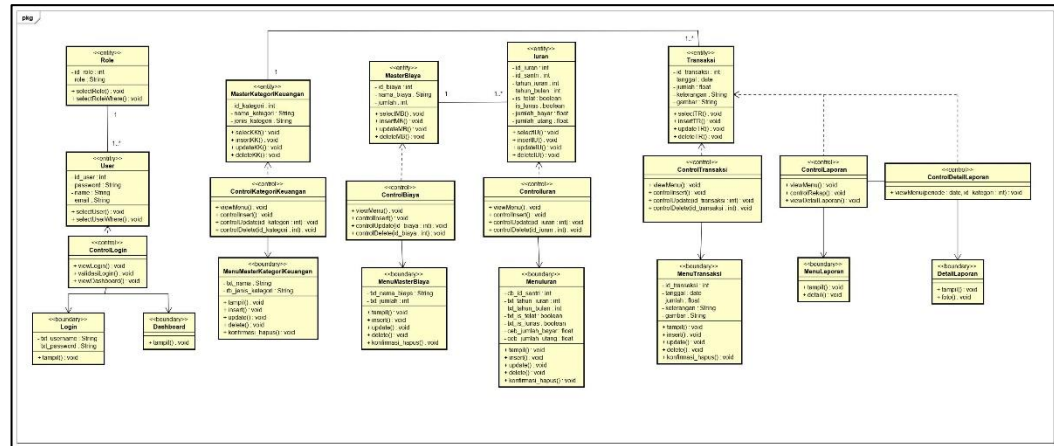


Gambar 4.20 Sequence Diagram Laporan

Pengguna mengakses halaman laporan, halaman laporan mengirim *request* tampil pada *controller*, setelah itu *controller* mengambil data transaksi dari tabel Transaksi dan dilakukan pengolahan data yaitu perekapan semua transaksi berdasarkan periode bulan dan tahun serta berdasarkan kategori keuangan, setelah itu *controller* menampilkan halaman laporan dengan transaksi yang telah direkap berbentuk tabel pada pengguna tersebut.

4.2.5 Class Diagram

Berikut merupakan *Class Diagram* dari modul keuangan aplikasi pada pesantren mahasiswa mahad thaybah.

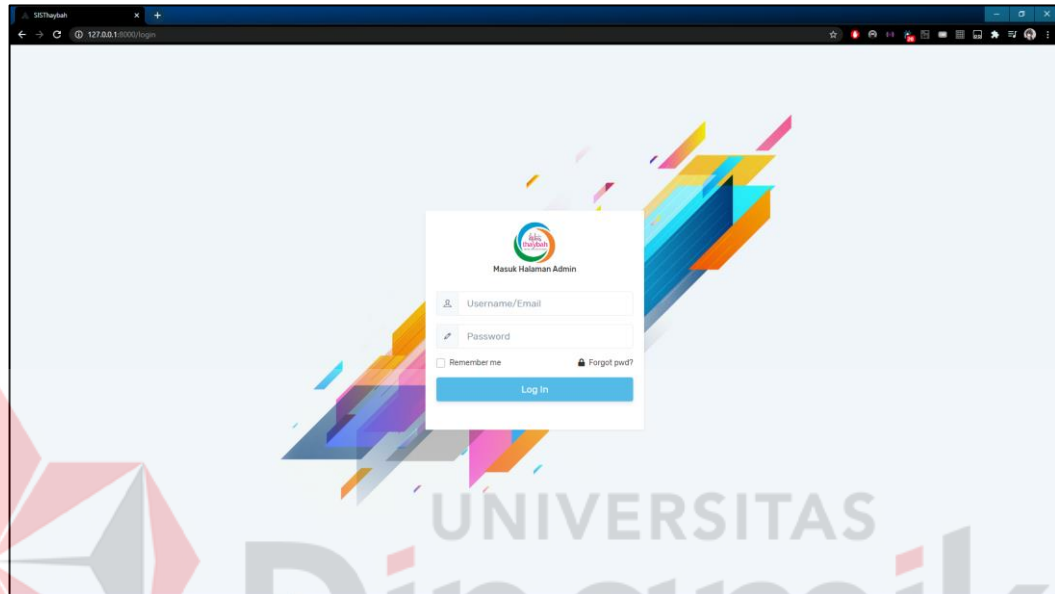


Gambar 4.21 *Class Diagram* Modul Keuangan

4.3 Desain *User Interface*

Sebelum membuat suatu aplikasi dibutuhkan suatu rancangan dasar tampilan yang akan digunakan sebagai landasan desain aplikasi, berikut adalah tampilan desain aplikasi :

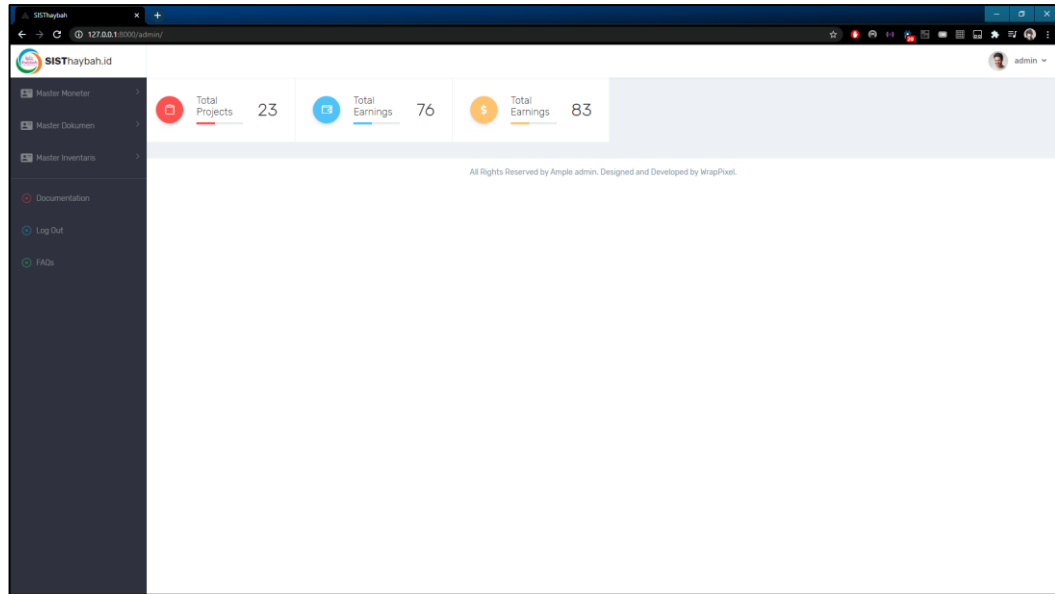
4.3.1 *User Interface Login*



Gambar 4.22 *User Interface Login*

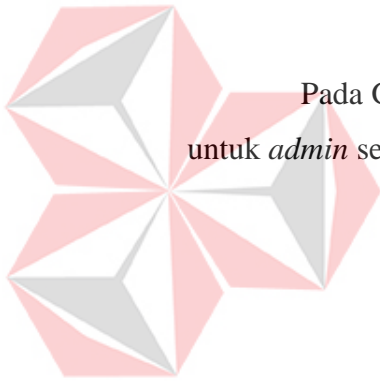
Pada Gambar diatas merupakan tampilan *menu login* untuk *admin*. Jika *admin* ingin masuk ke halaman utama, diharuskan mengisi *username* dan *password* dengan benar.

4.3.2 *User Interface* Menu Utama



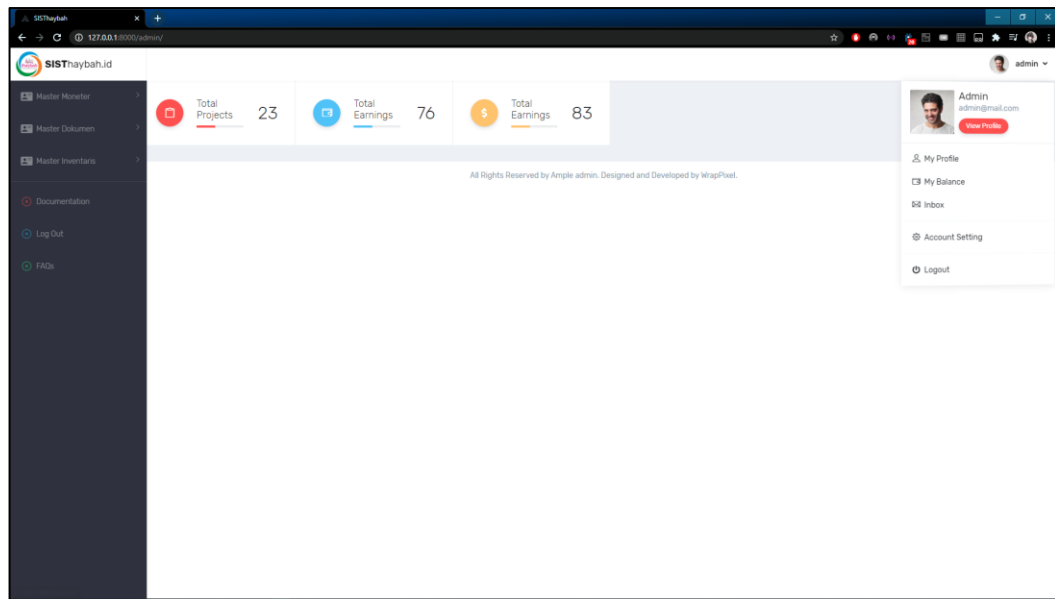
Gambar 4.23 *User Interface* Menu Utama

Pada Gambar diatas merupakan tampilan *dashboard* atau halaman utama untuk *admin* setelah melakukan *login*.



UNIVERSITAS
Dinamika

4.3.3 User Interface Logout

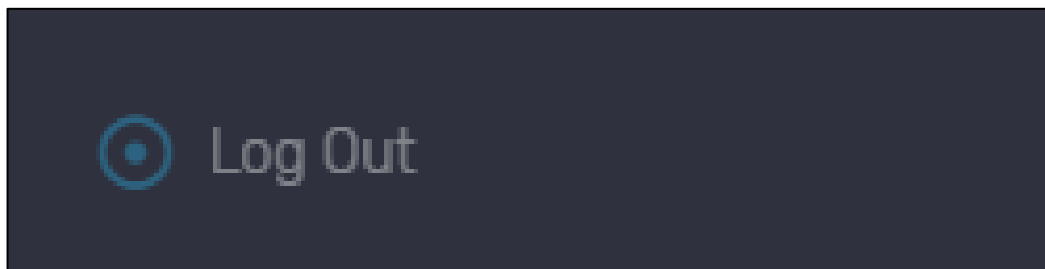


Gambar 4.24 User Interface Logout

Admin dapat melakukan *logout* dengan mengklik tombol *logout* yang berada di menu *profile* (Gambar 4.25) dan di *sidebar* (Gambar 4.26).

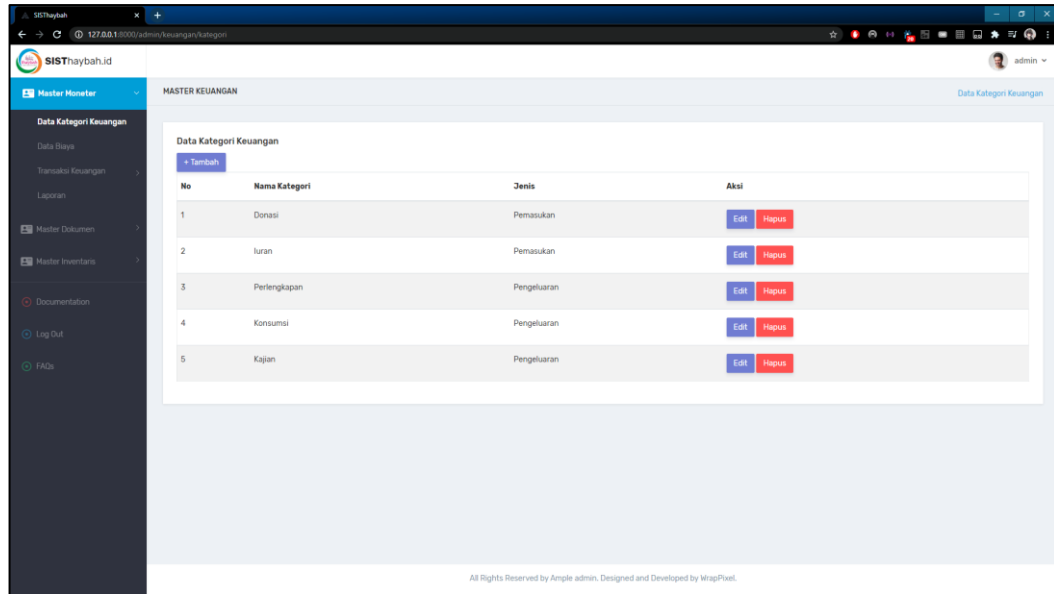


Gambar 4.25 User Interface tombol *logout* yang berada di menu *profile*



Gambar 4.26 User Interface tombol *logout* yang berada di *sidebar*

4.3.4 *User Interface* Menu Utama Kategori Keuangan



No	Nama Kategori	Jenis	Aksi
1	Donasi	Pemasukan	Edit Hapus
2	Iuran	Pemasukan	Edit Hapus
3	Penengkapan	Pengeluaran	Edit Hapus
4	Konsumsi	Pengeluaran	Edit Hapus
5	Kajian	Pengeluaran	Edit Hapus

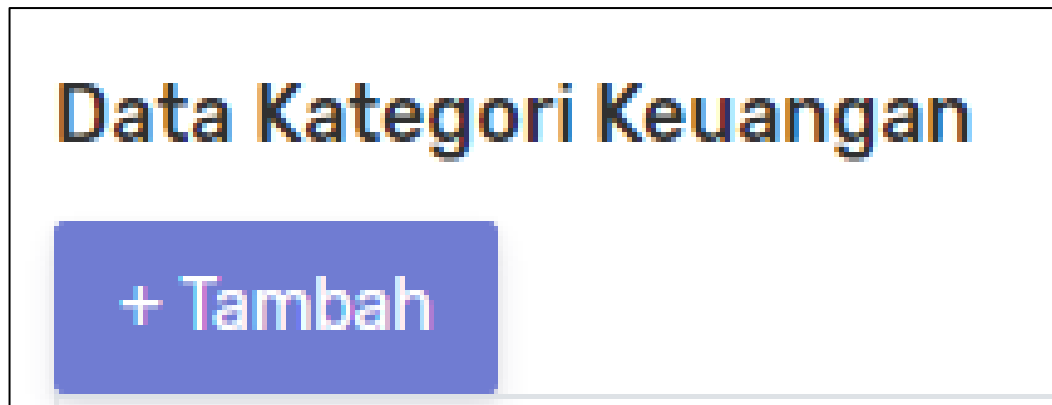
Gambar 4.27 *User Interface* Menu Utama Kategori Keuangan

Pada gambar diatas merupakan tampilan data master kategori keuangan yang terdapat 2 kolom yaitu Nama Kategori dan Jenis.



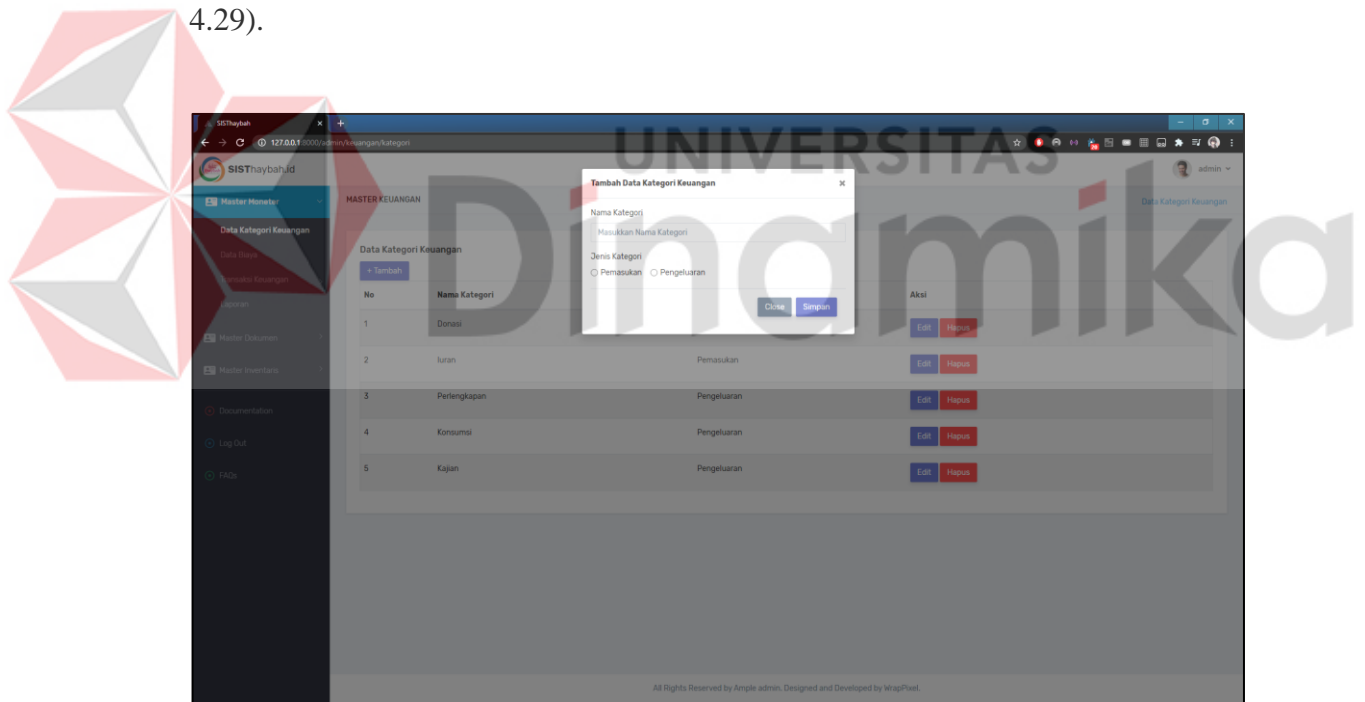
UNIVERSITAS
Dinamika

4.3.5 User Interface Tambah Kategori Keuangan



Gambar 4.28 User Interface Tombol Tambah Data Kategori Keuangan

Untuk melakukan penambahan data, *admin* melakukan klik tombol tambah (Gambar 4.28) setelah itu akan muncul *pop up form* tambah data (Gambar 4.29).



Gambar 4.29 User Interface Pop Up Form Tambah Kategori Keuangan

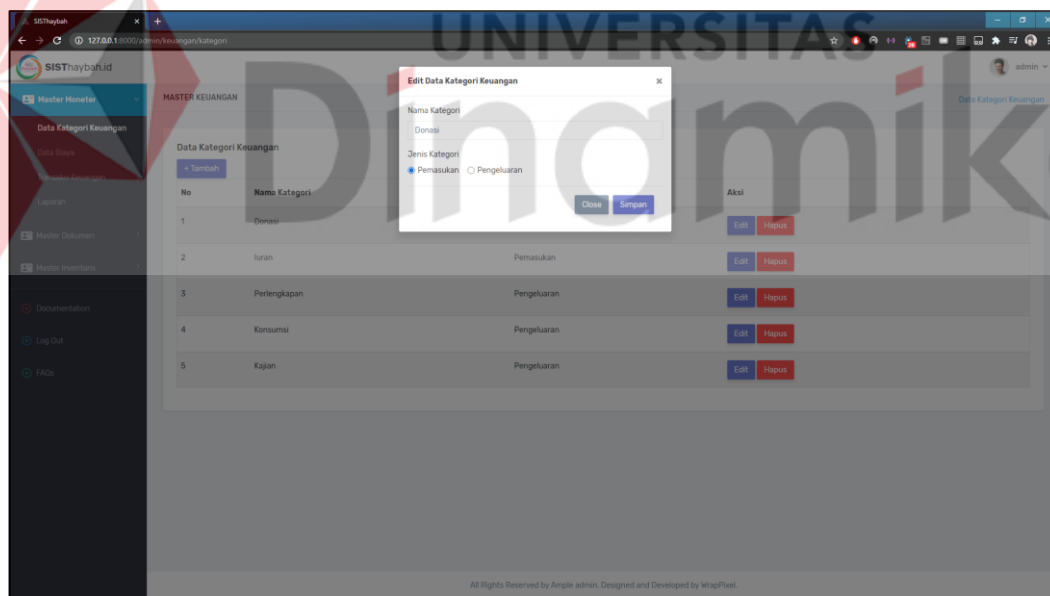
Pada form tambah data (Gambar 4.29), *admin* diminta untuk mengisi nama kategori dan memilih jenis kategori.

4.3.6 User Interface Ubah Kategori Keuangan

No	Nama Kategori	Jenis	Aksi
1	Donasi	Pemasukan	<button>Edit</button> <button>Hapus</button>
2	Iuran	Pemasukan	<button>Edit</button> <button>Hapus</button>
3	Perlengkapan	Pengeluaran	<button>Edit</button> <button>Hapus</button>
4	Konsumsi	Pengeluaran	<button>Edit</button> <button>Hapus</button>
5	Kajian	Pengeluaran	<button>Edit</button> <button>Hapus</button>

Gambar 4.30 User Interface data kategori keuangan

Jika *admin* ingin melakukan *update* data, maka *admin* dapat mengklik tombol *edit* yang terletak di kolom “Aksi” pada data yang ingin di lakukan *update* (Gambar 4.30), setelah itu akan muncul *pop up form edit* dengan data-data yang tersimpan tertulis di *textbox* (Gambar 4.31).

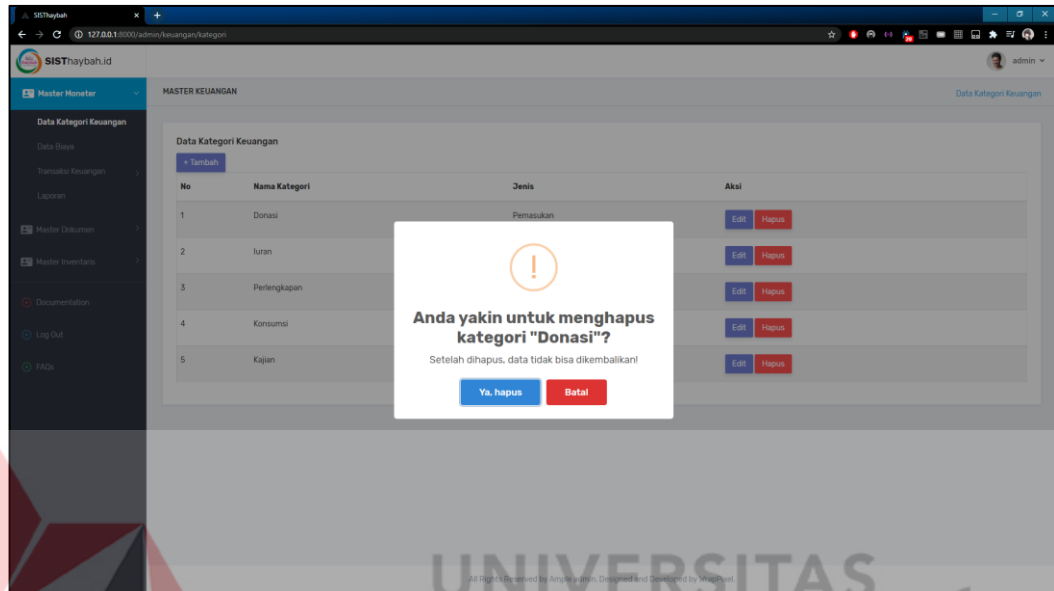


Gambar 4.31 User Interface Pop Up Form Ubah Kategori Keuangan

Pada *form edit* data (Gambar 4.31), *admin* dapat merubah data yang diinginkan.

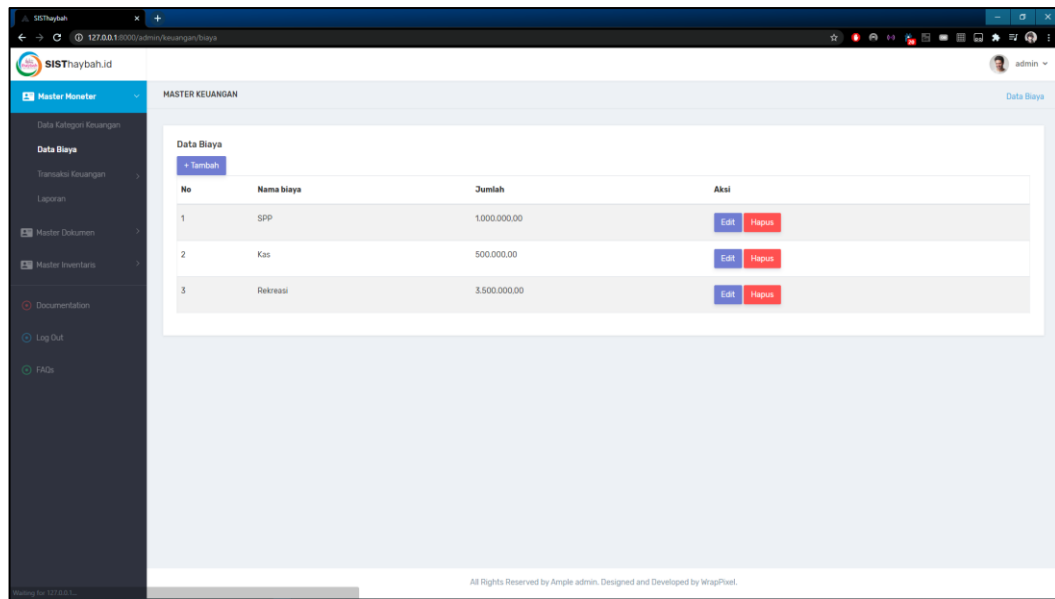
4.3.7 User Interface Hapus Kategori Keuangan

Jika *admin* ingin melakukan *delete* data, maka *admin* dapat mengklik tombol hapus yang terletak di kolom “Aksi” pada data yang ingin di hapus (Gambar 4.30), setelah itu akan muncul *pop up* konfirmasi hapus (Gambar 4.32).



Gambar 4.32 User Interface Konfirmasi Hapus Kategori Keuangan

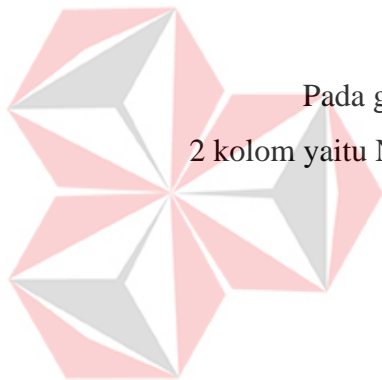
4.3.8 *User Interface* Menu Utama Biaya



No	Nama biaya	Jumlah	Aksi
1	SPP	1.000.000,00	Edit Hapus
2	Kas	500.000,00	Edit Hapus
3	Rekreasi	3.500.000,00	Edit Hapus

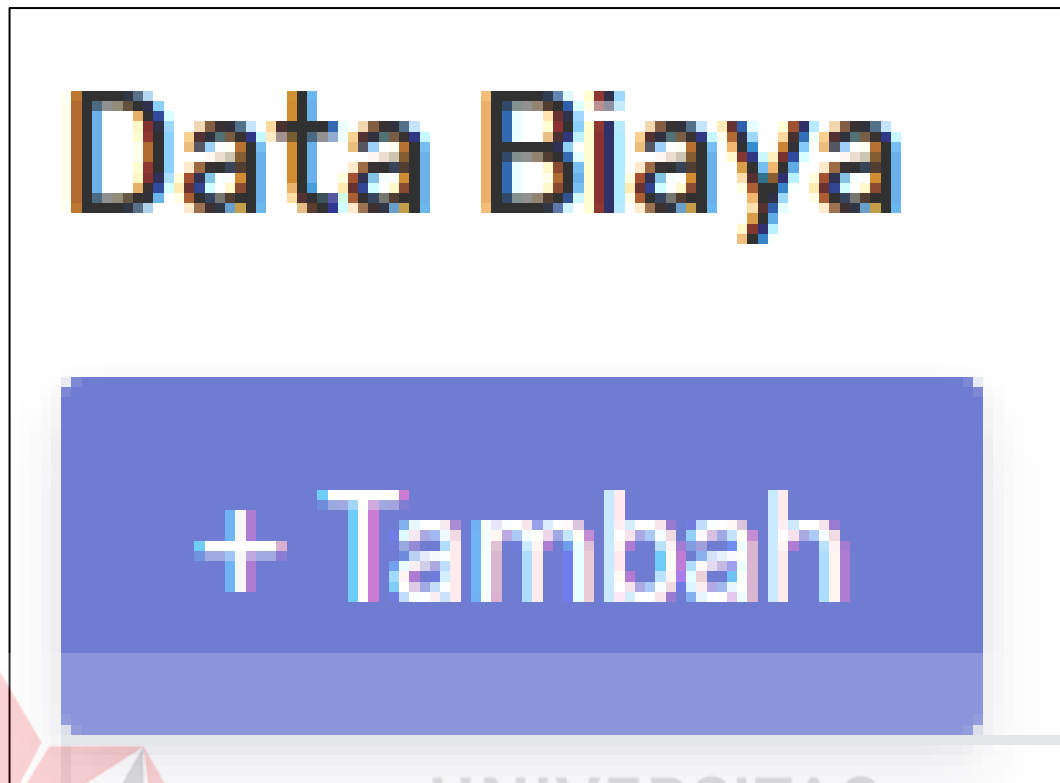
Gambar 4.33 *User Interface* Menu Utama Biaya

Pada gambar diatas merupakan tampilan data master biaya yang terdapat 2 kolom yaitu Nama Biaya dan Jumlah.



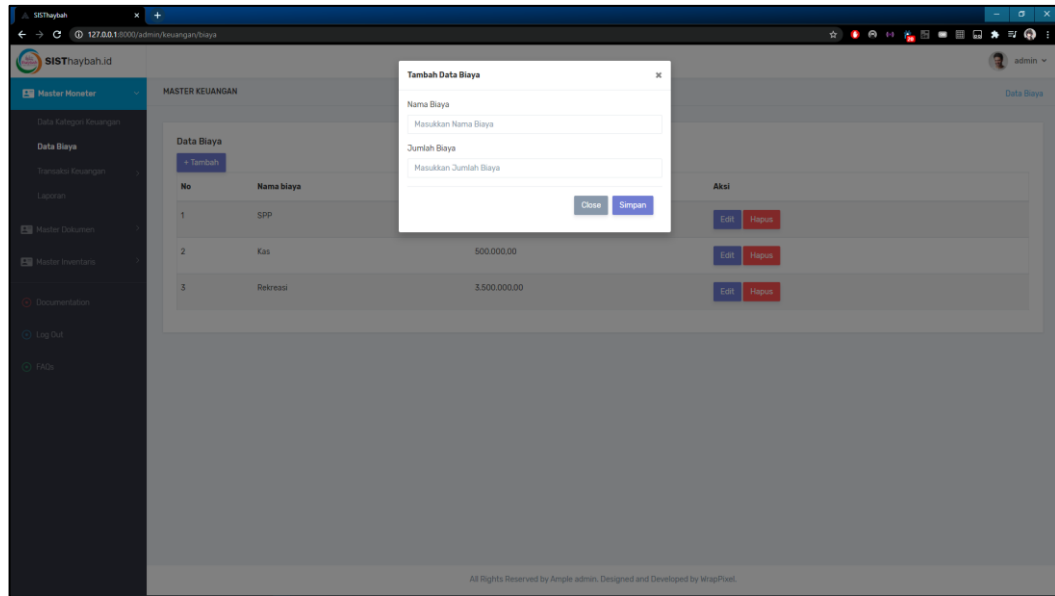
UNIVERSITAS
Dinamika

4.3.9 *User Interface* Tambah Biaya



Gambar 4.34 *User Interface* Tombol Tambah Data Biaya

Untuk melakukan penambahan data, *admin* melakukan klik tombol tambah (Gambar 4.34) setelah itu akan muncul *pop up form* tambah data (Gambar 4.35).



Gambar 4.35 *User Interface Pop Up Form Tambah Biaya*



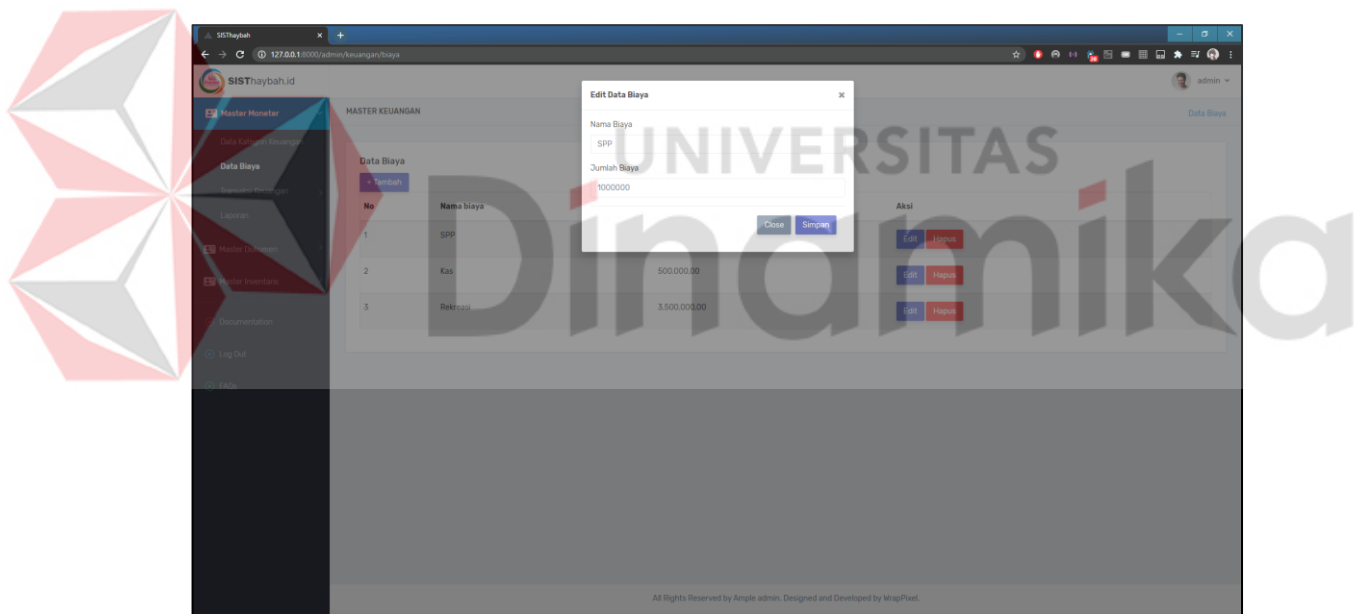
UNIVERSITAS
Dinamika

4.3.10 User Interface Ubah Biaya

No	Nama biaya	Jumlah	Aksi	
1	SPP	1000000	Edit	Hapus
2	Kas	500000	Edit	Hapus
3	Rekreasi	3500000	Edit	Hapus

Gambar 4.36 User Interface Data Biaya

Jika *admin* ingin melakukan *update* data, maka *admin* dapat mengklik tombol *edit* yang terletak di kolom “Aksi” pada data yang ingin di lakukan *update* (Gambar 4.36), setelah itu akan muncul *pop up form edit* dengan data-data yang tersimpan tertulis di *textbox* (Gambar 4.37).

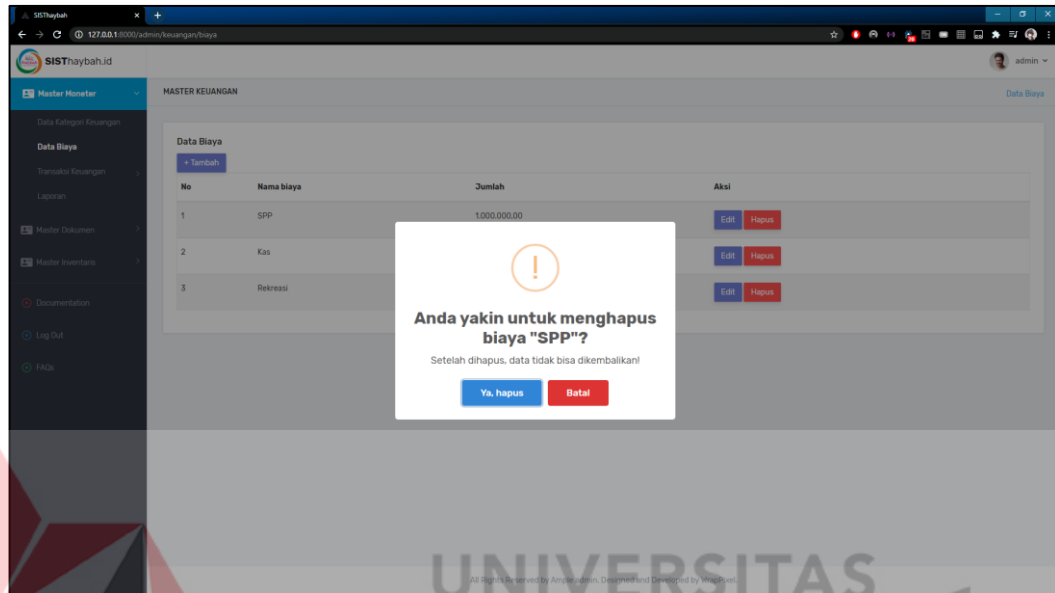


Gambar 4.37 User Interface Pop Up Form Ubah Biaya

Pada *form edit* data (Gambar 4.37), *admin* dapat merubah data yang diinginkan.







4.3.11 User Interface Hapus Biaya

Jika *admin* ingin melakukan *delete* data, maka *admin* dapat mengklik tombol hapus yang terletak di kolom “Aksi” pada data yang ingin di hapus (Gambar 4.36), setelah itu akan muncul *pop up* konfirmasi hapus (Gambar 4.38).



Gambar 4.38 User Interface Konfirmasi Hapus Biaya

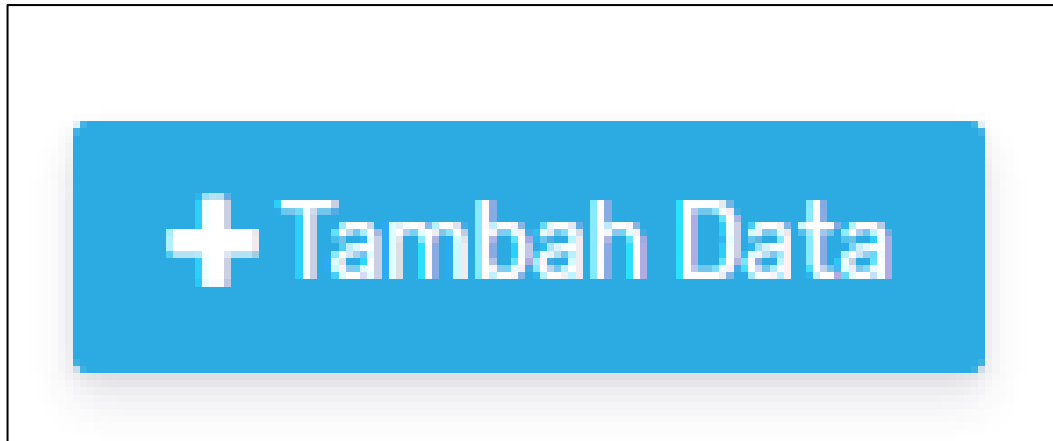
4.3.12 User Interface Menu Utama Iuran

#	Nama Biaya	Tanggal	Nama Santri	Jumlah Bayar	Telat	Jumlah Hutang	Aksi
1	SPP	2019-12	Aprilandi Nuru	Rp 750.000,00	Tidak	Rp 250.000,00	 
2	Kas	2020-1	Andhika Bati	Rp 500.000,00	Iya	Rp 0,00	 
3	Rekreasi	2020-1	Ghozzy Siemens	Rp 2.000.000,00	Tidak	Rp 1.500.000,00	 

Gambar 4.39 User Interface Menu Utama Iuran

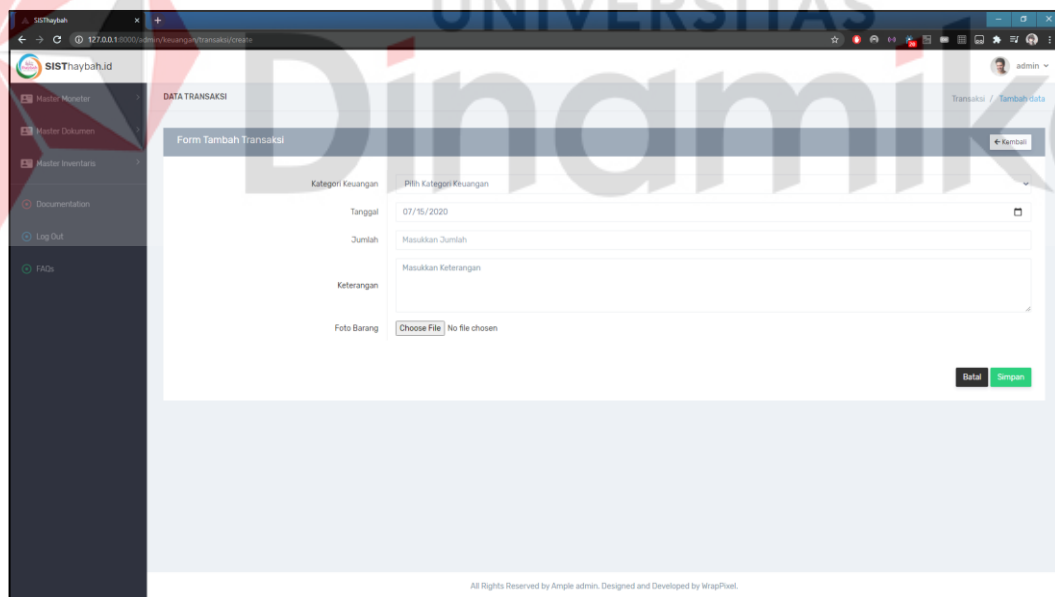
Pada gambar diatas merupakan tampilan data iuran yang terdapat 6 kolom yaitu Nama Biaya, Tanggal, Nama Santri, Jumlah Bayar, Telat, dan Jumlah Hutang.

4.3.13 User Interface Tambah Iuran



Gambar 4.40 User Interface Tombol Tambah Data Iuran

Untuk melakukan penambahan data, *admin* melakukan klik tombol tambah data (Gambar 4.40) setelah itu akan halaman akan menampilkan *form* tambah data (Gambar 4.41).

A screenshot of a web application interface. On the left is a dark sidebar with a logo and menu items: 'Master Monitor', 'Master Dokumen', 'Master Inventaris', 'Documentation', 'Log Out', and 'FAQs'. The main content area is titled 'DATA TRANSAKSI' and contains a 'Form Tambah Transaksi'. The form has fields for 'Kategori Keuangan' (with a dropdown), 'Tanggal' (with a date picker showing '07/16/2020'), 'Jumlah' (with a text input), 'Keterangan' (with a text area), and 'Foto Barang' (with a 'Choose File' button). At the bottom right of the form are 'Batal' and 'Simpan' buttons. A large, semi-transparent watermark 'UNIVERSITAS Dinamika' is overlaid across the center of the image.

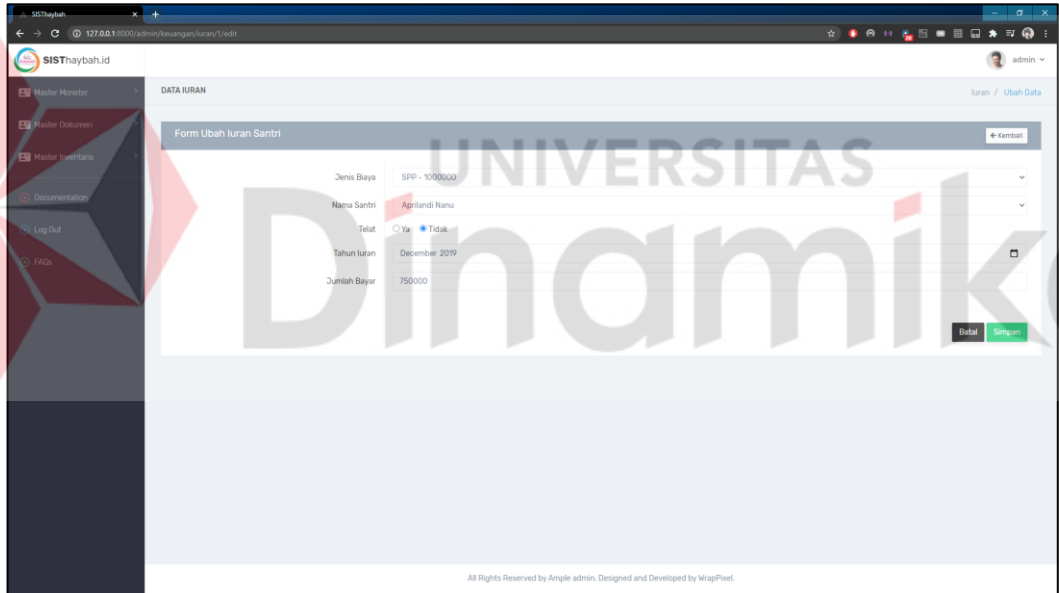
Gambar 4.41 User Interface Form Tambah Iuran

4.3.14 User Interface Ubah Iuran

#	Nama Biaya	Tanggal	Nama Santri	Jumlah Bayar	Telat	Jumlah Hutang	Aksi
1	SPP	2019-12	Aprilandi Nani	Rp 750.000,00	Tidak	Rp 250.000,00	 
2	Kas	2020-1	Andhika Bati	Rp 500.000,00	Iya	Rp 0,00	 
3	Rekreasi	2020-1	Ghozy Siemens	Rp 2.000.000,00	Tidak	Rp 1.500.000,00	 

Gambar 4.42 User Interface Data Iuran

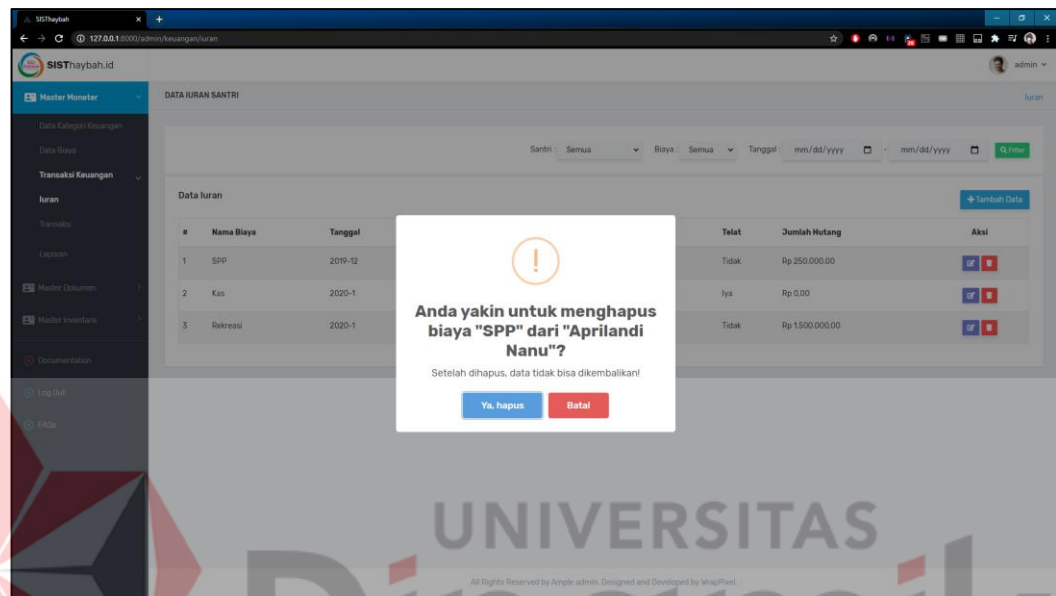
Jika *admin* ingin melakukan *update* data, maka *admin* dapat mengklik tombol yang berwarna biru tua yang berlogo pensil yang terletak di kolom “Aksi” pada data yang ingin di lakukan *update* (Gambar 4.42), setelah itu akan membuka halaman *form edit* dengan data-data yang tersimpan tertulis di *textbox* (Gambar 4.43).



Gambar 4.43 User Interface Form Ubah Iuran

4.3.15 User Interface Hapus Iuran

Jika *admin* ingin melakukan *delete* data, maka *admin* dapat mengklik tombol yang berwarna merah dengan logo *trash can* yang terletak di kolom “Aksi” pada data yang ingin di hapus (Gambar 4.42), setelah itu akan muncul *pop up* konfirmasi hapus (Gambar 4.38).



Gambar 4.44 User Interface Konfirmasi Hapus Iuran

4.3.16 User Interface Menu Utama Transaksi

Saldo : Rp 65.000,00

Kategori : Semua Jenis : Semua Tanggal : mm/dd/yyyy - mm/dd/yyyy

Data Transaksi

#	Tanggal	Kategori Keuangan	Jenis	Jumlah (Rp)	Aksi
1	2020-07-07	Donasi	Pemasukan	Rp 100.000,00	
2	2020-07-09	Konsumsi	Pengeluaran	Rp 10.000,00	
3	2020-07-14	Konsumsi	Pengeluaran	Rp 25.000,00	

+ Tambah Data

All Rights Reserved by Ample admin. Designed and Developed by WapPivot.

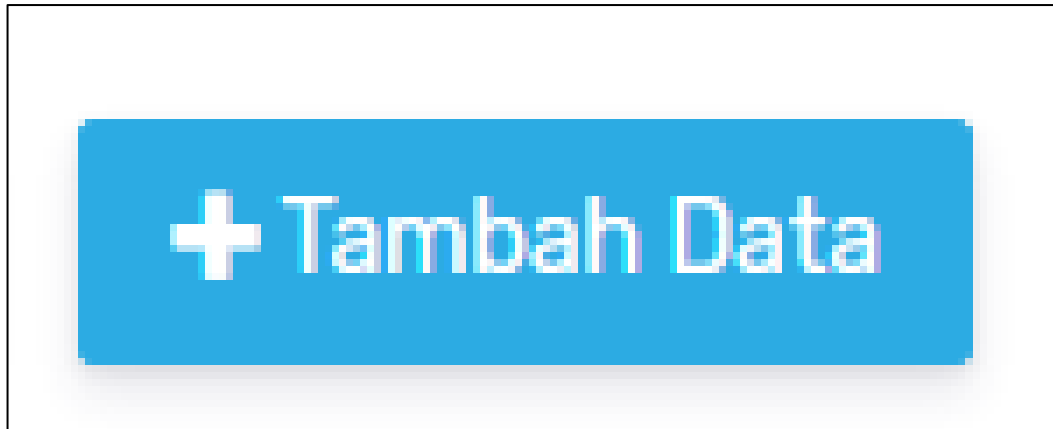
Gambar 4.45 User Interface Menu Utama Transaksi

Pada gambar diatas merupakan tampilan data transaksi yang terdapat 4 kolom yaitu Tanggal, Kategori Keuangan, Jenis, dan Jumlah.



UNIVERSITAS
Dinamika

4.3.17 User Interface Tambah Transaksi



Gambar 4.46 User Interface Tombol Tambah Data Transaksi

Untuk melakukan penambahan data, *admin* melakukan klik tombol tambah data (Gambar 4.46) setelah itu akan halaman akan menampilkan *form* tambah data (Gambar 4.47).

A screenshot of a web application interface for adding a transaction. The page has a dark sidebar on the left with a logo and menu items: "Master Monitor", "Master Dokumen", "Master Inventaris", "Documentation", "Log Out", and "FAQ". The main content area is titled "DATA TRANSAKSI" and contains a "Form Tambah Transaksi". The form fields include: "Kategori Keuangan" (a dropdown menu), "Tanggal" (a date picker set to 07/15/2020), "Jumlah" (a text input field with a placeholder "Masukkan Jumlah"), "Keterangan" (a text area with a placeholder "Masukkan Keterangan"), and "Foto Barang" (a file upload button labeled "Choose File" and a status "No file chosen"). At the bottom right of the form are two buttons: "Batal" (Cancel) and "Simpan" (Save). The browser's address bar shows the URL "127.0.0.1:8000/admin/keuangan/transaksi/create". A large, semi-transparent watermark "UNIVERSITAS Dinamika" is overlaid across the center of the image.

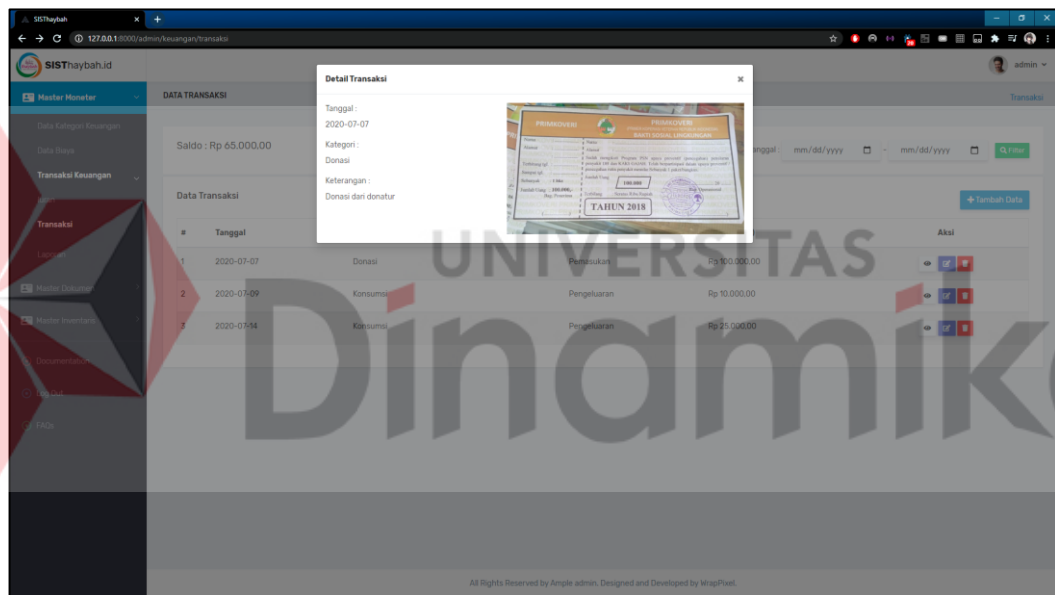
Gambar 4.47 User Interface Form Tambah Transaksi

4.3.18 User Interface Detail Transaksi

#	Tanggal	Kategori Keuangan	Jenis	Jumlah (Rp)	Aksi
1	2020-07-07	Donasi	Pemasukan	Rp 100.000.00	  
2	2020-07-09	Konsumsi	Pengeluaran	Rp 10.000.00	  
3	2020-07-14	Konsumsi	Pengeluaran	Rp 25.000.00	  

Gambar 4.48 User Interface Data Transaksi

Jika *admin* ingin melihat detail data transaksi, maka *admin* dapat mengklik tombol yang berwarna putih yang terdapat *icon* mata yang terletak di kolom “Aksi” pada data yang ingin di dilihat detail transaksinya (Gambar 4.48), setelah itu akan *pop up* detail dari transaksi tersebut (Gambar 4.49).



Gambar 4.49 User Interface Pop Up Detail Transaksi

Pada gambar diatas merupakan *pop up* dari detail transaksi yang terdapat 4 data yaitu Tanggal, Kategori, Keterangan, dan Foto Bukti.

4.3.19 User Interface Ubah Transaksi

Jika *admin* ingin melakukan *update* data, maka *admin* dapat mengklik tombol yang berwarna biru tua yang berlogo pensil yang terletak di kolom “Aksi” pada data yang ingin di lakukan *update* (Gambar 4.48), setelah itu akan membuka halaman *form edit* dengan data-data yang tersimpan tertulis di *textbox* (Gambar 4.50).

The screenshot displays a web application interface for editing a transaction. The main content area is titled 'Form Ubah Transaksi' and includes the following fields:

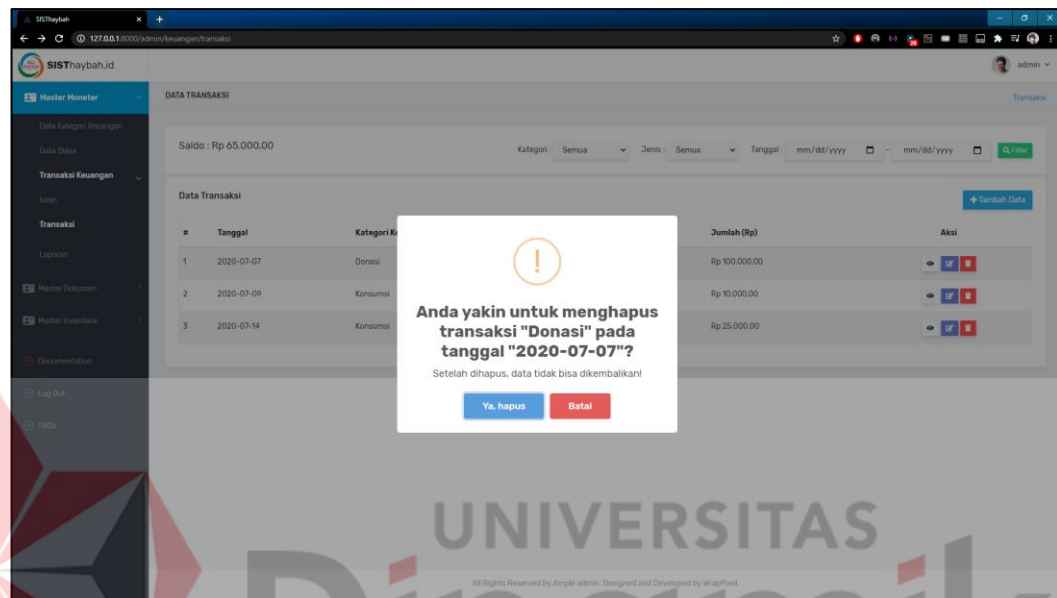
- Kategori Keuangan:** Pemasukan - Donasi
- Tanggal:** 07/07/2020
- Jumlah:** 100000
- Keterangan:** Donasi dari donatur
- Foto Barang:** Choose File (No file chosen)

At the bottom right of the form are two buttons: 'Batal' (Cancel) and 'Simpan' (Save). The interface also features a sidebar on the left with navigation links and a large watermark for 'UNIVERSITAS Dinamika'.

Gambar 4.50 User Interface Form Ubah Transaksi

4.3.20 User Interface Hapus Transaksi

Jika *admin* ingin melakukan *delete* data, maka *admin* dapat mengklik tombol yang berwarna merah dengan logo *trash can* yang terletak di kolom “Aksi” pada data yang ingin di hapus (Gambar 4.48), setelah itu akan muncul *pop up* konfirmasi hapus (Gambar 4.38).



Gambar 4.51 User Interface Konfirmasi Hapus Transaksi

4.3.21 User Interface Menu Utama Laporan

Saldo : Rp 65.000,00

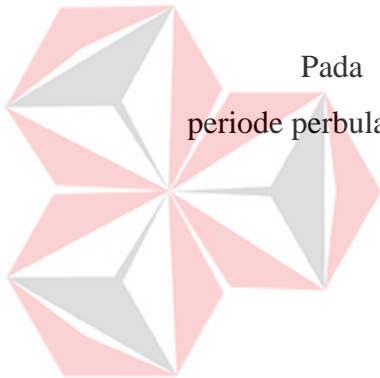
Kategori : Semua Jenis : Semua Tanggal : mm/dd/yyyy - mm/dd/yyyy Filter

#	Periode	Transaksi	Kategori Keuangan	Jenis	Total (Rp)	Aksi
1	07-2020	1	Donasi	Pemasukan	Rp 100.000,00	Detail
2	07-2020	2	Konsumsi	Pengeluaran	Rp 35.000,00	Detail

All Rights Reserved by Ample admin. Designed and Developed by MirapPixel.

Gambar 4.52 User Interface Menu Utama Laporan

Pada gambar diatas merupakan tampilan laporan yang berdasarkan periode perbulan dan per kategori keuangan.



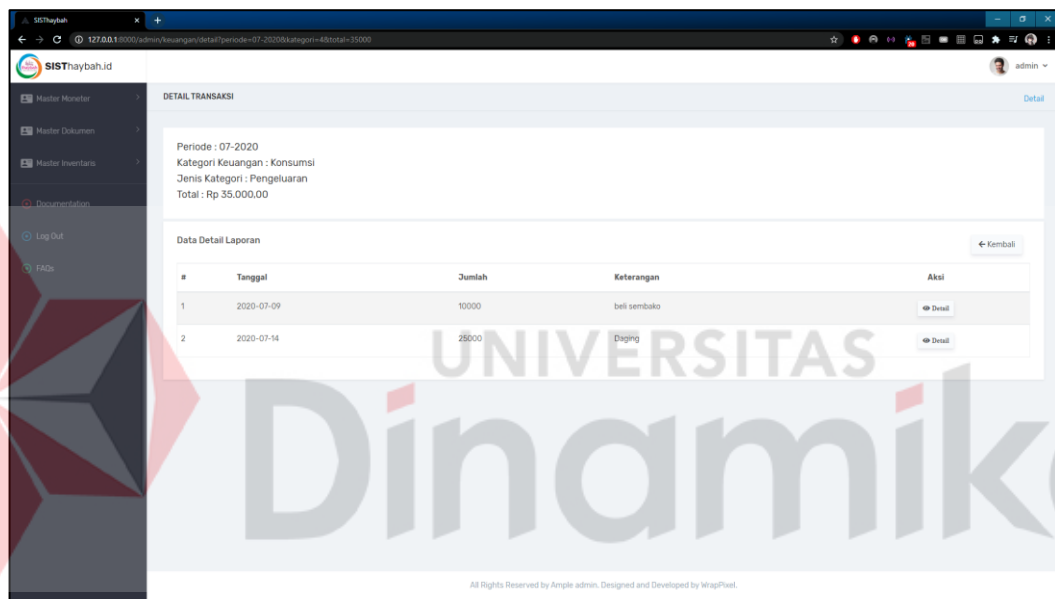
UNIVERSITAS
Dinamika

4.3.22 User Interface Menu Detail Laporan

Jika *admin* ingin melihat detail laporan, maka *admin* dapat mengklik tombol yang berwarna putih yang tertulis detail (Gambar 4.53), setelah itu akan diarahkan ke halaman detail laporan yang berisi detail dari transaksi tersebut (Gambar 4.54).

#	Periode	Transaksi	Kategori Keuangan	Jenis	Total (Rp)	Aksi
1	07-2020	1	Donasi	Pemasukan	Rp 100.000,00	Detail
2	07-2020	2	Konsumsi	Pengeluaran	Rp 35.000,00	Detail

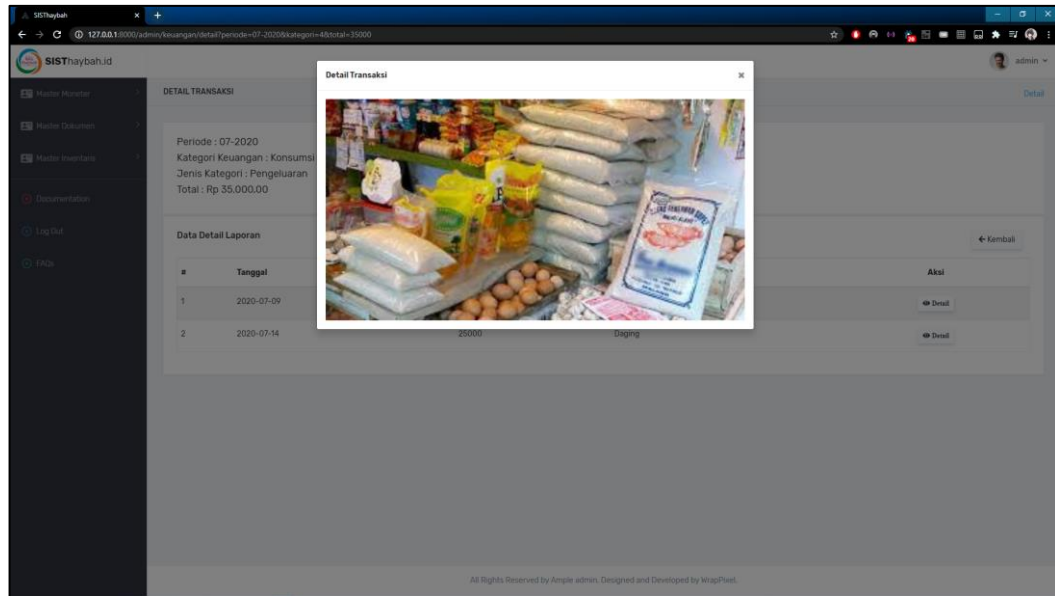
Gambar 4.53 User Interface Data Laporan yang Telah Di Rekap



#	Tanggal	Jumlah	Keterangan	Aksi
1	2020-07-09	10000	beli sembako	Detail
2	2020-07-14	25000	Daging	Detail

Gambar 4.54 User Interface Detail Laporan

Pada tombol detail yang terdapat pada gambar diatas merupakan tombol untuk melihat bukti transaksi tersebut, yang nantinya akan muncul sebuah *pop up* yang berisi gambar bukti transaksi tersebut (Gambar 4.55).



Gambar 4.55 *User Interface* Foto Bukti Transaksi



UNIVERSITAS
Dinamika

BAB V

PENUTUP

5.1 Kesimpulan

Kesimpulan yang didapat diperoleh dari rancang bangun pencatatan keuangan pada pesantren mahasiswa mahad thaybah meliputi pengelolaan master kategori keuangan, master biaya, pencatatan iuran, pencatatan transaksi serta *upload* bukti transaksi, dan pembuatan laporan. Aplikasi yang dibuat sesuai dengan kebutuhan Pesantren Mahasiswa Mahad Thaybah.

5.2 Saran

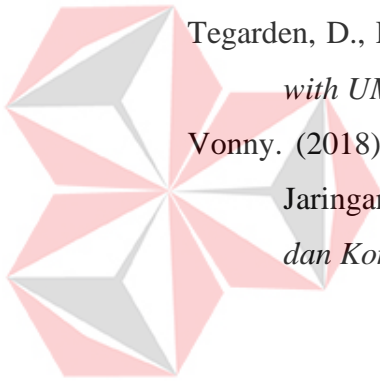
Saran untuk pengembang aplikasi yang akan melanjutkan aplikasi pencatatan keuangan berbasis *website* pada Pesantren Mahasiswa Mahad Thaybah di masa yang akan datang yaitu :

- a. Mengevaluasi terhadap aplikasi terkait kinerja aplikasi.
- b. Mengevaluasi terhadap aplikasi terkait *interface* aplikasi.
- c. Penambahan metodologi dalam perhitungan keuangan.
- d. Penambahan atau perluasan laporan keuangan.

DAFTAR PUSTAKA

- Agustini, & Kurniawan, W. J. (2019). Sistem E-Learning Do'a dan Iqro' dalam Peningkatan Proses Pembelajaran pada TK Amal Ikhlas. *Jurnal Mahasiswa Aplikasi Teknologi Komputer dan Informasi*, I(3), 154-159.
- Binarso, Y. A., Sarwoko, E. A., & Bahtiar, N. (2012). Pembangunan Sistem Informasi Alumni Berbasis Web Pada Program Studi Teknik Informatika Universitas Diponegoro. *Journal of Informatics and Technology*, I(1), 72-84.
- Budiman, D. A., & Nugraha, D. M. (2019). Aplikasi Raport Online Berbasis Web Menggunakan Framework Codeigniter. *Jurnal Computech & Bisnis*, XIII(2), 112-121.
- Firmansyah, Y., & Pitriani. (2017). Penerapan Metode SDLC Waterfall Dalam Pembuatan Aplikasi Pelayanan Anggota Pada CU Duta Usaha Bersama Pontianak. *Jurnal Bianglala Informatika*, V(2), 66-74.
- Firmansyah, Y., Maulana, R., & Fatin, N. (2020). Sistem Informasi Pengaduan Warga Berbasis Website. *Jurnal Cendika*, XIX(1), 397-404.
- Ghiffary, M. N., Susanto, T. D., & Herdiyanti, A. (2018). Analisis Komponen Desain Layout, Warna, dan Kontrol Pada Antarmuka Pengguna Aplikasi Mobile Berdasarkan Kemudahan Penggunaan (Studi Kasus: Aplikasi Olride). *Jurnal Teknik ITS*, VII(1), 143-148.
- Hermanto, B., Yusman, M., & Nagara. (2019). Sistem Informasi Manajemen Keuangan Pada PT Hulu Balang Mandiri Menggunakan Framework Laravel. *Jurnal Komputasi*, VII(1), 17-26.
- Isa, I. G., & Hartawan, G. P. (2017). Perancangan Aplikasi Koperasi Simpan Pinjam Berbasis Web. *Jurnal Ilmiah Ilmu Teknologi*, V(10), 139-151.
- Kieso, D. E., Weygandt, J. J., & Warfield, T. D. (2008). *Akuntansi Intermediate*. Jakarta: Erlangga.
- Martani, D., Siregar, S. V., Wardhani, R., Farahmita, A., & Tanujaya, E. (2012). *Akuntansi Keuangan Menengah*. Jakarta: Salemba Empat.
- Minokaura, M. F., Afisina, A., Rahman, A., R., E. A., Rizal, M. A., & Rakhmawati,

- N. A. (2020). Studi Empiris Penggunaan Framework pada Website Jaringan Dokumentasi dan Informasi Hukum Daerah Pulau Jawa. *Jurnal Ilmu Komunikasi UHO*, V(2), 93-101.
- Mukhlis, M. (2018). *Model Pengajaran dan Kepemimpinan di Pondok Pesantren Jamsaren Surakarta Tahun 1995-2015*. Surakarta: Universitas Muhammadiyah Surakarta.
- Prayitno, A., & Safitri, Y. (2015). Pemanfaatan Sistem Informasi Perpustakaan Digital Berbasis Website Untuk Para Penulis. *Indonesian Journal on Software Engineering*, I(1), 28-37.
- Putra, D. W., & Andriani, R. (2019). Unified Modelling Language (UML) dalam Perancangan Sistem. *Jurnal TEKNOIF*, VII(1), 32-39.
- Sari, M. (2019). Aplikasi Data Pasien dan Penentuan Gizi Ibu Hamil Pada Puskesmas Sungai Tabuk. *Technologia*, X(3), 172-178.
- Tegarden, D., Dennis, A., & Wixom, B. H. (2013). *Systems Analysis and Design with UML*. Singapore: John Wiley & Sons, Inc.
- Vonny. (2018). Sistem Informasi Pendistribusian Paterial Promosi Sparepart ke Jaringan Dealer di PT Daya Adicipta Motora. *Jurnal Teknologi Informasi dan Komunikasi*, VIII(2), 81-89.



UNIVERSITAS
Dinamika