

**RANCANG BANGUN SISTEM INFORMASI MANAJEMEN ASET PADA
PSDKU UNIVERSITAS LAMPUNG WAY KANAN MENGGUNAKAN
FRAMEWORK LARAVEL**

(Skripsi)

Oleh

NAOMI BELINDA MANURUNG



**PROGRAM STUDI S1 TEKNIK INFORMATIKA
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2024**

ABSTRAK

RANCANG BANGUN SISTEM INFORMASI MANAJEMEN ASET PADA PSDKU UNIVERSITAS LAMPUNG WAY KANAN MENGGUNAKAN *FRAMEWORK* LARAVEL

Oleh

NAOMI BELINDA MANURUNG

PSDKU Universitas Lampung Way Kanan memiliki banyak aset yang dapat mendukung kegiatan akademik mahasiswa. Namun, PSDKU Universitas Lampung Way Kanan belum memiliki sistem informasi manajemen aset yang terstruktur. Pengelolaan data, pemeliharaan dan perbaikan aset akan menjadi sulit tanpa sistem yang mendukung. Oleh karena itu, penelitian ini bertujuan untuk mengatasi permasalahan tersebut melalui implementasi Sistem Informasi Manajemen Aset dengan menggunakan *Framework* Laravel. Untuk mengembangkan sistem tersebut, digunakan metode Kanban untuk meningkatkan efisiensi dan kinerja. Pengembangan dalam metode Kanban terdiri dari *backlog*, *design*, *development*, *testing*, dan *done*. Hasil penelitian ini mencapai kesuksesan dengan menyediakan sistem informasi manajemen aset yang mampu mengelola data aset, menyediakan layanan peminjaman aset, melaporkan kerusakan aset, dan menyajikan riwayat aset dikarenakan penggunaan *framework* laravel dan metode kanban. Sistem tersebut telah diuji menggunakan metode *blackbox* dan menghasilkan *output* sesuai harapan, sehingga dapat disimpulkan bahwa aplikasi sesuai antara kebutuhan pengguna dengan fungsional sistem.

Kata kunci: PSDKU, Sistem Informasi, Aset, Laravel, Kanban

ABSTRACT

DESIGN OF AN ASSET MANAGEMENT INFORMATION SYSTEM AT PSDKU UNIVERSITY LAMPUNG WAY KANAN USING THE LARAVEL FRAMEWORK

BY

NAOMI BELINDA MANURUNG

PSDKU Lampung University Way Kanan has many assets that can support student academic activities. However, PSDKU Lampung University Way Kanan does not yet have a structured asset management information system. Data management, maintenance, and repair of assets will be difficult without a supporting system. Therefore, this research aims to overcome these problems by implementing an Asset Management Information System using the Laravel Framework. To develop the system, the Kanban method is used to improve efficiency and performance. Development in the Kanban method consists of backlog, design, development, testing, and done. The results of this study achieved success by providing an asset management information system that can manage asset data, provide asset lending services, report asset damage, and present asset history due to the use of the Laravel framework and the Kanban method. The system has been tested using the black box method and produces output as expected, so it can be concluded that the application is in accordance with user needs and system functionality.

Keywords: PSDKU, Information System, Assets, Laravel, Kanban

**RANCANG BANGUN SISTEM INFORMASI MANAJEMEN ASET PADA
PSDKU UNIVERSITAS LAMPUNG WAY KANAN MENGGUNAKAN
FRAMEWORK LARAVEL**

**Oleh
NAOMI BELINDA MANURUNG**

Skripsi

Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA TEKNIK

**Pada
Program Studi S1 Teknik Informatika
Fakultas Teknik**



**PROGRAM STUDI S1 TEKNIK INFORMATIKA
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2024

Judul Skripsi : **RANCANG BANGUN SISTEM
INFORMASI MANAJEMEN ASET PADA
PSDKU UNIVERSITAS LAMPUNG WAY
KANAN MENGGUNAKAN *FRAMEWORK*
LARAVEL.**

Nama Mahasiswa : **Naomi Belinda Manurung**

Nomor Pokok Mahasiswa : **2055061006**

Program Studi : **SI Teknik Informatika**

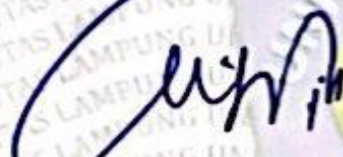
Fakultas : **Teknik**

MENYETUJUI

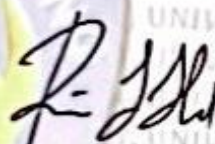
1. Komisi Pembimbing

Pembimbing Utama

Pembimbing Pendamping



I. Bigih Forda Nama, S.T., M.T.I., IPM.
NIP. 198307122008121003

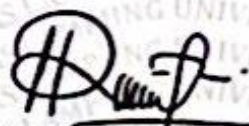


Rio Ariestia Pradipta, S.Kom., M.T.I.
NIP. 198603232019031013

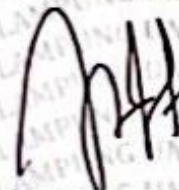
2. Mengetahui

Ketua Jurusan Teknik Elektro

Ketua Program Studi Informatika



Herlinawati, S.T., M.T.
NIP. 197103141999032001



Yessi Mulyani, S.T., M.T.
NIP. 197312262000122001

MENGESAHKAN

1. Tim Penguji

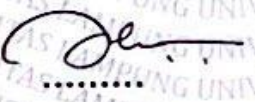
Ketua : Ir. Gigih Forda Nama, S.T., M.T.I, IPM.



Sekretaris : Rio Ariestia Pradipta, S.Kom., M.T.I.



Penguji : Ir. M. Komarudin, S.T., M.T



2. Dekan Fakultas Teknik



Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc. }

NIP 197509282001121002

Tanggal Lulus Ujian Skripsi: 03 Januari 2024

SURAT PERNYATAAN

Saya yang bertandatangan dibawah ini, menyatakan bahwa skripsi saya yang berjudul “RANCANG BANGUN SISTEM INFORMASI MANAJEMEN ASET PADA PSDKU UNIVERSITAS LAMPUNG WAY KANAN MENGGUNAKAN *FRAMEWORK* LARAVEL” dengan ini menyatakan bahwa skripsi saya dibuat oleh saya sendiri. Semua hasil yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi ini merupakan salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan hukum atau akademik yang berlaku.

Bandar Lampung, 29 Januari 2024

Pembuat Pernyataan,



Naomi Belinda Manurung

NPM. 2055061006

RIWAYAT HIDUP



Penulis bernama Naomi Belinda Manurung yang merupakan anak pertama dari pasangan Batoanto Manurung dan Angela Panjaitan. Penulis lahir di Bekasi pada tanggal 18 September 2002. Penulis telah menyelesaikan sekolah dasar di SD Negeri 173642 Hasahatan pada tahun 2014, SMP Negeri 2 Porsea pada tahun 2017 dan SMA Negeri 1 Siantar Narumonda pada tahun 2020. Penulis melanjutkan pendidikan di Fakultas Teknik Jurusan Teknik Elektro Program Studi Teknik Informatika Universitas Lampung pada tahun 2020. Selama menempuh pendidikan di jurusan Teknik Elektro, program studi Teknik Informatika Universitas Lampung, penulis juga aktif mengikuti beberapa kegiatan seperti:

1. Anggota organisasi Himpunan Mahasiswa Teknik Elektro (2021-2022).
2. Anggota organisasi Unit Kegiatan Mahasiswa Sains dan Teknologi (2021-2022)
3. Anggota organisasi Badan Eksekutif Mahasiswa Fakultas Teknik (2021-2022).
4. Peserta program KMMI (Kredensial Mikro Mahasiswa Indonesia) dengan kursus Konsep, Strategi, dan Implementasi Internet of Things (IoT) pada tahun 2021.
5. *Volunteer* sebagai *content designer* di Puan Bisa (Agustus 2022-Februari2023).
6. *Volunteer* sebagai desain grafis di NguliahKuy (2023-2024).
7. Magang sebagai desain grafis di Sekretariat Direktorat Jenderal Pendidikan Tinggi, Riset, dan Teknologi Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi (Agustus 2022-Desember 2022).
8. Studi independen *Full Stack Developer* di Metrodata Academy (Februari 2023-Juni 2023).
9. Magang sebagai desain grafis di SF Group Services Enterprise (Maret 2023-Juni 2023).
10. Magang sebagai Web Developer di Hendrawan Media Artha (Juli 2023-Oktober 2023).

PERSEMBAHAN

Segala puji syukur saya panjatkan kepada Allah Bapa di surga atas berkat karunia yang telah diberikan. Dengan segala kerendahan hati, saya persembahkan skripsi ini kepada:

Mama tercinta, yang telah melahirkanku, merawatku, membesarkanku, yang selalu menyayangi dan mendidikku dengan sepenuh hati.

Bapak tersayang, yang telah membesarkanku dengan seluruh cinta kasih, memberikan pengetahuannya, dan selalu mendukung serta mendoakan untuk keberhasilanku.

Serta, almamater yang sangat saya banggakan
UNIVERSITAS LAMPUNG

SANWACANA

Segala puji syukur hanya bagi Tuhan Yang Maha Esa, yang telah melimpahkan berkat dan rahmat-Nya sehingga penulis dapat menyelesaikan penulisan skripsi yang berjudul: RANCANG BANGUN SISTEM INFORMASI MANAJEMEN ASET PADA PSDKU UNIVERSITAS LAMPUNG WAY KANAN MENGGUNAKAN *FRAMEWORK* LARAVEL, sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik pada Fakultas Teknik Universitas Lampung. Dalam penyelesaian skripsi ini, penulis mendapatkan banyak bimbingan dan dukungan sehingga penyusunan skripsi ini berjalan dengan baik. Maka pada kesempatan ini penulis mengucapkan terima kasih kepada:

1. Bapak Dr. Eng. Helmy Fitriawan, S.T., M.Sc selaku Dekan Fakultas Teknik Universitas Lampung.
2. Ibu Herlinawati, S.T., M.T. selaku Ketua Jurusan Teknik Elektro Universitas Lampung.
3. Ibu Yessi Mulyani, S.T., M.T. selaku Ketua Program Studi Teknik Elektro Universitas Lampung.
4. Bapak Ir. Gigih Forda Nama, S.T., M.T.I, IPM. selaku Pembimbing Utama, yang telah bersedia meluangkan waktu untuk memberikan pengarahan dan bimbingan dalam pembuatan skripsi ini hingga selesai.
5. Bapak Rio Ariestia P, S. Kom. M.T.I. selaku Pembimbing Kedua, yang telah bersedia meluangkan waktu untuk memberikan pengarahan dan bimbingan dalam pembuatan skripsi ini hingga selesai.
6. Bapak Mona Arif Muda, S.T., M.T. selaku Pembimbing Akademik yang telah bersedia meluangkan waktu untuk memberikan pengarahan dan bimbingan selama menempuh pendidikan Program Studi Teknik Informatika Universitas Lampung.

7. Bapak Dr. Sudrajat, S.E., M. Acc., Akt sebagai Ketua Program Studi Akuntansi yang telah bersedia meluangkan waktu untuk memberikan masukan dalam pengembangan skripsi ini.
8. Mbak Rika beserta seluruh jajarannya atas bantuannya dalam menyelesaikan urusan administrasi di Teknik Informatika Universitas Lampung.
9. Kak Suta, Kak Tomi, Kak El, selaku pihak PSDKU yang banyak membantu selama penelitian di PSDKU Universitas Lampung Way Kanan.
10. Bapak, mama, serta adik tercinta yaitu Polin, dan Malvin yang selalu memberikan arahan, semangat, dukungan, doa untuk menyelesaikan skripsi ini.
11. Kak Anwar Sahid dan kak Verry Gusti Andrea yang banyak membantu dalam penyelesaian skripsi ini.
12. Azzah Shafiyah dan Ferdinand Pakpahan selaku sahabat yang selalu menemani, memberikan bantuan, dukungan, dan doa dalam penyelesaian skripsi ini.
13. Seluruh dosen program studi Teknik Informatika Universitas Lampung yang banyak memberikan ilmu kepada penulis.
14. Semua pihak yang turut serta dalam penyelesaian penelitian yang tidak bisa penulis sebutkan satu persatu.

Semoga skripsi ini dapat memberikan manfaat bagi setiap orang yang membacanya.

Bandar Lampung, Januari 2024

Penulis,

Naomi Belinda Manurung

NPM 2055061006

DAFTAR ISI

DAFTAR ISI.....	xii
DAFTAR GAMBAR	xiv
DAFTAR TABEL.....	xvii
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.....	3
1.6 Sistematika Penulisan.....	3
II. TINJAUAN PUSTAKA	4
2.1 PSDKU Unila Way Kanan	4
2.2 Aset.....	4
2.3 Manajemen Aset.....	5
2.4 PHP.....	5
2.5 <i>Framework</i>	5
2.6 Laravel.....	6
2.7 Composer.....	6
2.8 Basis Data.....	6
2.9 MySQL	7
2.10 <i>Software Development Life Cycle (SDLC)</i>	7
2.11 Agile	8
2.12 Kanban.....	8
2.13 <i>Unified Modelling Language (UML)</i>	9
2.14 <i>Use Case Diagram</i>	9
2.15 <i>Activity Diagram</i>	10
2.16 Vmware	10
2.17 VMware vSphere.....	10
2.18 <i>Blackbox Testing</i>	11

2.19 <i>Equivalence Partitioning</i>	11
2.20 Penelitian Terkait	11
III. METODE PENELITIAN.....	16
3.1 Waktu dan Tempat Penelitian	16
3.2 Perangkat Penelitian	16
3.2.1 Perangkat Keras (<i>Hardware</i>).....	16
3.2.2 Perangkat Lunak (<i>Software</i>).....	17
3.3 Tahapan Penelitian	17
3.3.1 Studi Literatur	18
3.3.2 Pengembangan Sistem	18
3.3.3 Instalasi ke Server.....	52
IV. HASIL DAN PEMBAHASAN.....	53
4.1 Observasi.....	53
4.2 Pengembangan.....	54
4.2.1 Pengembangan Sistem Menggunakan Kanban.....	54
4.2.2 Model.....	67
4.2.3 Controller.....	68
4.2.4 View.....	74
4.3 Instalasi ke Server.....	80
4.4 Pengujian.....	89
4.5 Hasil.....	90
4.5.1 Fitur Admin.....	91
4.5.2 Fitur Staf Aset.....	102
4.6 Alur Kerja Pengguna pada Sistem Informasi Manajemen Aset.....	108
V. KESIMPULAN DAN SARAN	110
5.1 Kesimpulan.....	110
5.2 Saran.....	110
DAFTAR PUSTAKA	111

DAFTAR GAMBAR

Gambar II.1. Fase SDLC.....	7
Gambar II.2. <i>Kanban Board</i>	14
Gambar III.1 Tahapan Penelitian	17
Gambar III.2 <i>Use Case Diagram</i>	20
Gambar III.3 <i>Activity Diagram Login</i>	21
Gambar III.4 <i>Activity Diagram Menambah Aset</i>	22
Gambar III.5 <i>Activity Diagram Update Aset</i>	23
Gambar III.6 <i>Activity Diagram Hapus Aset</i>	24
Gambar III.7 <i>Activity Diagram Tambah Kategori Aset</i>	25
Gambar III.8 <i>Activity Diagram Update Kategori Aset</i>	26
Gambar III.9 <i>Activity Diagram Hapus Kategori Aset</i>	27
Gambar III.10 <i>Activity Diagram Tambah Data Perbaikan Aset</i>	28
Gambar III.11 <i>Activity Diagram Mengajukan Peminjaman Aset</i>	29
Gambar III.12 <i>Activity Diagram Pengajuan Laporan Kerusakan Aset</i>	30
Gambar III.13 <i>Activity Diagram Konfirmasi Pengajuan Peminjaman Aset</i>	31
Gambar III.14 <i>Entity Relationship Diagram</i>	32
Gambar III.15 <i>Mockup Login</i>	33
Gambar III.16 <i>Mockup Beranda</i>	33
Gambar III.17 <i>Mockup Data Aset</i>	34
Gambar III.18 <i>Mockup Tambah Aset</i>	35
Gambar III.19 <i>Mockup Kategori Aset</i>	36
Gambar III.20 <i>Mockup Tambah Kategori Aset</i>	37
Gambar III.21 <i>Mockup Data Lokasi Aset</i>	37
Gambar III.22 <i>Mockup Tambah Data Lokasi Aset</i>	38
Gambar III.23 <i>Mockup Laporan Kerusakan Aset</i>	39
Gambar III.24 <i>Mockup Peminjaman Aset</i>	39
Gambar III.25 <i>Mockup Pengajuan Peminjaman Aset</i>	40
Gambar III.26 <i>Mockup Pengajuan Laporan Kerusakan Aset</i>	41

Gambar III.27 <i>Mockup</i> Perbaikan Aset.....	42
Gambar III.28 <i>Mockup</i> Tambah Data Perbaikan Aset	43
Gambar III.29 <i>Mockup</i> Data Riwayat Aset.....	44
Gambar IV.1 Aset PSDKU	53
Gambar IV.2 Pengembangan Sistem Menggunakan Kanban	54
Gambar IV.3 Card Login ke Sistem.....	55
Gambar IV.4 Card Admin CRUD Aset	56
Gambar IV.5 Card Admin CRUD Kategori Aset	57
Gambar IV.6 Card Admin CRUD Lokasi Aset	58
Gambar IV.7 Card Admin Mengelola Riwayat Aset.....	59
Gambar IV.8 Card Staf Aset CRUD Aset.....	60
Gambar IV.9 Card Staf Aset Melaporkan Kerusakan Aset	61
Gambar IV.10 Card Admin Melihat Laporan Kerusakan Aset.....	62
Gambar IV.11 Card Staf Aset Melakukan Peminjaman Aset.....	63
Gambar IV.12 Card Admin Mengelola Peminjaman Aset	64
Gambar IV.13 Card Admin Mengelola Data Perbaikan Aset.....	65
Gambar IV.14 Card Admin Menghubungkan Aset dengan Kategori dan Lokasi Tertentu	66
Gambar IV.15 Card Profile	67
Gambar IV.16 Error Code.....	78
Gambar IV.17 Error Web.....	79
Gambar IV.18 Perbaikan Code	79
Gambar IV.19 Mempersiapkan server Vmware vSphere	80
Gambar IV.20 Melakukan Instalasi Ubuntu 23.04	81
Gambar IV.21 Instalasi Apache	82
Gambar IV.22 Transfer Proyek Laravel ke Server	83
Gambar IV.23 Instalasi Composer.....	83
Gambar IV.24 Instalasi PHP	84
Gambar IV.25 Konfigurasi Environment Laravel	85
Gambar IV.26 Generate Key Aplikasi Laravel.....	86
Gambar IV.27 Konfigurasi Web Server.....	86
Gambar IV.28 Migrasi Database.....	87

Gambar IV.29 Menjalankan Web	88
Gambar IV.30 Halaman Login Admin	91
Gambar IV.31 Halaman Dashboard Admin.....	92
Gambar IV.32 Halaman Data Aset Admin	93
Gambar IV.33 Halaman Tambah Data Aset Admin	93
Gambar IV.34 Halaman Detail Data Aset Admin	94
Gambar IV.35 Halaman Edit Data Aset Admin	95
Gambar IV.36 Halaman Data Kategori Aset Admin	95
Gambar IV.37 Halaman Tambah Kategori Aset Admin.....	96
Gambar IV.38 Halaman Data Lokasi Aset Admin	96
Gambar IV.39 Halaman Tambah Lokasi Aset Admin.....	97
Gambar IV.40 Halaman Data Laporan Kerusakan Aset Admin.....	98
Gambar IV.41 Halaman Data Peminjaman Aset Admin	98
Gambar IV.42 Halaman Data Perbaikan Aset Admin	99
Gambar IV.43 Halaman Tambah Data Perbaikan Aset Admin	100
Gambar IV.44 Halaman Detail Perbaikan Aset Admin.....	100
Gambar IV.45 Halaman Data Riwayat Aset Admin.....	101
Gambar IV.46 Halaman Tambah Data Riwayat Aset Admin.....	101
Gambar IV.47 Halaman Profile Admin	102
Gambar IV.48 Halaman Login Staf Aset.....	102
Gambar IV.49 Halaman Dashboard Staf Aset	103
Gambar IV.50 Halaman Data Aset Staf Aset	104
Gambar IV.51 Halaman Tambah Data Aset Staf Aset	104
Gambar IV.52 Halaman Data Laporan Kerusakan Staf Aset	105
Gambar IV.53 Halaman Tambah Data Laporan Kerusakan Staf Aset	105
Gambar IV.54 Halaman Edit Laporan Kerusakan Staf Aset	106
Gambar IV.55 Halaman Data Peminjaman Aset Staf Aset	106
Gambar IV.56 Halaman Tambah Data Peminjaman Aset Staf Aset	107
Gambar IV.57 Halaman Edit Data Peminjaman Aset Staf Aset.....	107
Gambar IV.58 Halaman Profile Staf Aset	108

DAFTAR TABEL

Tabel 1 Jadwal Pelaksanaan Penelitian.....	16
Tabel 2 <i>User Stories</i>	18
Tabel 3 Skenario Pengujian <i>Login</i>	45
Tabel 4 Skenario Pengujian Menambah Aset	45
Tabel 5 Skenario Pengujian Mengedit Aset.....	46
Tabel 6 Skenario Pengujian Menghapus Aset	46
Tabel 7 Skenario Pengujian Menambah Kategori Aset	47
Tabel 8 Skenario Pengujian Mengedit Kategori Aset.....	47
Tabel 9 Skenario Pengujian Menghapus Kategori Aset	48
Tabel 10 Skenario Pengujian Menambah Lokasi Aset	48
Tabel 11 Skenario Pengujian Mengedit Lokasi Aset.....	49
Tabel 12 Skenario Pengujian Menghapus Lokasi Aset.....	49
Tabel 13 Skenario Pengujian Mengajukan Peminjaman Aset	50
Tabel 14 Skenario Pengujian Melaporkan Kerusakan Aset.....	50
Tabel 15 Skenario Pengujian Menyetujui Pengajuan Peminjaman Aset	51
Tabel 16 Skenario Pengujian Menolak Pengajuan Peminjaman Aset	51
Tabel 17 Skenario Pengujian Menambah Data Perbaikan Aset.....	52
Tabel 18 Model	68
Tabel 19 Controller	69
Tabel 20 View	75
Tabel 21 Pengujian.....	89

I. PENDAHULUAN

1.1 Latar Belakang

Universitas Lampung, sebagai salah satu perguruan tinggi terkemuka di Indonesia, memiliki banyak aset. Aset-aset ini meliputi berbagai macam barang, properti, perangkat teknologi, dan sumber daya lainnya yang digunakan untuk mendukung proses pembelajaran, penelitian, dan administrasi. Pengelolaan aset yang efektif dan efisien sangat penting untuk memastikan kelangsungan operasional universitas dan menjaga nilai investasi yang signifikan dalam aset tersebut.

PSDKU Universitas Lampung Way Kanan adalah salah satu program studi di luar Universitas Lampung yang terletak di Way Kanan, Provinsi Lampung. Sebagai wujud pendidikan tinggi, PSDKU memiliki berbagai jenis aset, termasuk gedung, perpustakaan, dan sumber daya lainnya yang penting untuk mendukung kegiatan akademik dan operasionalnya. PSDKU bertanggung jawab atas aset-aset yang digunakan dalam program-program akademik dan operasional di lokasinya. Salah satu tantangan dalam manajemen aset adalah pengelolaan yang akurat, terpadu, dan mudah diakses. Tanpa sistem yang tepat, risiko kehilangan aset, pemborosan sumber daya, dan ketidakakuratan data dapat meningkat. Oleh karena itu, penggunaan sistem informasi dalam mengelola aset menjadi semakin penting. Seiring dengan perkembangan teknologi informasi, penting bagi PSDKU Way Kanan untuk memiliki sistem informasi manajemen aset yang modern, efisien, dan terintegrasi guna memudahkan pengelolaan aset-aset mereka.

Framework Laravel merupakan salah satu kerangka kerja pengembangan perangkat lunak yang populer, terutama dalam pengembangan aplikasi web. *Framework* ini memiliki berbagai fitur yang memungkinkan pengembang untuk membangun aplikasi yang kuat, aman, dan dapat dikelola dengan baik. Penggunaan Laravel dalam pengembangan sistem informasi manajemen aset di PSDKU Universitas Lampung Way Kanan dapat membantu dalam mempercepat proses pengembangan dan menyediakan antarmuka yang mudah digunakan.

Berdasarkan masalah yang telah diuraikan, maka pada penelitian ini dibuatlah

sistem informasi yang digunakan untuk mengelola aset dan sebagai bahan laporan tugas akhir (skripsi) dengan judul “Rancang Bangun Sistem Informasi Manajemen Aset pada PSDKU Universitas Lampung Way Kanan Menggunakan *Framework* Laravel”. Dengan adanya sistem ini, diharapkan pengelolaan aset dapat lebih efisien, transparan, dan terkelola dengan baik. Sistem ini dapat memungkinkan untuk memantau aset, mengelola ruang, pelaporan dan analisis yang dapat mempermudah perkuliahan, terutama ketika ada pertumbuhan jumlah jurusan. Selain itu, penggunaan teknologi informasi seperti ini juga dapat membantu universitas dalam mengadopsi praktik manajemen terbaik dan meningkatkan kualitas layanan kepada pengguna aset.

1.2 Rumusan Masalah

Berdasarkan latar belakang permasalahan yang telah dijelaskan diatas, dapat dirumuskan masalah sebagai berikut:

1. Bagaimana mengembangkan sistem informasi manajemen aset menggunakan framework Laravel?
2. Apa manfaat yang dapat diperoleh oleh PSDKU Universitas Lampung Way Kanan dari implementasi sistem informasi manajemen aset ini?

1.3 Batasan Masalah

Batasan masalah dilakukan agar penelitian lebih terarah dan memudahkan dalam pembahasan, sehingga tujuan dari penelitian dapat tercapai. Batasan masalah dalam pengembangan sistem ini yaitu:

- a. Menggunakan *web framework* Laravel 10.
- b. Pengembangan sistem dilakukan dengan menggunakan metode Kanban.
- c. Instalasi web dilakukan menggunakan VMware vSphere.

1.4 Tujuan

Tujuan dari penelitian ini adalah mengembangkan sistem informasi manajemen aset menggunakan *framework* Laravel dan menerapkan metode Kanban dalam pengembangannya.

1.5 Manfaat

Adapun manfaat dari penelitian ini diantaranya:

1. Membantu PSDKU Universitas Lampung Way Kanan dalam melacak, memantau, dan mengelola semua aset dengan lebih efisien.
2. Data mengenai aset akan tersedia secara transparan.

1.6 Sistematika Penulisan

Sistematika penulisan yang digunakan pada laporan penelitian ini adalah sebagai berikut:

BAB I : PENDAHULUAN

Pada bab ini secara umum meliputi latar belakang penelitian, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan penelitian serta sistematika penulisan.

BAB II : TINJAUAN PUSTAKA

Pada bab ini berisi tentang prinsip, dasar teori dan teori penunjang yang digunakan sebagai acuan dalam melakukan penelitian.

BAB III : METODOLOGI PENELITIAN

Pada bab ini membahas mengenai metodologi penelitian yang digunakan dalam Rancang Bangun Sistem Informasi Manajemen Aset PSDKU Universitas Lampung Way Kanan Menggunakan *Framework* Laravel.

BAB IV : HASIL DAN PEMBAHASAN

Pada bab ini membahas mengenai hasil serta pembahasan yang diperoleh dalam penelitian Rancang Bangun Sistem Informasi Manajemen Aset PSDKU Universitas Lampung Way Kanan Menggunakan *Framework* Laravel.

BAB V : KESIMPULAN DAN SARAN

Pada bab ini berisi tentang kesimpulan dan saran hasil penelitian.

DAFTAR PUSTAKA

LAMPIRAN

II. TINJAUAN PUSTAKA

2.1 PSDKU Unila Way Kanan

Program Studi di Luar Kampus Utama (PSDKU) adalah program studi yang diselenggarakan oleh sebuah perguruan tinggi yang menyediakan pendidikan akademik di luar wilayah kabupaten/kota/kota administratif tempat kampus utama perguruan tinggi tersebut berada. Perguruan tinggi harus mendapatkan izin dari Mendikbud sebelum menjalankan program studi ini. Kampus utama adalah tempat domisili perguruan tinggi yang menyelenggarakan pendidikan akademik di wilayah kabupaten/kota/kota administratif, sesuai dengan keputusan Menteri yang mengatur pendirian perguruan tinggi tersebut.

PSDKU Unila Way Kanan merupakan bagian dari inisiatif Universitas Lampung (Unila) untuk memperkuat pembangunan nasional dan daerah, terutama dalam meningkatkan akses pendidikan tinggi dan pengembangan sumber daya manusia di Kabupaten Waykanan. Program PSDKU Unila Way Kanan adalah bukti nyata komitmen Unila dalam mengurangi kesenjangan akses pendidikan tinggi di wilayah Lampung. Pemilihan Kabupaten Way Kanan sebagai lokasi program PSDKU didasarkan pada letak geografisnya yang strategis, memungkinkan akses dari beberapa kabupaten di Provinsi Lampung. Saat ini, program studi yang tersedia adalah Diploma III Akuntansi, sementara program Sarjana Ilmu Komputer akan segera dibuka. Dalam tahun ajaran baru 2023/2024 ini, terdapat 40 mahasiswa baru prodi D3 Akuntansi yang terdaftar menjadi mahasiswa pada program PSDKU Kab. Way Kanan melalui ujian mandiri. Para mahasiswa baru ini akan mengikuti pendidikan yang telah disesuaikan dengan kebutuhan Kabupaten Way Kanan dan belajar sesuai dengan kurikulum yang sudah disesuaikan dengan diajar oleh dosen yang berkualitas.

2.2 Aset

Aset adalah sesuatu yang memiliki nilai ekonomi yang dapat dimiliki oleh individu, perusahaan, maupun dimiliki pemerintah yang dapat dinilai secara finansial [1].

Aset sering kali digunakan untuk mengukur kekayaan atau nilai ekonomi dari suatu entitas. Aset dikelompokkan menjadi dua kategori utama yaitu:

a. Aset Tetap

Aset tetap digunakan untuk operasi bisnis dalam jangka panjang. Contoh aset tetap adalah tanah, bangunan, kendaraan, dan lain-lain.

b. Aset Lancar

Aset lancar merupakan aset yang diharapkan diubah menjadi uang tunai dan digunakan dalam operasi bisnis jangka waktu satu tahun atau kurang. Contoh aset lancar adalah kas, inventaris, surat berharga, dan lain-lain.

2.3 Manajemen Aset

Manajemen aset adalah pendekatan terstruktur untuk mengelola seluruh jenis aset (baik itu aset fisik, finansial, teknologi informulirasi, atau aset lainnya) dalam suatu organisasi atau entitas dengan tujuan untuk memaksimalkan nilai aset tersebut selama siklus hidupnya. Manajemen aset adalah tindakan mengelola berbagai jenis aset dengan efisien agar dapat memberikan nilai terbesar kepada organisasi atau individu yang memiliki atau menggunakannya. Ini melibatkan tahap perencanaan, perolehan, pemeliharaan, penggunaan, evaluasi, dan penghapusan aset sesuai dengan kebijakan dan rencana strategis organisasi [1].

2.4 PHP

PHP atau *Hypertext Preprocessor* adalah bahasa pemrograman *server-side* yang dirancang khusus untuk *website* [2]. Di dalam halaman HTML, kode PHP dapat disisipkan dan akan dieksekusi setiap kali halaman tersebut dikunjungi. PHP memiliki beberapa keunggulan diantaranya: (1). Performa handal. (2). Mendukung hampir semua basisdata, mulai dari MySQL, Oracle, PostgreSQL, Sybase, dan lainnya. (3). Dapat dijalankan pada hampir semua sistem operasi seperti Linux, Unix, Windows, Mac Os, dan lain-lain. (4). Perintah PHP mudah dipahami karena sebagian besar diambil dari bahasa pemrograman C/C++, Java, dan Perl [3].

2.5 Framework

Framework merupakan elemen pemrograman yang dapat digunakan ulang kapan

saja, sehingga *programmer* tidak harus membuat skrip yang serupa untuk tugas yang sama [4]. *Framework* memungkinkan pengembang untuk fokus pada logika bisnis khusus aplikasi mereka daripada harus membangun ulang seluruh infrastruktur perangkat lunak.

2.6 Laravel

Laravel dikembangkan oleh Taylor Otwell pada April 2011. Laravel adalah sebuah *framework web* berbasis PHP yang bersifat *open-source* dan dibangun dengan konsep MVC (*Model View Controller*). MVC merupakan sebuah pendekatan perangkat lunak yang memisahkan aplikasi logika dari presentasi.

a. *Model*

Umumnya, model berisi fungsi-fungsi yang mendukung pengelolaan *database*, seperti memasukkan data ke *database*, pembaruan data, dan lain-lain.

b. *View*

View merupakan bagian yang bertanggung jawab tampilan halaman *web* ke pengguna.

c. *Controller*

Controller merupakan bagian yang menghubungkan *model* dan *view* [5].

Laravel merupakan kerangka kerja PHP yang dirancang dengan baik, yang menyediakan berbagai perangkat dan fitur untuk memudahkan pengembangan. Ini termasuk Eloquent ORM yang kuat, rute, validitas, dan lainnya yang dapat menghemat waktu dalam pengembangan.

2.7 Composer

Composer adalah suatu *dependency manager* khusus PHP. Suatu *library* dapat diinstall melalui Composer dan akan secara otomatis menginstall maupun memperbaharui *library* yang dibutuhkan [6].

2.8 Basis Data

Basis data merupakan sekumpulan data yang terstruktur dalam bentuk tabel yang saling terhubung maupun berdiri sendiri [7]. Basis data adalah salah satu komponen penting dalam sistem informulirasi karena menjadi landasan dalam menyediakan

informulirasi.

2.9 MySQL

MYSQL adalah DBMS (*Database Management System*) yang *open source* yang mendukung *database* SQL sebagai bahasa interaktif dalam mengelola data. MySQL mampu beroperasi secara stabil pada berbagai sistem operasi termasuk Windows, Linux, dan Unix. MySQL adalah *database* yang menggunakan enkripsi *password*, sehingga lebih terlindungi dengan keberadaan *password* untuk mengaksesnya [8]. MySQL memiliki beberapa kelebihan yaitu:

- a. Gratis, stabil, dan tangguh
- b. Flesibel dengan berbagai pemrograman
- c. Keamanan yang baik
- d. Kemudahan *management database*
- e. Perkembangan software cukup kuat

Sementara kekurangan MySQL sendiri yaitu kurang mendukung koneksi ke bahasa pemrograman visual dan data yang ditangani belum terlalu besar.

2.10 Software Development Life Cycle (SDLC)

SDLC adalah pendekatan bertahap untuk analisis dan desain yang menyatakan bahwa sistem dapat dikembangkan secara optimal melalui penerapan siklus tertentu dalam kegiatan analisis dan pengguna [9].



Gambar II.1. Fase SDLC

Ada berbagai jenis SDLC yaitu Waterfall, Agile, Iterative, spiral, RAD (Rapid

Application Development), Big Bang, Incremental, Lean, dan Crystal Model.

2.11 Agile

Agile adalah suatu kumpulan metode dan metodologi yang membantu berpikir dengan lebih efektif, bekerja secara lebih efisien, dan mengambil keputusan yang lebih baik. Metode ini mencakup seluruh aspek yang terdapat dalam rekayasa perangkat lunak tradisional, termasuk manajemen proyek, desain serta arsitektur perangkat lunak, dan perbaikan proses. Setiap metode dan metodologi tersebut terdiri dari praktik-praktik yang disederhanakan dan dioptimalkan untuk mempermudah penerapannya. Selain itu, Agile juga mewakili suatu cara berpikir, karena pendekatan berpikir yang tepat dapat memberikan dampak besar terhadap sejauh mana dapat mengadopsi praktik-praktik tersebut dengan efektif. Cara berpikir ini membantu untuk berbagi informulirasi secara lebih baik dan memungkinkan untuk bersama-sama mengambil keputusan proyek yang krusial [10]. Metode Agile memiliki beberapa keunggulan, diantaranya fleksibilitas, peningkatan kualitas, transparansi, dan keterlibatan tim yang tinggi. Ada beberapa jenis metode Agile yang populer, yaitu: Scrum, Kanban, Extreme Programming (XP), Agile Model, dan lain-lain.

2.12 Kanban

Kanban adalah sistem penjadwalan kerja menggunakan kartu atau papan nama yang bertujuan meningkatkan produktivitas tim dengan mengurangi waktu yang tidak produktif. Kanban digunakan untuk memvisualisasikan pekerjaan, membentuk alur kerja terstruktur, membatasi pekerjaan yang sedang berlangsung dan meningkatkan efektivitas dan efisiensi jadwal atau waktu pengembangan. Dalam pengembangan perangkat lunak, kanban *board* memiliki WIP (*Work In Progress*) Limit yang membatasi berapa banyak pekerjaan yang dapat dilakukan pada rentang waktu tertentu [11]. Proses perancangan sistem dengan metode kanban mencakup langkah-langkah berikut:

a. Visualisasi alur kerja

Kanban memungkinkan memecah pekerjaan menjadi segmen yang dapat dipantau. Setiap tugas atau langkah penelitian dapat diwakili sebagai kartu pada papan

Kanban, yang dapat membantu melihat dengan jelas apa yang telah dilakukan, sedang dikerjakan, dan masih harus dilakukan.

b. Mengatur WIP (*work in progress*)

Kanban menetapkan batasan untuk tugas yang sedang dikerjakan.

c. Mengukur lead time

Kanban menghitung waktu penyelesaian untuk setiap segmen pekerjaan.

d. Menyusun *product backlog*

Kanban menentukan kebutuhan sistem sesuai permintaan pelanggan.

e. Mengimplementasikan sistem berbasis kanban

Kanban menggunakan papan kanban untuk mengelola alur kerja proyek pengembangan perangkat lunak.

Kanban adalah metode yang sangat fleksibel. Kanban memungkinkan dengan mudah menyesuaikan penelitian dengan perubahan dalam prioritas atau tujuan penelitian. Dengan menggunakan kanban, dapat memastikan pekerjaan yang paling penting atau mendesak dapat selalu mendapat perhatian yang tepat, mengendalikan alur pekerjaan dengan lebih baik, dan meningkatkan transparansi dalam sebuah proyek. Kanban dapat menambahkan, menghapus, atau menggeser tugas dengan mudah.

2.13 *Unified Modelling Language (UML)*

Unified Modelling Language (UML) merupakan bahasa pemodelan yang mengadopsi konsep orientasi objek. UML merupakan alat untuk menggambarkan dan mendokumentasikan hasil analisis dan desain yang berisi sintaks dalam memodelkan sistem. Tujuan utama dalam desain UML adalah memberikan kepada pengguna (analisis dan desain sistem) suatu bahasa pemodelan visual yang ekspresif, sehingga memungkinkan untuk mengembangkan dan berbagi model data yang bermakna [12].

2.14 *Use Case Diagram*

Use case diagram adalah representasi dari fungsi sistem yang mengilustrasikan sejumlah aktor eksternal dan relasinya dengan *use case* yang diberikan oleh sistem. Sebuah *use case* direpresentasikan sebagai elips horizontal dalam suatu diagram

UML *use case*. Terdapat 2 istilah dalam *use case* yaitu *system use case* (interaksi dengan sistem) dan *business use case* (interaksi bisnis dengan konsumen atau kejadian nyata) [12].

2.15 Activity Diagram

Activity diagram adalah gambaran dari alur aktivitas, objek, *state*, transisi *state* dan *event* [12]. *Activity diagram* digunakan untuk menggambarkan urutan kegiatan sehingga dapat diterapkan pada aktivitas lainnya seperti *use case* atau interaksi.

2.16 VMware

VMware adalah perusahaan yang terkenal dalam dunia komputasi virtual yang menyediakan berbagai solusi perangkat lunak untuk virtualisasi, manajemen cloud, dan infrastruktur data-center. Beberapa produk dari VMware yaitu VMware vSphere, VMware ESXi (hypervisor), VMware Workstation (untuk penggunaan desktop), VMware Fusion (untuk penggunaan Mac). Beberapa manfaat utama dari menggunakan VMware adalah konsolidasi server, meningkatkan keamanan karena VM dapat diatur untuk tidak memiliki akses ke VM lain, manajemen fleksibel, mudah dicadangkan dan dipulihkan, dan dapat membantu mengoptimalkan penggunaan daya dan mendorong efisiensi energi dengan mengkonsolidasikan beban kerja ke dalam jumlah host yang lebih sedikit.

2.17 VMware vSphere

VMware vSphere (*bare-metal hypervisor*) merupakan produk utama dari VMware yang digunakan untuk membangun virtualisasi server. Virtualisasi yang dimaksud disini yaitu bentuk sebuah server atau yang terkait dengannya dapat dipecah menjadi beberapa piranti virtual (*virtual appliance*) yang berdiri sendiri dan mempunyai fungsionalitas masing-masing [13]. VMware vSphere menyediakan sejumlah fitur dan komponen yang memungkinkan virtualisasi, manajemen sumber daya, dan penyediaan aplikasi yang efisien. VMware vSphere diinstall langsung pada server yang terutama digunakan untuk memvirtualisasi pusat data dalam bisnis yang dapat diintegrasikan dengan VMware vCenter Server sehingga pusat data ini menjadi bagian dari cloud. Penggunaan virtualisasi server lebih optimal karena

dapat mengurangi biaya yang besar dalam membeli hardware untuk membangun server, bisa menjalankan lebih dari satu sistem operasi secara bersamaan dengan fungsi yang berbeda, dan lebih mudah dalam proses *maintance* dan *recovery* karena hanya berfokus pada satu server.

2.18 Blackbox Testing

Blackbox Testing adalah suatu metode yang dipakai untuk menguji sebuah perangkat lunak tanpa harus memperhatikan rincian internal perangkat lunak. Proses *Blackbox Testing* melibatkan mencoba program yang telah dibuat dengan memasukkan data pada setiap formulirnya. Pengujian ini dilakukan untuk menilai apakah program beroperasi sesuai dengan yang dibutuhkan [14]. Secara umum, metode blackbox yang sering digunakan adalah *Equivalence Partitioning*, *Boundary Value Analysis*, *Decision Table Testting*, *State Transition Testing*, *Error Guessing*, dan *Graph-Based Testing*.

2.19 Equivalence Partitioning

Equivalence Partitioning adalah metode pengujian berdasarkan masukan data pada setiap formulir dengan membagi domain masukan ke dalam kelompok valid dan tidak valid [15]. Proses pengujian ini melibatkan beberapa fase. Langkah awal dimulai dengan pembuatan uji kasus (*test case*) perangkat lunak yang memerlukan pengujian. Selanjutnya, setiap kasus uji yang dibuat dieksekusi, dan hasil pengujian didokumentasikan untuk mengevaluasi keefektifannya. Hasil dari pengujian ini kemudian direfleksikan dalam tabel kasus uji, yang berfungsi sebagai referensi untuk memastikan keberhasilan atau kegagalan pengujian. Teknik *equivalence partitioning* disorot karena kemampuannya untuk menentukan kasus uji yang lebih tepat dan mendeteksi kekurangan sistem berdasarkan kasus uji yang dieksekusi.

2.20 Penelitian Terkait

Beberapa penelitian terkait yang menjadi referensi penelitian ini adalah sebagai berikut:

- a. Penelitian yang dilakukan oleh Fariz Sujatmiko dan Dwi Fatrianto Suyatno dengan judul Rancang Bangun Sistem Informatisasi Manajemen Aset Alat Kantor

Berbasis Website Menggunakan *Framework* Laravel dan Metode LIFO [16]. Metode penelitian yang digunakan adalah *Research and Development (R&D)*. Metode LIFO dipilih untuk memprioritaskan aset yang baru ditambah untuk dipinjam. Sistem ini mencakup fitur seperti menampilkan data karyawan, data aset, data gedung dan ruangan, serta fungsi pencarian.

b. Penelitian yang dilakukan oleh Herdian Afrody, Wida Prima Mustika, dan Andi Sanjaya dengan judul Sistem Informulirasi Manajemen Aset (SIMASET) Berbasis *Web*. Metodologi pengembangan yang dilakukan dalam penelitian ini adalah metode *waterfall* dan pemrograman PHP dengan *framework* Laravel [17].

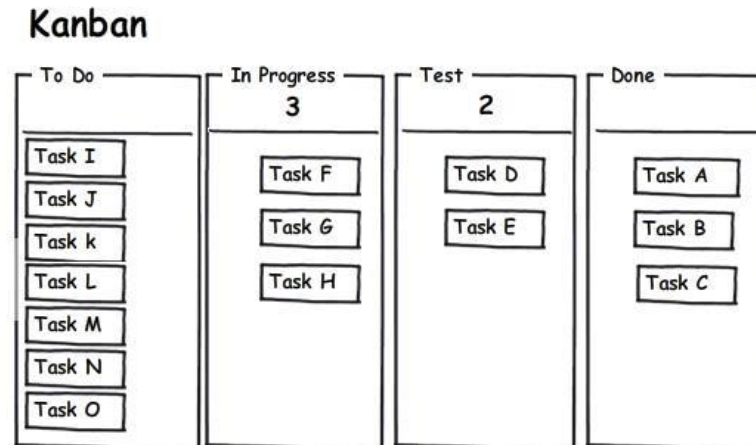
c. Penelitian yang dilakukan oleh Fifit Alfiah, Fuad Muhammad Nur, dan Dedi Gunawan dengan judul Aplikasi Sistem Manajemen Aset Laboratorium Komputer Pada SMKN2 Kab.Tangerang Menggunakan Framework Laravel 8 [18]. Isu utama yang ditemukan dalam penelitian ini adalah kesulitan yang dihadapi sekolah dalam mendapatkan data secara cepat dan rinci mengenai *history* peminjaman ruangan serta pengelolaan aset laboratorium komputer yang masih menggunakan metode konvensional. Metode yang digunakan dalam penelitian ini adalah *Extreme Programming*.

d. Penelitian yang dilakukan oleh Juliana Roshinta dengan judul Pengembangan Sistem Informulirasi Manajemen Aset Dan Keuangan Untuk Meningkatkan Akuntabilitas Manajemen Masjid [19]. Pokok masalah dalam penelitian ini adalah pengelolaan aset dan keuangan yang manual di Masjid Hidayatullah Falah, pendataan aset dan keuangan tidak tertata dengan baik, dan kesulitan dalam melaporkan laporan.

e. Penelitian yang dilakukan oleh Danang Ardiyanto, IGL, dan Putra Eka Prisma dengan judul Implementasi Rancang Bangun Manajemen Aset Berbasis *Website* di PT Petrokimia Gresik [20]. Penelitian ini bertujuan untuk merancang dan mengembangkan sistem manajemen aset berbasis situs *web* serta menilai kinerja situs *web* manajemen aset yang saat ini digunakan oleh PT Petrokimia Gresik. Metodologi yang digunakan dalam penelitian ini adalah *Research and Development (R&D)* dengan menerapkan pendekatan *agile* dalam proses pengembangan, dan pengkodean sistem menggunakan laravel. Penelitian ini menggunakan teknik survei untuk mengumpulkan data yang dibutuhkan.

- f. Penelitian yang dilakukan oleh Ditha Nurcahya Avianty, Andy Hidayat Jatmika, dan Lalu Miftahul Ulum dengan judul Sistem Informulirasi Manajemen Aset pada BPKAD provinsi Nusa Tenggara Barat Berbasis *Website* dengan *Framework* Laravel [21]. Pokok masalah yang terdapat dalam penelitian ini adalah yaitu pelaporan dan monitoring masih dilakukan secara manual dan kurang efektif. Pengembangan sistem dilakukan menggunakan model *waterfall*.
- g. Penelitian yang dilakukan oleh Andry Prayoga, Indrini Dyah Irawati, Muahammd Zero Fernandes dengan materi Perancangan dan Implementasi *Website* Manajemen Pendataan dan Detail Penggunaan Aset Jaringan Iconnet Studi Kasus di PT Indonesia Connets Plus Regional Jawa Barat [22]. Penelitian ini bertujuan untuk memudahkan urusan pendataan dan detail aset jaringan yang ada. Penelitian ini memiliki potensi untuk memberikan kontribusi positif pada manajemen data perusahaan, terutama dalam hal pendataan dan penggunaan aset jaringan. Penelitian ini memberikan gambaran umum tentang website yang dibangun, tetapi kurang memberikan detail teknis tentang arsitektur sistem, *database* yang digunakan, dan teknologi lainnya yang terlibat dalam pengembangan website.
- h. Penelitian yang dilakukan oleh Rini Rizki Junita dengan materi *Asset Management Informuliration System at The Health Service of Aceh Province* [23]. Penelitian ini dibuat untuk merancang sistem informulirasi manajemen aset di Dinas Kesehatan Provinsi Aceh menggunakan Visual BASIC 6.0 dan *Microsoft Office Access* dengan metode *waterfall*. Sistem ini hanya mengelola data aset yaitu menambah, mengedit, dan mencari data aset, serta menyediakan laporan data aset.
- i. Penelitian yang dilakukan oleh Azizah Rahmat dan Nur Aisyah Mohd Hanifah dengan judul penelitian *Usability Testing in Kanban Agile Process for Club Management System* [24]. Penelitian ini membahas pengembangan sistem manajemen klub universitas dengan menerapkan metode Kanban. Tujuan utama sistem ini adalah untuk menjadi sebuah *platform* yang memfasilitasi komunikasi dan manajemen klub serta perkumpulan di lingkungan universitas. Penelitian ini menjelaskan tentang bagaimana Kanban dapat bermanfaat dalam pengembangan perangkat lunak dan bagaimana relevansinya dalam konteks manajemen klub di lingkungan universitas.
- j. Penelitian yang dilakukan oleh Nevenka Kirovska dan Saso Koceski dengan judul

penelitian *Usage Of Kanban Methodology At Software Development Teams* [25]. Artikel ini membahas tentang penggunaan metodologi Kanban dalam pengembangan perangkat lunak di tim pengembangan perangkat lunak. Metodologi Kanban awalnya merupakan konsep manufaktur yang efisien, dan penggunaannya dalam konteks lain terus berkembang karena keberhasilannya yang terbukti.



Gambar II.2. Kanban Board

Dalam penelitian ini, Kanban digunakan sebagai alat untuk memvisualisasikan alur kerja, mengatur jumlah pekerjaan yang sedang dalam proses (*Work-In-Progress*), dan mengelola alur kerja secara efisien. Penggunaan KanbanMAK, sebuah aplikasi berbasis web, telah berhasil meningkatkan proses pengembangan perangkat lunak dengan lebih baik, memungkinkan lebih banyak kolaborasi, dan menghasilkan estimasi waktu yang lebih akurat dalam penyelesaian tugas-tugas pengembangan perangkat lunak.

k. Penelitian yang dilakukan oleh Rayhan Alya Chaerul, Widhy Hayuhardhika Nugraha Putra, dan Buce Trias Hanggara dengan judul penelitian *Utilizing of the Trello API Within the Development of a Monitoring Informuliration System Recording of Project Activities Using a Website-Based Kanban System (Case Study: Electrical Project of PT. XYZ)* [26]. Penelitian ini membahas tentang pengembangan sistem informulirasi pemantauan untuk merekam aktivitas proyek dengan menggunakan API Trello dan sistem Kanban berbasis *website*.

l. Penelitian yang dilakukan oleh Ucu Nugraha dan Tiodor Sianturi dengan judul

penelitian *Blackbox Testing On E-Commerce System Web-Based Evermos (Feature: Registration Experiment & Revamp)* [27]. Penelitian ini menjelajahi penerapan teknik *blackbox testing* pada sistem *e-commerce* yang disebut Evermos, dengan fokus khusus pada fitur registrasi.

m. Penelitian yang dilakukan oleh Daryl J. Powell dengan judul *Kanban for Lean Production in High Mix, Low Volume Environments* [28]. Penelitian ini menjelaskan konsep dasar dari sistem Kanban untuk mengendalikan produksi secara *just-in-time* dengan menggunakan papan dan kartu kanban yang diterapkan dalam pesanan di pabrik sebagai perintah untuk menghasilkan komponen atau produk yang dibutuhkan saat mereka habis.

n. Penelitian yang dilakukan oleh Dwi Yulistiyanti, Tri Yani Akhirina, Thomas Afrizal, Aulia Paramita, dan Naely Farkhatin dengan judul penelitian *Testing Learning Media for English Learning Applications Using BlackBox Testing Based on Equivalence Partitions* [29]. Penelitian ini berfokus pada pengujian aplikasi pembelajaran Bahasa Inggris yaitu Duolingo, menggunakan teknik pengujian *Blackbox* dengan teknik *Equivalence Partitions*.

III. METODE PENELITIAN

3.1 Waktu dan Tempat Penelitian

Penelitian ini dimulai pada semester ganjil tahun ajaran 2023 pada bulan Agustus 2023. Penelitian ini dilaksanakan di PSDKU Universitas Lampung Way Kanan.

Tabel 1 Jadwal Pelaksanaan Penelitian

Kegiatan	Agustus	September	Oktober	November	Desember
Studi Literatur					
<i>Backlog</i>					
<i>Design</i>					
<i>Development</i>					
<i>Testing</i>					
Instalasi ke Server					
Penyusunan Dokumen Akhir					

3.2 Perangkat Penelitian

Perangkat yang digunakan dalam rancang bangun sistem yang akan dibuat dalam penelitian adalah sebagai berikut.

3.2.1 Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan dalam penelitian ini adalah laptop dengan spesifikasi sebagai berikut.

- a. *System Manufacturer* : Dell Inc.
- b. *System Model* : Inspiron 3593

c. *Processor* : Intel® Core™ i7-1065G7 CPU@ 1.30GHz (8 CPUs), ~1,5GHz

3.2.2 Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan dalam penelitian ini adalah sebagai berikut.

- a. Sistem Operasi Windows 11
- b. Laravel *Framework* 10.19.0
- c. Visual Studio Code versi 1.81.1
- d. Google Chrome versi 116.0.5845.97
- e. XAMPP versi 3.3.0
- f. Balsamiq versi 8.1.2
- f. Trello
- g. UMLet version 15.1
- h. VMware vSphere (32 CPUs x Intel® Xeon®, *Memory* 16 GB, *Storage* 2TB)
- i. Ubuntu 23.04

3.3 Tahapan Penelitian

Penelitian ini memiliki beberapa tahapan yang dapat dilihat pada gambar di bawah.



Gambar III.1 Tahapan Penelitian

Gambar III.1 tentang tahapan penelitian dimulai dengan melakukan studi literatur, pengembangan sistem dengan kanban, dan diakhiri dengan penyusunan dokumen akhir. Pada tahap pengembangan sistem dengan kanban, penelitian dimulai dari *backlog*, *design*, *development*, *testing*, dan *done*. *Backlog* adalah tempat dimana semua pekerjaan yang perlu dilakukan dalam proyek yang biasa disebut *task card* (kartu tugas). Bisa berupa tugas, fitur, atau perubahan yang dilakukan. *Backlog* biasanya diurutkan berdasarkan prioritas dengan menentukan terlebih dahulu waktu

penyelesaian. Setelah item dari *backlog* dipilih untuk dikerjakan, kemudian akan masuk ke tahap *design*. *Design* meliputi perencanaan bagaimana sistem atau fitur akan dibangun yang melibatkan rancangan antarmuka pengguna, arsitektur perangkat lunak, dan rincian teknis lainnya. Setelah desain selesai, pekerjaan akan masuk ke tahap *development* (pengembangan). Di tahap ini mulai diimplementasikan fitur yang sesuai dengan spesifikasi yang telah dirancang. Setelah pengembangan selesai, tahap pengujian dimulai. Sistem akan diuji untuk memastikan bahwa semuanya berfungsi dengan baik. *Item* yang sudah selesai dapat dipindahkan ke kolom *done*. Apabila sistem berjalan dengan lancar, langkah selanjutnya adalah menginstalasikannya di server menggunakan VMware vSphere. Ketika pengembangan sistem selesai dilakukan, maka proses selanjutnya adalah perilisan sistem dan diakhiri dengan penulisan dokumen akhir.

3.3.1 Studi Literatur

Tahapan studi literatur dilakukan dengan mengumpulkan, mengevaluasi, dan merangkum sumber-sumber literatur yang berasal dari buku, jurnal, maupun *search engine* yang relevan untuk mencari referensi terkait manajemen aset, Laravel, VMware vSphere, dan sebagainya. Proses ini berperan penting dalam memperoleh pemahaman tentang kerangka kerja teoritis, menemukan pengetahuan yang relevan, serta membangun fondasi teori yang kuat untuk penelitian ini.

3.3.2 Pengembangan Sistem

Pengembangan sistem dalam penelitian ini dilakukan dengan menggunakan metode Kanban yang memiliki tahapan sebagai berikut:

3.3.2.1 Backlog

Pada tahap *backlog* dilakukan analisa terkait kebutuhan pengguna dengan melakukan wawancara kepada Kepala Program Studi PSDKU Unila Way Kanan dan akan melakukan observasi terkait aset yang ada di PSDKU Unila Way Kanan. Berdasarkan hasil wawancara yang telah dilaksanakan, menghasilkan *user stories* yang dimasukkan ke *backlog* sebagaimana ditunjukkan pada tabel 2.

Tabel 2 User Stories

Peran	Fungsi	Prioritas
-------	--------	-----------

Admin	1. Admin dapat melakukan login ke sistem	Tinggi
	2. Admin dapat melihat, menambah, memperbaharui, dan menghapus informulirasi aset	Tinggi
	3. Admin dapat menambah, memperbaharui, dan menghapus kategori aset	Tinggi
	4. Admin dapat menambah, memperbaharui, dan menghapus lokasi aset	Tinggi
	5. Admin dapat menghubungkan aset dengan kategori tertentu	Medium
	6. Admin dapat melihat, menyetujui dan menolak pengajuan peminjaman aset	Tinggi
	7. Admin dapat melihat laporan kerusakan aset	Tinggi
	8. Admin dapat mengelola data perbaikan aset	Tinggi
	9. Admin dapat mengelola riwayat aset	Tinggi
	10. Admin dapat melihat profil	Medium
Staf Aset	1. Staf aset dapat melakukan login ke dalam sistem	Tinggi
	2. Staf aset dapat melihat, menambah, memperbaharui, dan menghapus informulirasi aset	Tinggi
	3. Staf aset dapat melaporkan kerusakan aset	Tinggi
	4. Staf aset dapat melakukan peminjaman aset	Tinggi
	5. Staf aset dapat melihat profil	Medium

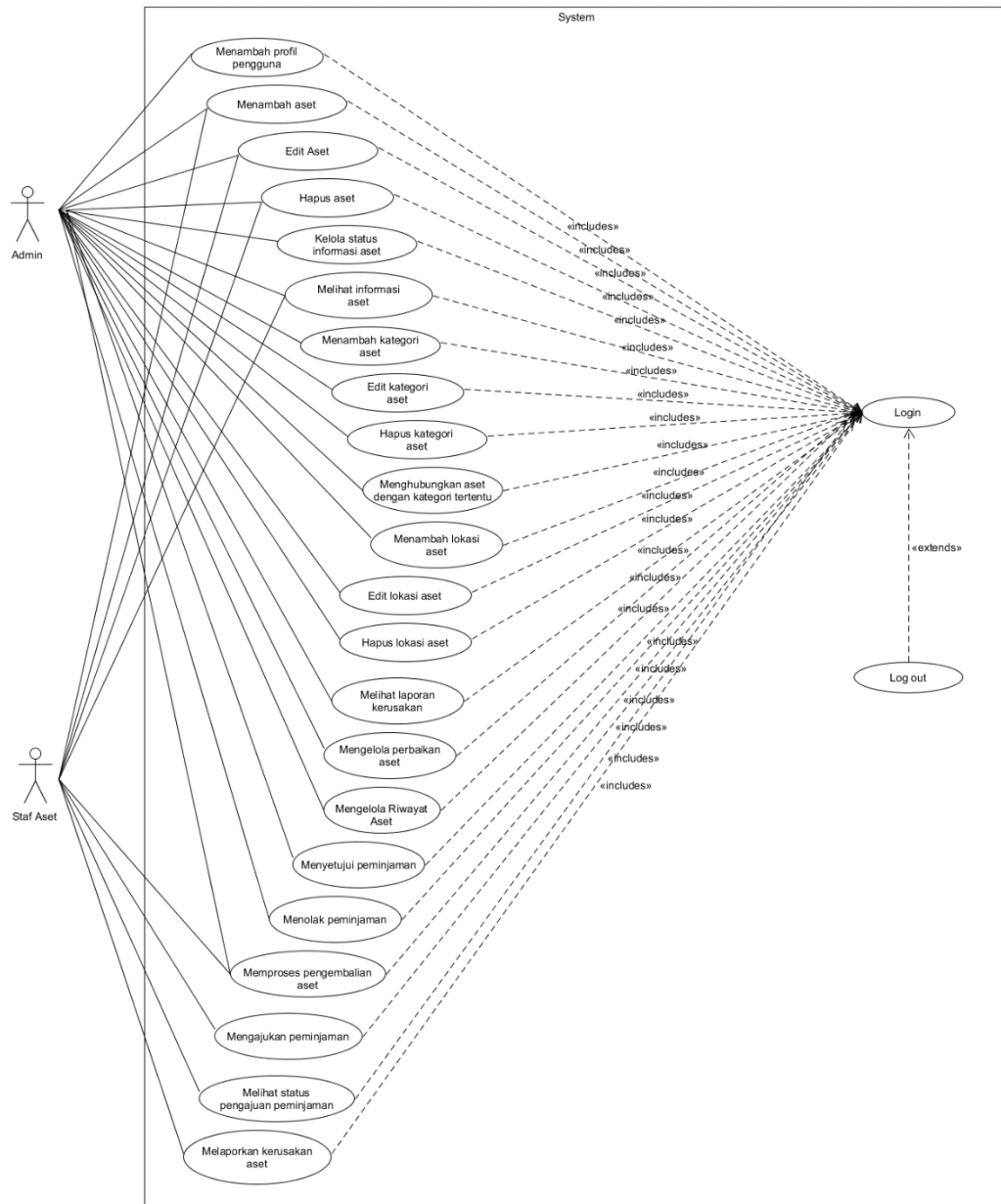
3.3.2.2 Design

Pada tahap ini, dilakukan pemodelan sistem dalam bentuk *use case diagram*, *activity diagram*, dan *entity relationship diagram* menggunakan UMLet serta desain antarmuka sistem dalam bentuk *mockup* dengan menggunakan Balsamiq *Wireframes*.

a. Use Case Diagram

Use case diagram dibuat berdasarkan *user stories* pada tahapan analisis kebutuhan.

Hal apa saja yang dapat dilakukan oleh admin dan staf aset digambarkan dalam *use case diagram* berikut.



Gambar III.2 Use Case Diagram

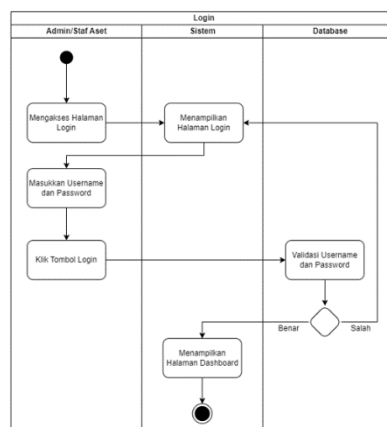
Gambar III.2 terdapat 2 aktor yang dapat mengakses sistem, yaitu admin dan staf aset. Admin dan staf aset harus melakukan *login* terlebih dahulu. Admin sebagai aktor yang bertanggung jawab atas manajemen aset di PSDKU Universitas Lampung Way Kanan memiliki akses penuh ke sistem dalam mengelola semua

aset, pengguna, lokasi, dan kategori. Admin memiliki wewenang dalam meninjau laporan kerusakan aset yang dilaporkan oleh pengguna lain. Admin juga memiliki wewenang dalam meninjau pengajuan peminjaman aset dari mahasiswa ataupun dosen, menyetujui atau menolak pengajuan peminjaman. Admin juga dapat memproses pengembalian aset yang telah dipinjamkan, mengelola data perbaikan aset, dan mengelola riwayat aset. Staf aset merupakan aktor yang bertugas membantu admin dalam manajemen aset. Staf aset memiliki akses terbatas untuk mengelola kategori dan lokasi aset, dan melihat laporan kerusakan. Staf aset memiliki wewenang dalam menambah, memperbaharui, dan menghapus aset dengan izin terbatas oleh admin. Staf aset juga memiliki wewenang untuk melaporkan kerusakan aset, memproses pengembalian aset, dan mengajukan peminjaman aset.

b. Activity Diagram

Activity diagram digunakan untuk memodelkan aliran aktivitas atau tugas dalam suatu sistem atau proses. *Activity diagram* adalah alat penting dalam pemodelan dan analisis sistem yang membantu dalam memahami, merancang, dan mengelola proses atau aktivitas dalam berbagai konteks, baik dalam pengembangan perangkat lunak maupun dalam analisis bisnis. Berikut adalah *activity diagram* sistem informulirasi manajemen aset.

1) Activity Diagram Login

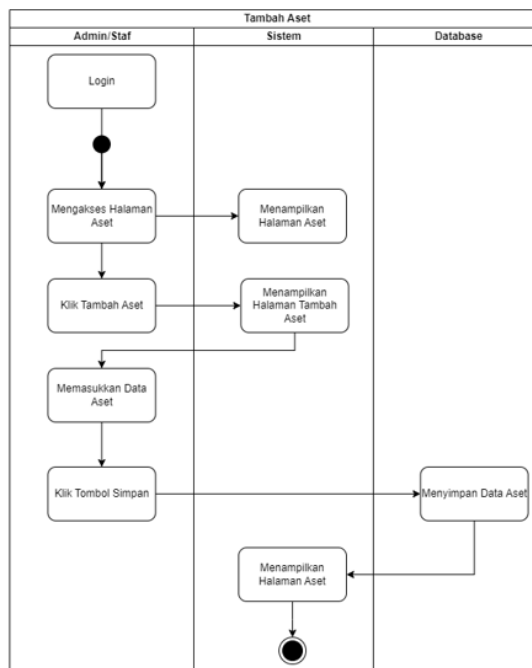


Gambar III.3 Activity Diagram Login

Gambar III.3 menggambarkan langkah-langkah yang dilakukan oleh pengguna, baik admin maupun staf aset melakukan proses *login*. Tahap awal dalam melakukan

login adalah pengguna mengakses URL dari halaman *login*, kemudian sistem menampilkan formulir halaman *login*. Pengguna kemudian mengisi formulir dengan memasukkan *username* dan *password* yang sudah terdaftar. Setelah formulir terisi, pengguna menekan tombol *login*. Proses melibatkan pengecekan atau autentikasi pada *database*. Apabila *username* dan *password* yang dimasukkan tidak sesuai, maka sistem akan menampilkan kembali formulir *login*, dan jika sesuai, maka sistem akan menampilkan halaman *dashboard*.

2) Activity Diagram Menambah Aset

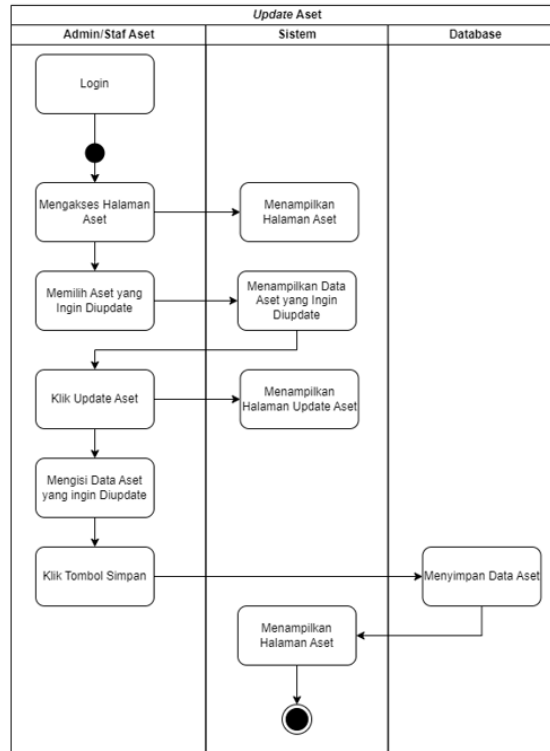


Gambar III.4 Activity Diagram Menambah Aset

Gambar III.4 menggambarkan proses penambahan aset oleh admin dan staf aset. Proses ini dimulai ketika admin dan staf aset mengakses halaman aset. Setelah mengakses halaman tersebut, sistem akan menampilkan halaman aset. Selanjutnya, admin dan staf aset dapat mengklik tombol tambah untuk mengakses halaman formulir penambahan aset. Di halaman formulir ini, mereka dapat mengisi semua data yang diperlukan terkait aset yang akan ditambahkan. Setelah semua data telah diisi, admin dan staf aset dapat mengklik tombol simpan. Dengan melakukan itu, data aset akan disimpan ke dalam *database*. Setelah proses penyimpanan selesai, sistem akan secara otomatis mengarahkan kembali ke halaman aset, dan dapat

melihat aset yang baru saja ditambahkan dengan berhasil.

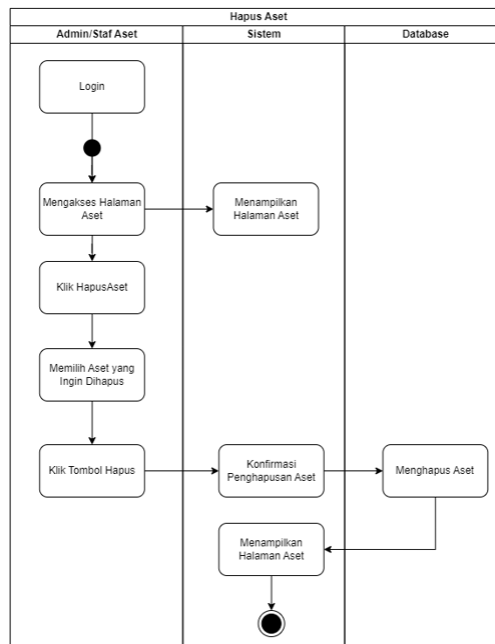
3) Activity Diagram Update Aset



Gambar III.5 Activity Diagram Update Aset

Gambar III.5 menggambarkan proses pembaharuan data aset oleh admin dan staf aset. Proses ini dimulai ketika admin dan staf aset mengakses halaman aset. Setelah mengakses halaman tersebut, sistem akan menampilkan halaman aset. Kemudian, admin atau staf aset dapat memilih aset yang ingin di-*update*. Sistem akan menampilkan halaman aset yang ingin di-*update*. Selanjutnya, admin dan staf aset dapat mengklik tombol *update* untuk mengakses halaman update aset. Admin atau staf aset dapat mengisi data aset yang ingin di-*update*. Setelah pengisian data selesai, admin atau staf aset dapat mengklik tombol simpan. Dengan melakukan itu, data aset akan disimpan ke dalam *database*. Setelah proses penyimpanan selesai, sistem akan secara otomatis mengarahkan kembali ke halaman aset, dan dapat melihat aset yang baru saja di-*update* dengan berhasil.

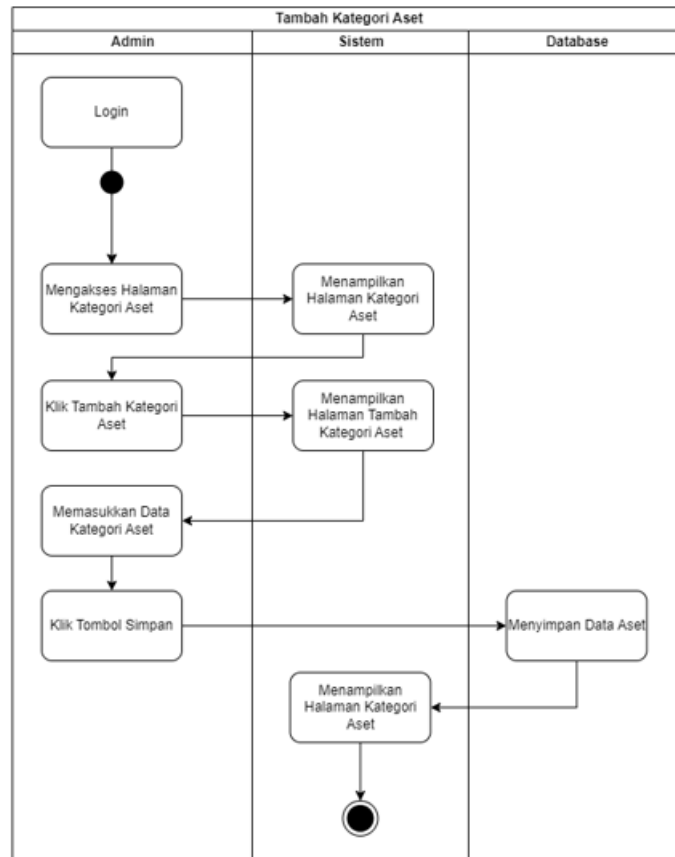
4) Activity Diagram Hapus Aset



Gambar III.6 Activity Diagram Hapus Aset

Gambar III.6 menggambarkan proses penghapusan aset oleh admin dan staf aset. Proses ini dimulai ketika admin dan staf aset mengakses halaman aset. Setelah mengakses halaman tersebut, sistem akan menampilkan halaman aset. Kemudian, admin atau staf aset dapat memilih aset yang ingin dihapus. Sistem akan menampilkan halaman aset yang ingin dihapus. Selanjutnya, admin dan staf aset dapat mengklik tombol hapus untuk menghapus aset. Sistem akan melakukan validasi apakah aset benar ingin dihapus atau tidak. Jika benar, maka data aset di *database* berhasil terhapus dan sistem akan mengarahkan kembali ke halaman aset.

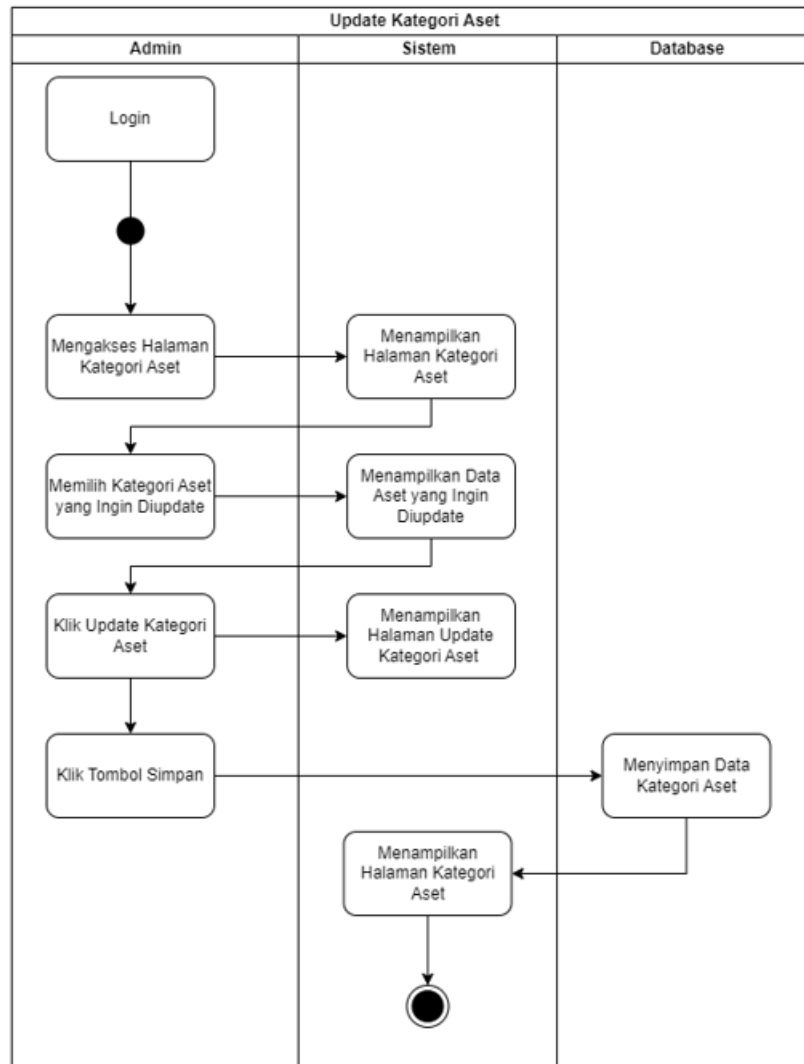
5) Activity Diagram Tambah Kategori Aset



Gambar III.7 Activity Diagram Tambah Kategori Aset

Gambar III.7 menggambarkan proses penambahan kategori aset oleh admin. Proses ini dimulai ketika admin mengakses halaman kategori aset. Setelah mengakses halaman tersebut, sistem akan menampilkan halaman kategori aset. Selanjutnya, admin dapat mengklik tombol tambah untuk mengakses halaman formulir penambahan kategori aset. Di halaman formulir ini, admin dapat mengisi semua data yang diperlukan terkait kategori aset yang akan ditambahkan. Setelah semua data telah diisi, admin dapat mengklik tombol simpan. Dengan melakukan itu, data kategori aset akan disimpan ke dalam *database*. Setelah proses penyimpanan selesai, sistem akan secara otomatis mengarahkan kembali ke halaman kategori aset, dan dapat melihat aset yang baru saja ditambahkan dengan berhasil.

6) Activity Diagram Update Kategori Aset

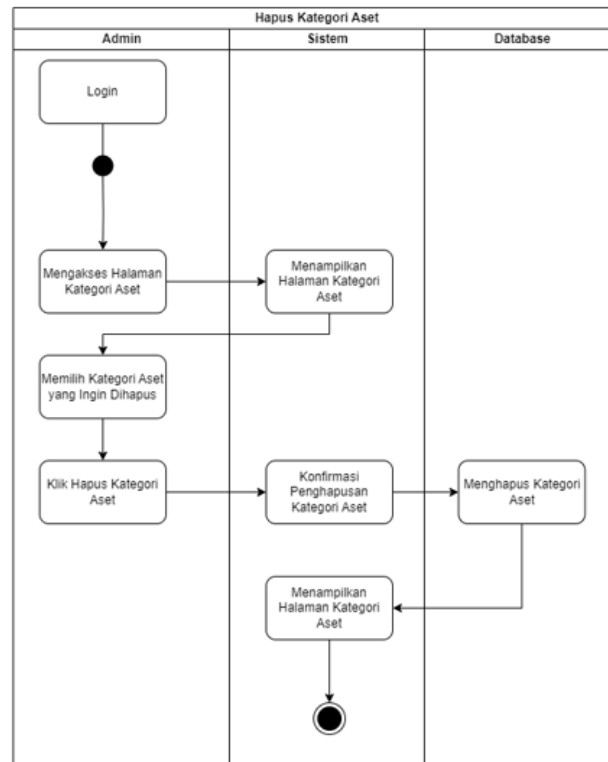


Gambar III.8 Activity Diagram Update Kategori Aset

Gambar III.8 menggambarkan proses pembaharuan data kategori aset oleh admin. Proses ini dimulai ketika admin mengakses halaman kategori aset. Setelah mengakses halaman tersebut, sistem akan menampilkan halaman kategori aset. Kemudian, admin dapat memilih kategori aset yang ingin di-*update*. Sistem akan menampilkan halaman kategori aset yang ingin di-*update*. Selanjutnya, admin dapat mengklik tombol *update* untuk mengakses halaman *update* aset. Admin dapat mengisi data kategori aset yang ingin di-*update*. Setelah pengisian data selesai, admin dapat mengklik tombol simpan. Dengan melakukan itu, data aset akan disimpan ke dalam *database*. Setelah proses penyimpanan selesai, sistem akan

secara otomatis mengarahkan kembali ke halaman kategori aset, dan dapat melihat aset yang baru saja di-update dengan berhasil.

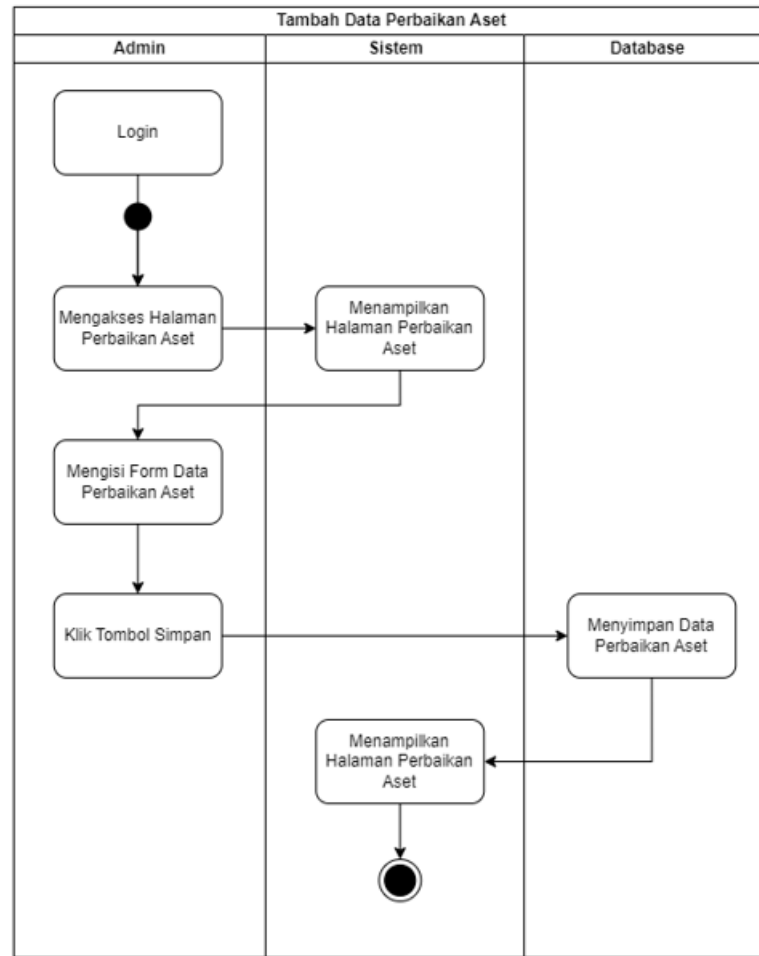
7) Activity Diagram Hapus Kategori Aset



Gambar III.9 Activity Diagram Hapus Kategori Aset

Gambar III.9 menggambarkan proses penghapusan kategori aset oleh admin. Proses ini dimulai ketika admin mengakses halaman kategori aset. Setelah mengakses halaman tersebut, sistem akan menampilkan halaman kategori aset. Kemudian, admin dapat memilih kategori aset yang ingin dihapus. Sistem akan menampilkan halaman kategori aset yang ingin dihapus. Selanjutnya, admin dapat mengklik tombol hapus untuk menghapus aset. Sistem akan melakukan validasi apakah kategori aset benar ingin dihapus atau tidak. Jika benar, maka data kategori aset di *database* berhasil terhapus dan sistem akan mengarahkan kembali ke halaman kategori aset.

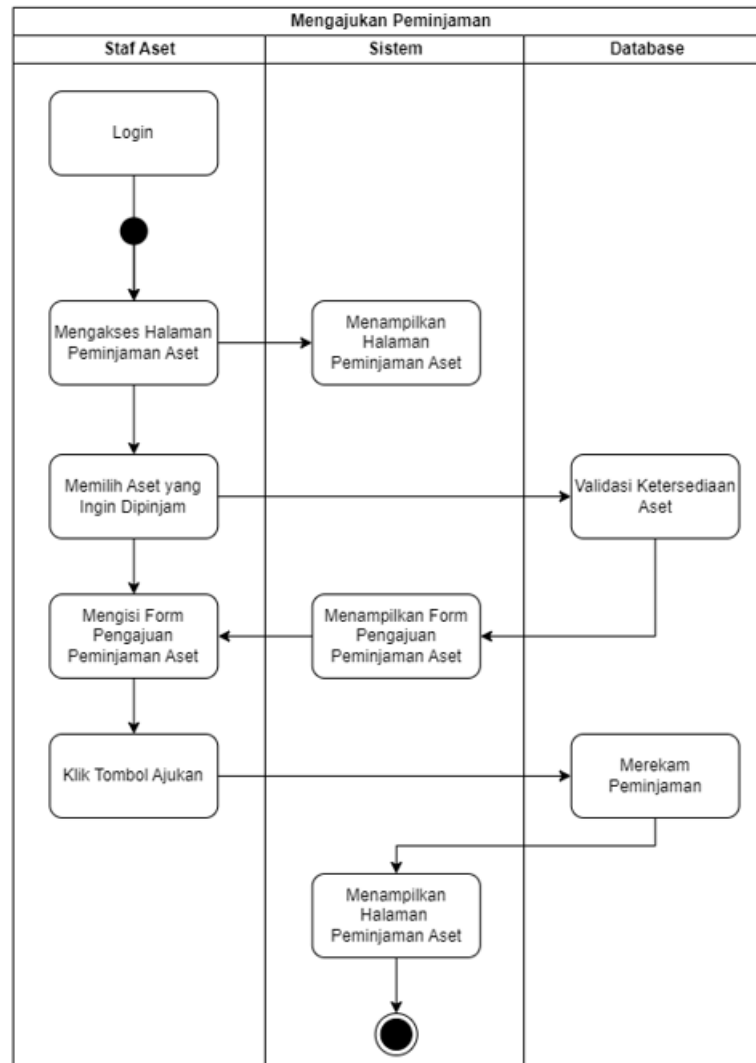
8) *Activity Diagram* Tambah Data Perbaikan Aset



Gambar III.10 *Activity Diagram* Tambah Data Perbaikan Aset

Gambar III.10 menggambarkan proses untuk menambah data perbaikan aset oleh admin. Proses ini dimulai ketika admin mengakses halaman perbaikan aset. Setelah mengakses halaman tersebut, sistem akan menampilkan halaman perbaikan aset. Kemudian, admin dapat mengklik tombol tambah. Sistem akan menampilkan halaman formulir tambah data perbaikan aset. Selanjutnya, admin dapat mengklik tombol simpan. *Database* akan menyimpan data perbaikan aset, dan sistem akan mengakses kembali halaman perbaikan aset.

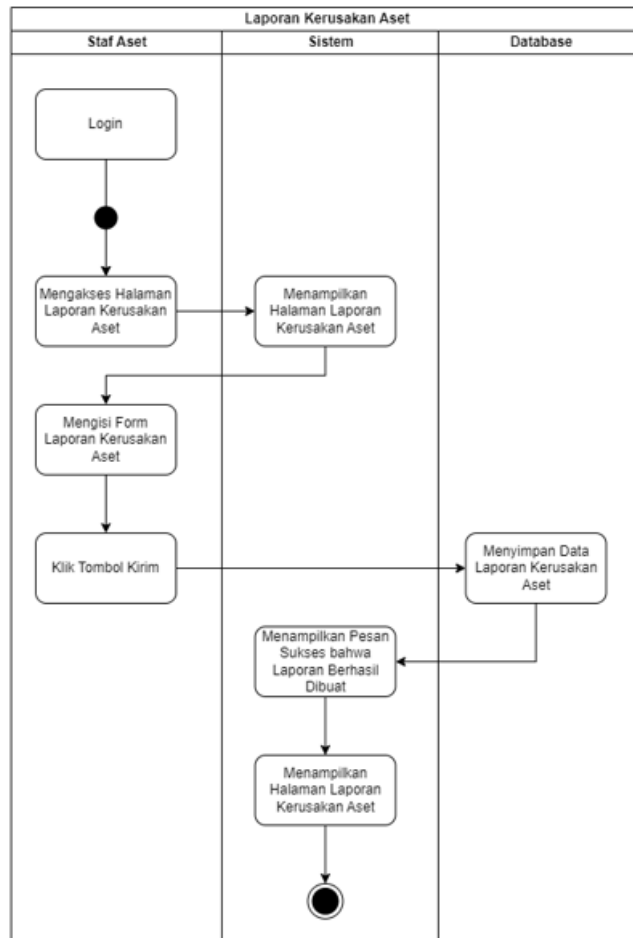
8) Activity Diagram Mengajukan Peminjaman Aset



Gambar III.11 Activity Diagram Mengajukan Peminjaman Aset

Gambar III.11 menggambarkan proses pengajuan aset oleh staf aset. Proses ini dimulai ketika staf aset mengakses halaman peminjaman aset. Setelah mengakses halaman tersebut, sistem akan menampilkan halaman peminjaman aset. Kemudian, staf aset dapat memilih aset yang ingin dipinjam. *Database* akan memvalidasi ketersediaan aset terlebih dahulu. Jika aset tersedia, maka sistem akan menampilkan formulir pengajuan peminjaman aset. Staf aset dapat mengisi formulir pengajuan peminjaman aset tersebut. Setelah formulir selesai diisi maka, staf aset dapat mengklik tombol ajukan. *Database* akan merekam pengajuan peminjaman aset. Kemudian, sistem akan mengarahkan kembali ke halaman peminjaman aset.

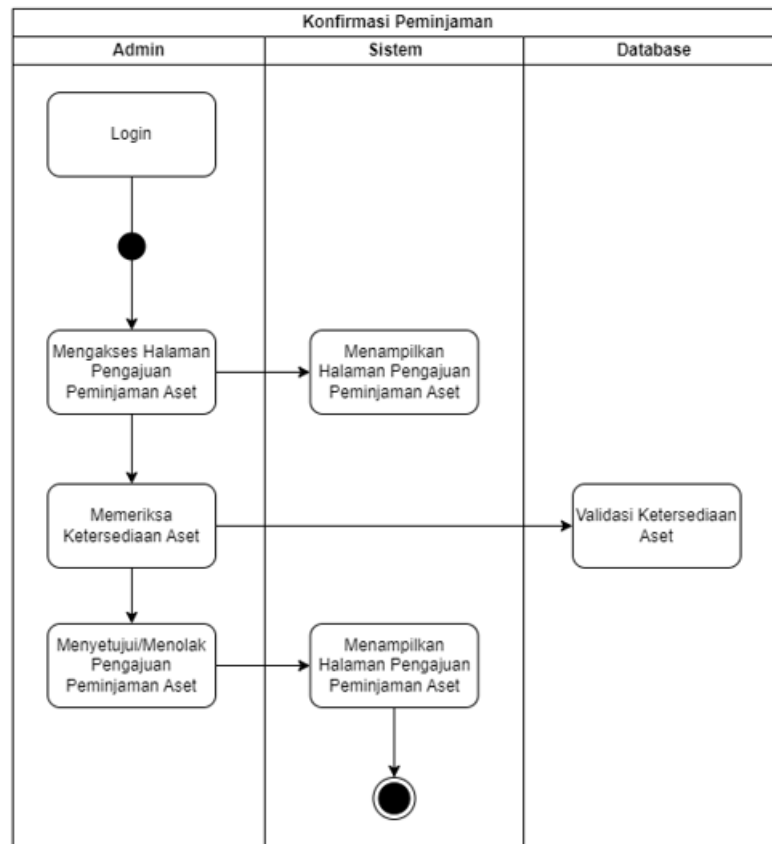
9) Activity Diagram Pengajuan Laporan Kerusakan Aset



Gambar III.12 Activity Diagram Pengajuan Laporan Kerusakan Aset

Gambar III.12 menggambarkan proses pengajuan laporan kerusakan aset oleh staf aset. Proses ini dimulai ketika staf aset mengakses halaman pengajuan laporan kerusakan aset. Setelah mengakses halaman tersebut, sistem akan menampilkan halaman pengajuan laporan kerusakan aset. Kemudian, staf aset dapat mengisi formulir laporan kerusakan aset. Setelah selesai mengisi formulir, staf aset dapat mengklik tombol kirim. *Database* akan langsung menyimpan data laporan kerusakan aset. Lalu, sistem akan menampilkan pesan sukses bahwa laporan berhasil dibuat dan mengarahkan kembali ke halaman laporan kerusakan aset.

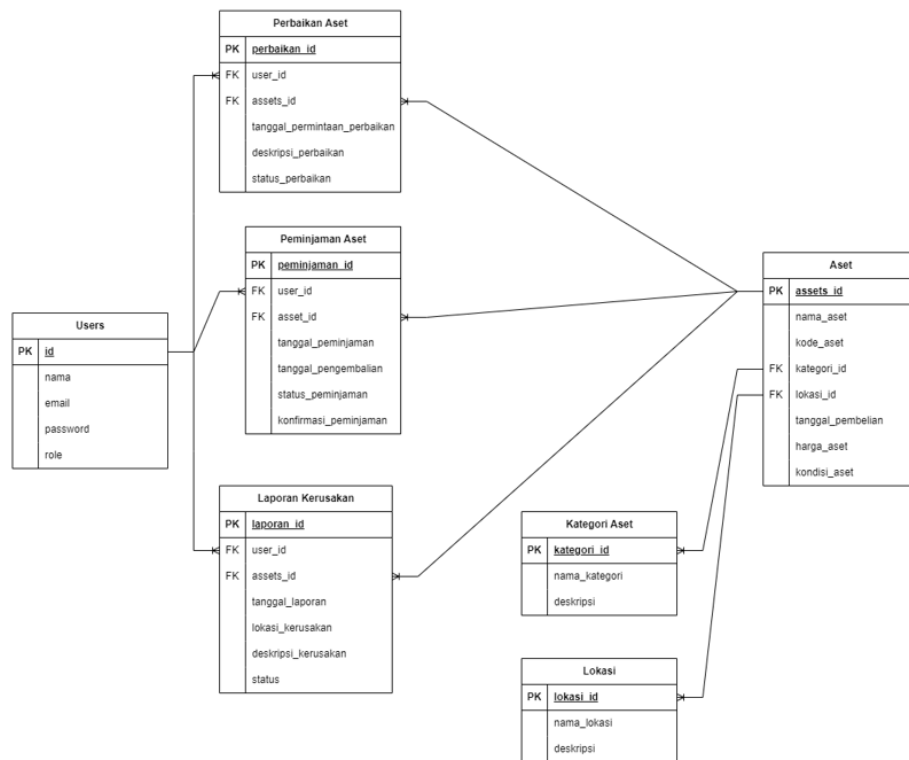
10) *Activity Diagram* Konfirmasi Pengajuan Peminjaman Aset



Gambar III.13 *Activity Diagram* Konfirmasi Pengajuan Peminjaman Aset

Gambar III.13 menggambarkan proses konfirmasi pengajuan peminjaman aset oleh admin. Proses ini dimulai ketika admin mengakses halaman pengajuan peminjaman aset. Setelah mengakses halaman tersebut, sistem akan menampilkan halaman pengajuan peminjaman aset. Kemudian, admin akan memeriksa ketersediaan aset dan *database* akan memvalidasi apakah aset tersebut masih tersedia atau tidak. Admin dapat menyetujui atau menolak pengajuan peminjaman aset. Setelah melakukan konfirmasi pengajuan peminjaman aset, sistem akan mengarahkan kembali ke halaman pengajuan peminjaman aset.

c. Entity Relationship Diagram



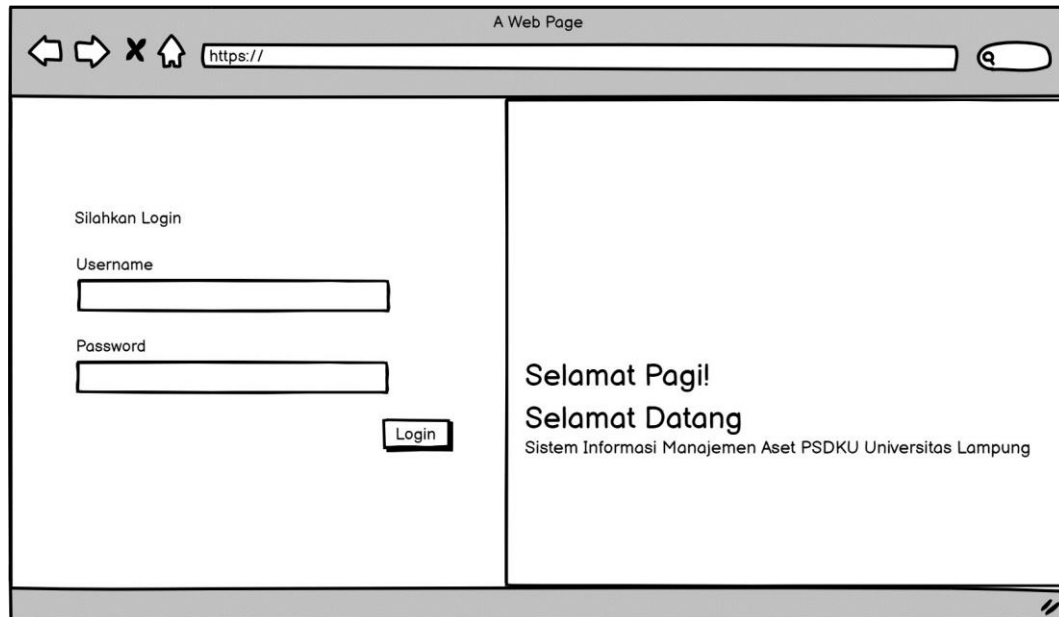
Gambar III.14 Entity Relationship Diagram

Gambar III.14 menampilkan *Entity Relationship Diagram* (ERD) yang digunakan untuk merepresentasikan struktur *database*. Dalam sistem ini, terdapat lima tabel yang meliputi tabel pengguna (*users*), perbaikan aset, peminjaman aset, laporan kerusakan, kategori, lokasi, dan aset. Tabel-tabel ini memiliki keterhubungan satu sama lain, yaitu tabel *users* berhubungan dengan tabel peminjaman aset. Tabel aset terkait dengan tabel perbaikan aset, peminjaman aset, dan laporan kerusakan. Sementara itu, laporan kerusakan memiliki hubungan dengan tabel perbaikan aset.

d. Mockup

Mockup adalah gambaran visual dari tampilan atau desain awal suatu produk atau proyek. *Mockup* sering digunakan untuk memberikan gambaran kasar tentang bagaimana produk atau situs web akan terlihat dan berfungsi sebelum desain akhir dan pengembangan sebenarnya dimulai.

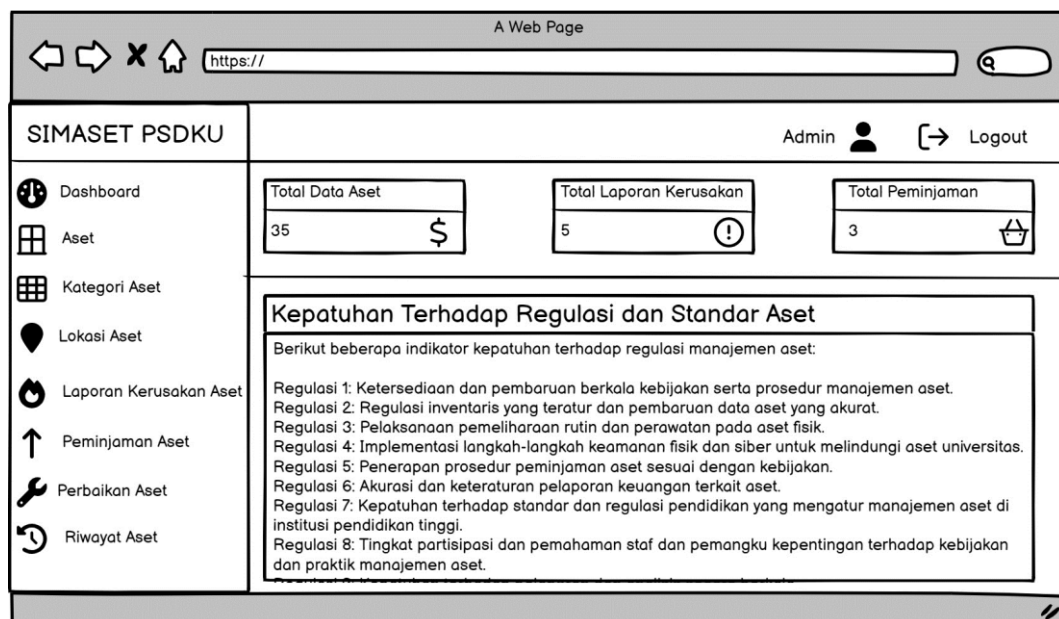
1) Mockup Login



Gambar III.15 Mockup Login

Gambar III.15 adalah tampilan awal saat pengguna mengakses sistem, yaitu *mockup* halaman *login*. Halaman ini memiliki dua kotak input, yaitu satu untuk *username* dan satu lagi untuk *password*, serta sebuah tombol *login*.

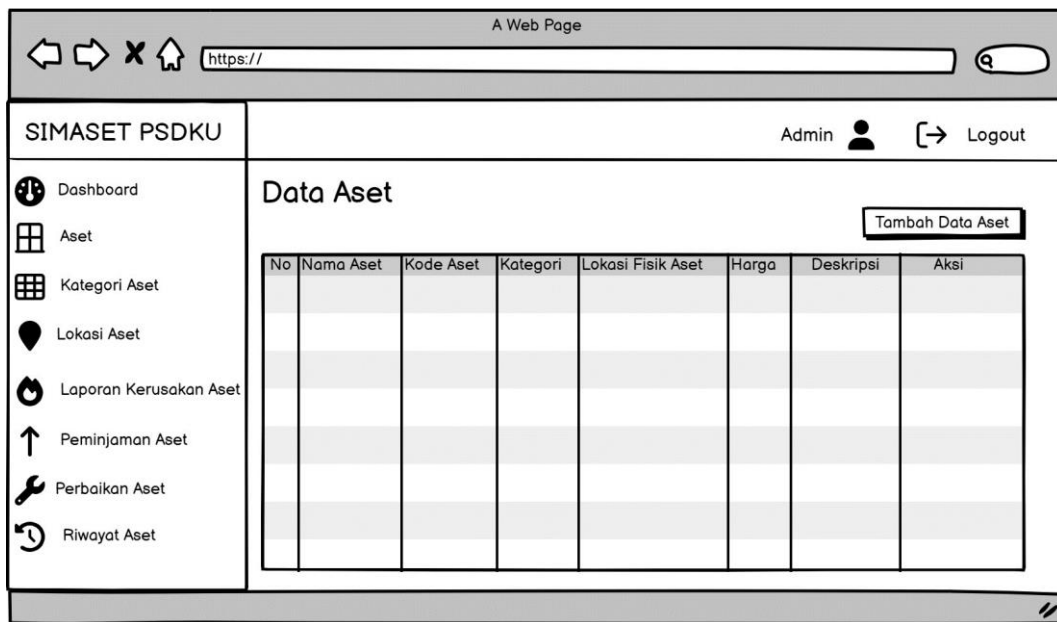
2) Mockup Beranda



Gambar III.16 Mockup Beranda

Gambar III.16 adalah tampilan saat admin berhasil login, yaitu *mockup* halaman beranda. Di halaman ini terdapat tampilan total aset, total laporan kerusakan, total peminjaman aset dan terdapat informasi mengenai kepatuhan terhadap regulasi dan standar aset.

3) *Mockup* Data Aset



Gambar III.17 *Mockup* Data Aset

Gambar III.17 menampilkan *mockup* halaman data aset yang dapat diakses oleh admin dan staf aset. Pada halaman ini, terdapat informasi-informasi mengenai aset, seperti nama aset, kode aset, kategori aset, lokasi aset, harga, deskripsi dan pilihan aksi, seperti melihat *detail*, mengedit, atau menghapus aset. Admin juga memiliki kemampuan untuk menambahkan aset baru langsung dari halaman ini.

4) *Mockup* Tambah Aset

The mockup shows a web browser window with the URL 'https://'. The page title is 'SIMASET PSDKU'. The user is logged in as 'Admin' and has a 'Logout' button. The sidebar contains the following menu items: Dashboard, Aset, Kategori Aset, Lokasi Aset, Laporan Kerusakan Aset, Peminjaman Aset, Perbaikan Aset, and Riwayat Aset. The main content area is titled 'Tambah Data Aset' and contains the following form fields:

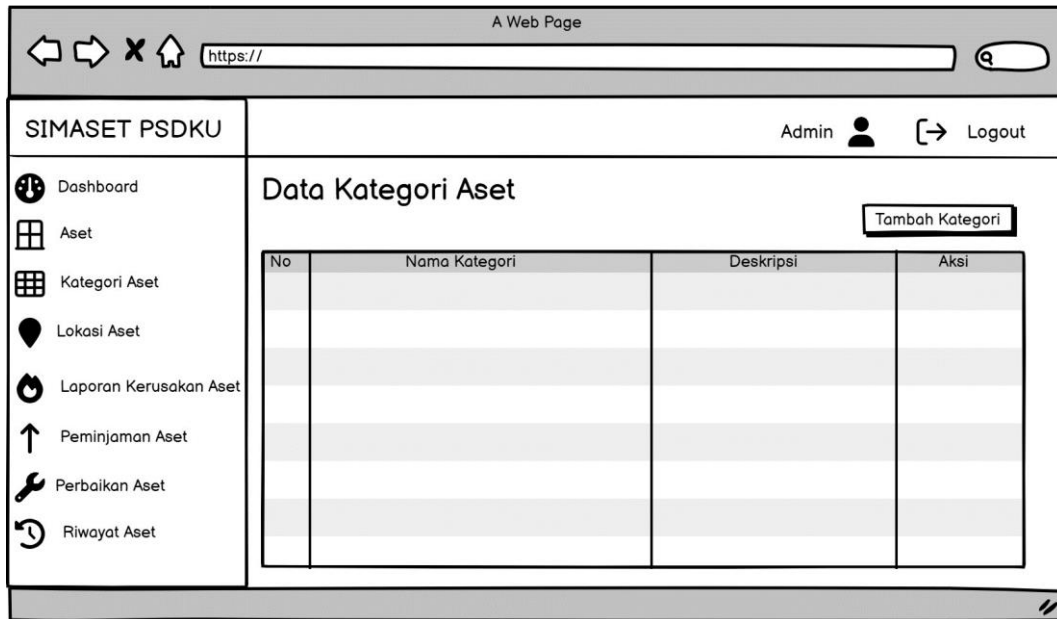
- Nama Aset
- Kode Aset
- Tanggal Pembelian
- Kategori
- Lokasi Fisik Aset
- Harga
- Merk
- Status Kepemilikan
- Kondisi Aset
- Deskripsi

A 'Submit' button is located at the bottom right of the form.

Gambar III.18 *Mockup* Tambah Aset

Gambar III.18 menampilkan *mockup* halaman untuk menambah aset yang dapat diakses oleh admin dan staf aset. Admin dan staf aset dapat mengisi nama aset, kode aset, tanggal pembelian, kategori, lokasi fisik aset, harga, merk, status kepemilikan, kondisi aset, dan deskripsi.

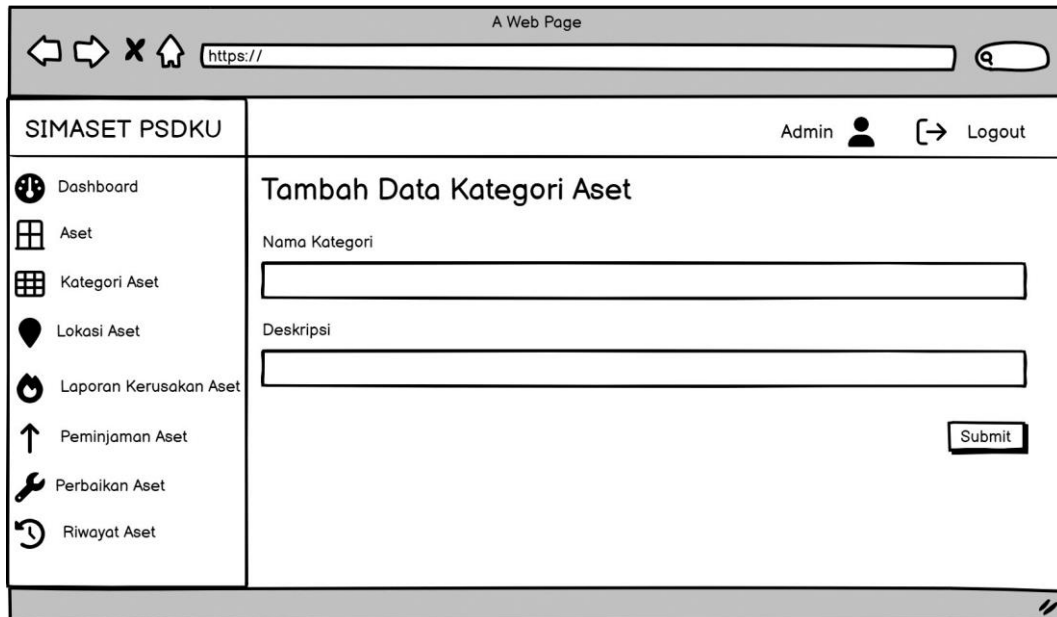
5) Mockup Kategori Aset



Gambar III.19 Mockup Kategori Aset

Gambar III.19 menampilkan *mockup* halaman untuk mengakses data kategori aset yang dapat diakses oleh admin. Data tersebut berisi nama kategori aset, deskripsi, dan pilihan aksi untuk mengedit dan menghapus aset. Admin juga memiliki kemampuan untuk menambahkan data kategori aset baru langsung dari halaman ini.

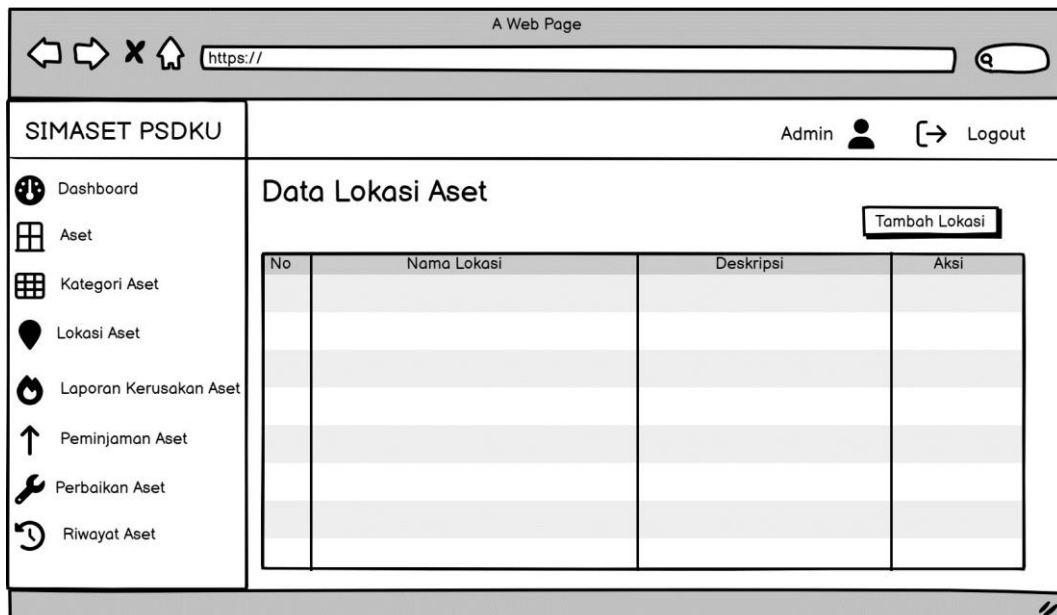
6) Mockup Tambah Kategori Aset



Gambar III.20 Mockup Tambah Kategori Aset

Gambar III.20 menampilkan *mockup* halaman untuk menambah data kategori aset yang dapat diakses oleh admin. Admin dapat mengisi nama kategori dan deskripsi.

7) Mockup Data Lokasi Aset



Gambar III.21 Mockup Data Lokasi Aset

Gambar III.21 menampilkan *mockup* halaman untuk data lokasi aset yang dapat

diakses oleh admin. Pada halaman ini, terdapat informasi-informasi mengenai nama lokasi, deskripsi, dan pilihan aksi untuk mengedit dan menghapus aset. Admin juga memiliki kemampuan untuk menambahkan data lokasi aset baru langsung dari halaman ini.

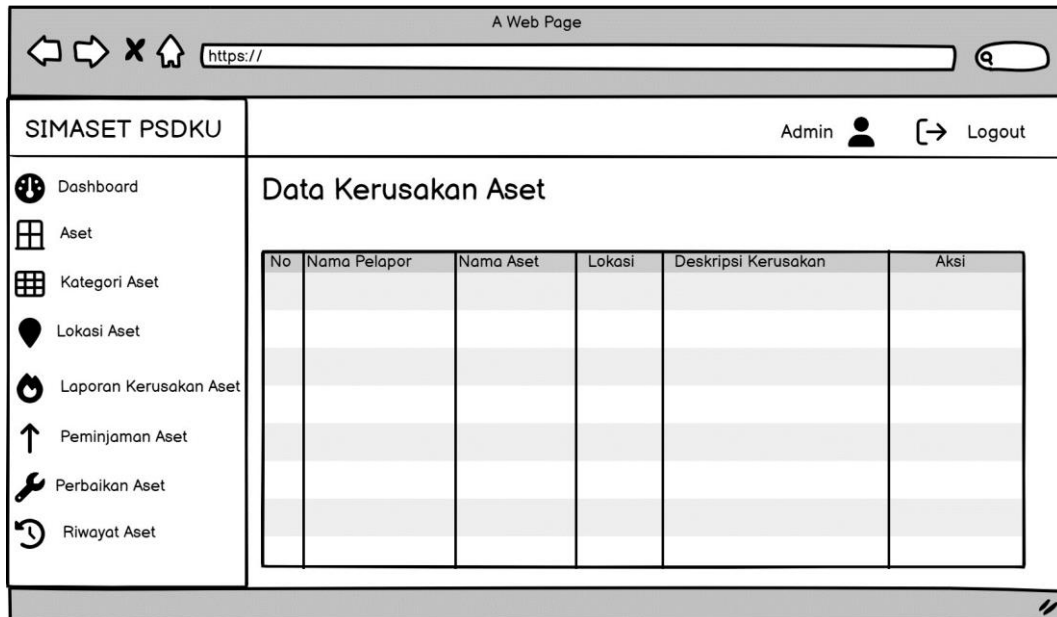
8) *Mockup* Tambah Data Lokasi Aset

The mockup shows a web browser window titled "A Web Page" with a URL bar containing "https://". The page content is divided into a sidebar and a main area. The sidebar, titled "SIMASET PSDKU", contains a list of menu items: Dashboard, Aset, Kategori Aset, Lokasi Aset, Laporan Kerusakan Aset, Peminjaman Aset, Perbaikan Aset, and Riwayat Aset. The main area is titled "Tambah Data Lokasi Aset" and contains two text input fields labeled "Nama Lokasi" and "Deskripsi". A "Submit" button is located at the bottom right of the form. The top right of the page shows the user is logged in as "Admin" with a "Logout" link.

Gambar III.22 Mockup Tambah Data Lokasi Aset

Gambar III.22 menampilkan *mockup* halaman untuk menambah aset yang dapat diakses oleh admin. Admin dapat mengisi nama lokasi dan deskripsi.

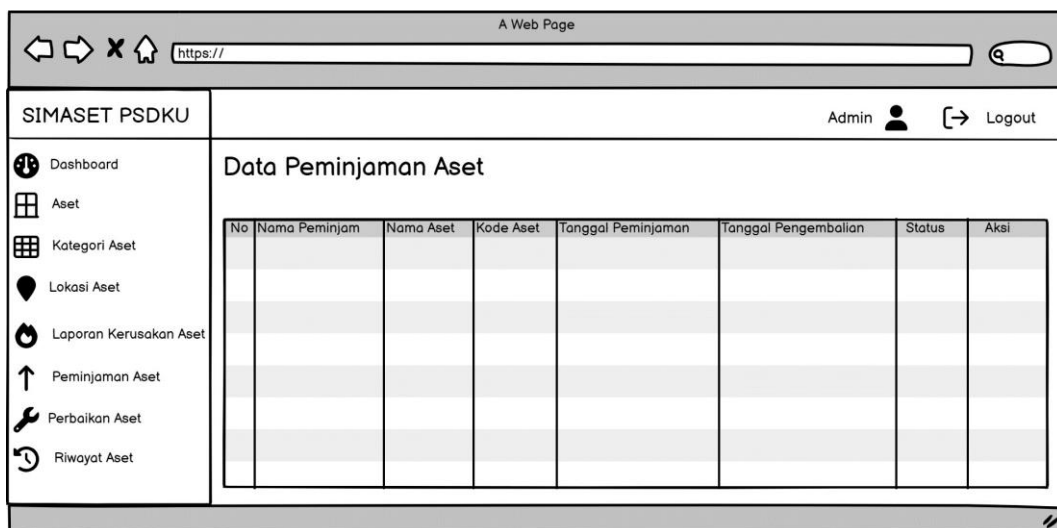
9) Mockup Laporan Kerusakan Aset



Gambar III.23 Mockup Laporan Kerusakan Aset

Gambar III.23 menampilkan *mockup* halaman untuk laporan kerusakan aset yang hanya dapat diakses oleh admin. Pada halaman ini, terdapat informasi nama pelapor, nama aset, lokasi kerusakan, dan deskripsi kerusakan aset.

10) Mockup Peminjaman Aset



Gambar III.24 Mockup Peminjaman Aset

Gambar III.24 menampilkan *mockup* halaman untuk peminjaman aset yang dapat diakses oleh admin. Pada halaman ini, terdapat informasi nama peminjam, nama

aset, kode aset, tanggal peminjaman, tanggal pengembalian, status dan pilihan aksi detail, menyetujui atau menolak pengajuan peminjaman aset.

11) *Mockup* Pengajuan Peminjaman Aset

The mockup shows a web browser window titled "A Web Page" with a URL bar containing "https://". The application header is "SIMASET PSDKU" and includes a "Logout" button. The sidebar lists: Dashboard, Aset, Laporan Kerusakan Aset, and Peminjaman Aset. The main form is titled "Tambah Peminjaman Aset" and includes the following fields: Nama Peminjam, Nama Aset, Kode Aset, Tanggal Peminjaman, Tanggal Pengembalian, and Keterangan. A "Submit" button is positioned at the bottom right of the form.

Gambar III.25 *Mockup* Pengajuan Peminjaman Aset

Gambar III.25 menampilkan *mockup* halaman pengajuan peminjaman aset yang dapat diakses oleh staf aset. Pada halaman ini, terdapat formulir yang mencakup nama peminjam, nama aset, kode aset, tanggal peminjaman, dan keterangan peminjaman aset.

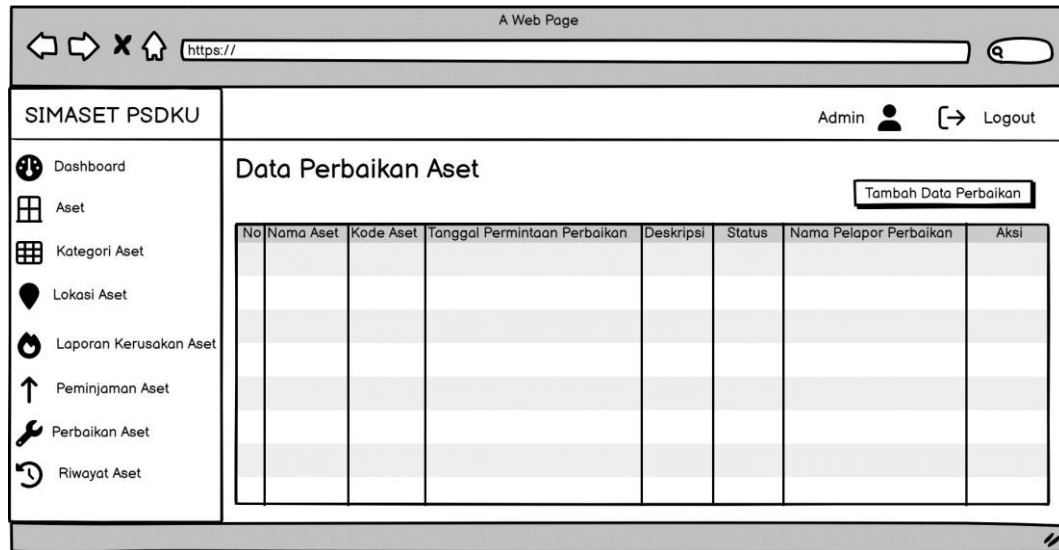
12) Mockup Pengajuan Laporan Kerusakan Aset

The image shows a web browser window displaying a form for reporting asset damage. The browser's address bar shows 'https://'. The page title is 'SIMASET PSDKU'. In the top right corner, there is a 'Staf' label with a user icon and a 'Logout' button. The left sidebar contains a menu with the following items: 'Dashboard', 'Aset', 'Laporan Kerusakan Aset' (highlighted), and 'Peminjaman Aset'. The main content area is titled 'Kerusakan Aset' and contains the following form fields: 'Nama Pelapor', 'Nama Aset', 'Tanggal Laporan', 'Lokasi', and 'Deskripsi'. A 'Submit' button is located at the bottom right of the form area.

Gambar III.26 Mockup Pengajuan Laporan Kerusakan Aset

Gambar III.26 menampilkan *mockup* halaman pengajuan laporan kerusakan yang dapat diakses oleh staf aset dan dosen. Pada halaman ini, terdapat formulir yang mencakup nama pelapor, nama aset, tanggal laporan, lokasi kerusakan, dan deskripsi kerusakan.

13) Mockup Perbaikan Aset



Gambar III.27 Mockup Perbaikan Aset

Gambar III.27 menampilkan *mockup* halaman perbaikan aset yang dapat diakses oleh admin. Pada halaman ini, terdapat yang mencakup nama aset, kode aset, tanggal permintaan perbaikan, deskripsi perbaikan, status, dan nama pelapor perbaikan.

14) *Mockup* Tambah Data Perbaikan Aset

The mockup shows a web browser window with the URL 'https://'. The page title is 'SIMASET PSDKU'. In the top right corner, it displays 'Admin' with a user icon and a 'Logout' button. The left sidebar contains the following menu items: Dashboard, Aset, Kategori Aset, Lokasi Aset, Laporan Kerusakan Aset, Peminjaman Aset, Perbaikan Aset, and Riwayat Aset. The main content area is titled 'Tambah Perbaikan Aset' and contains the following form fields:

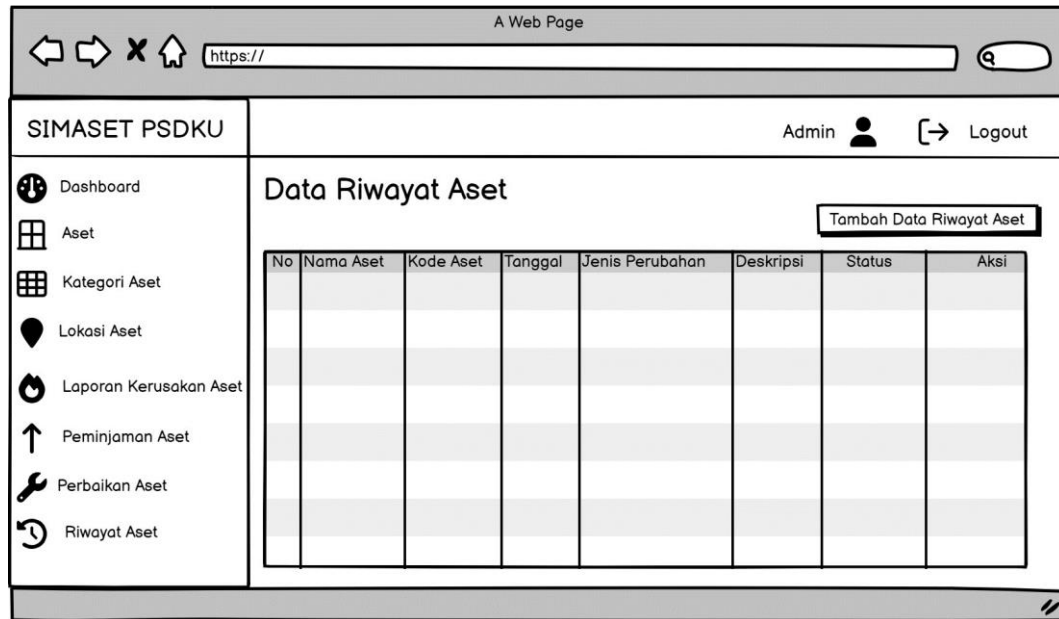
- Nama Aset
- Kode Aset
- Tanggal Permintaan Perbaikan
- Deskripsi Perbaikan
- Status
- Nama Pelapor Perbaikan
- Informasi Umum Keadaan Aset

A 'Submit' button is located at the bottom right of the form area.

Gambar III.28 *Mockup* Tambah Data Perbaikan Aset

Gambar III.28 menampilkan *mockup* halaman untuk menambah data perbaikan aset yang dapat diakses oleh admin. Pada halaman ini, terdapat formulir yang mencakup nama aset, kode aset, tanggal permintaan perbaikan, deskripsi perbaikan, status, nama pelapor perbaikan, dan informasi umum keadaan aset.

15) Mockup Data Riwayat Aset



Gambar III.29 *Mockup* Data Riwayat Aset

Gambar III.29 menampilkan *mockup* halaman riwayat aset yang dapat diakses oleh admin. Pada halaman ini, terdapat data yang mencakup nama aset, kode aset, tanggal, jenis perubahan, deskripsi dan status.

3.3.2.3 Development

Dalam tahap ini, sistem akan diimplementasikan dalam bahasa pemrograman PHP dengan menggunakan *framework* Laravel versi 10 dan akan berinteraksi dengan *database* MySQL sesuai dengan desain serta *mockup* yang telah dirancang pada tahap sebelumnya. Proses pengembangan sistem akan dilakukan menggunakan *code editor* Visual Studio Code dan akan diuji menggunakan *browser* Google Chrome.

3.3.2.4 Testing

Setelah menyelesaikan tahap *development*, langkah selanjutnya adalah melakukan *testing* (pengujian). Pengujian ini bertujuan untuk memverifikasi bahwa perangkat lunak yang telah dikembangkan dan terintegrasi berfungsi sesuai dengan spesifikasi dan persyaratan yang telah ditetapkan. Jika setelah pengujian ditemukan bahwa sistem masih belum memenuhi kebutuhan pengguna, maka akan dilakukan analisis

kebutuhan ulang. Dalam penelitian ini, metode pengujian yang digunakan adalah *Blackbox Testing* dengan metode *Equivalence Partitioning*.

1) Skenario Pengujian *Login*

Tabel 3 Skenario Pengujian Login

ID	Deskripsi	Skenario	Hasil yang Diharapkan
P-01	Admin dan staf aset melakukan login menggunakan kombinasi <i>username</i> dan <i>password</i>	<i>Username</i> dan <i>password</i> benar	Diarahkan ke halaman <i>dashboard</i>
		<i>Username</i> atau <i>password</i> salah	Menampilkan pesan kesalahan “ <i>username</i> atau <i>password</i> salah”
		<i>Username</i> atau <i>password</i> kosong	Menampilkan pesan kesalahan “ <i>mohon masukkan email/password</i> ”

2) Skenario Pengujian Menambah Aset

Tabel 4 Skenario Pengujian Menambah Aset

ID	Deskripsi	Skenario	Hasil yang Diharapkan
P-02	Admin dan staf aset ingin menambahkan data aset baru ke dalam sistem	Klik <i>button</i> tambah aset	Menampilkan halaman formulir penambahan data aset
		Input data aset valid	Sistem menyimpan data aset baru ke dalam basis data
		Input data aset tidak valid	Menampilkan pesan kesalahan

3) Skenario Pengujian Mengedit Aset

Tabel 5 Skenario Pengujian Mengedit Aset

ID	Deskripsi	Skenario	Hasil yang Diharapkan
P-03	Admin dan staf aset ingin mengedit informasi suatu aset yang sudah ada dalam sistem.	Klik <i>icon</i> edit aset	Menampilkan halaman formulir untuk mengedit aset
		Input data aset valid	Sistem menyimpan data aset baru ke dalam basis data
		Input data aset tidak valid	Menampilkan pesan kesalahan

4) Skenario Pengujian Menghapus Aset

Tabel 6 Skenario Pengujian Menghapus Aset

ID	Deskripsi	Skenario	Hasil yang Diharapkan
P-04	Admin dan staf aset ingin menghapus suatu aset dari sistem	Klik <i>icon</i> hapus aset	Menampilkan pesan konfirmasi apakah ingin menghapus aset atau tidak
		ID aset benar	Data aset berhasil dihapus dari sistem
		ID aset tidak valid	Menampilkan pesan kesalahan

5) Skenario Pengujian Menambah Kategori Aset

Tabel 7 Skenario Pengujian Menambah Kategori Aset

ID	Deskripsi	Skenario	Hasil yang Diharapkan
P-05	Admin ingin menambah kategori aset baru ke dalam sistem	Klik <i>button</i> tambah kategori aset	Menampilkan halaman formulir untuk menambah data kategori aset
		Input data kategori aset valid	Sistem menyimpan data kategori aset baru ke dalam basis data
		Input data kategori aset tidak valid	Menampilkan pesan kesalahan

6) Skenario Pengujian Mengedit Kategori Aset

Tabel 8 Skenario Pengujian Mengedit Kategori Aset

ID	Deskripsi	Skenario	Hasil yang Diharapkan
P-06	Admin ingin mengedit informasi suatu kategori aset yang sudah ada dalam sistem	Klik <i>icon</i> untuk mengedit kategori aset	Menampilkan halaman formulir untuk mengedit aset
		Input data kategori aset valid	Sistem menyimpan data kategori aset baru ke dalam basis data
		Input data kategori aset tidak valid	Menampilkan pesan kesalahan

7) Skenario Pengujian Menghapus Kategori Aset

Tabel 9 Skenario Pengujian Menghapus Kategori Aset

ID	Deskripsi	Skenario	Hasil yang Diharapkan
P-07	Admin ingin menghapus suatu kategori aset dari sistem	Klik <i>icon</i> hapus kategori aset	Menampilkan pesan konfirmasi apakah ingin menghapus kategori aset atau tidak
		ID kategori aset benar	Data kategori aset berhasil dihapus dari sistem
		ID kategori aset tidak valid	Menampilkan pesan kesalahan

8) Skenario Pengujian Menambah Lokasi Aset

Tabel 10 Skenario Pengujian Menambah Lokasi Aset

ID	Deskripsi	Skenario	Hasil yang Diharapkan
P-08	Admin ingin menambah lokasi aset baru ke dalam sistem	Klik <i>button</i> tambah lokasi aset	Menampilkan halaman formulir untuk menambah data lokasi aset
		Input data lokasi aset valid	Sistem menyimpan data lokasi aset baru ke dalam basis data
		Input data lokasi aset tidak valid	Menampilkan pesan kesalahan

9) Skenario Pengujian Mengedit Lokasi Aset

Tabel 11 Skenario Pengujian Mengedit Lokasi Aset

ID	Deskripsi	Skenario	Hasil yang Diharapkan
P-09	Admin ingin mengedit informasi lokasi aset yang sudah ada dalam sistem	Klik <i>icon</i> edit lokasi aset	Menampilkan halaman formulir untuk mengedit aset
		Input data lokasi aset valid	Sistem menyimpan data lokasi aset baru ke dalam basis data
		Input data lokasi aset tidak valid	Menampilkan pesan kesalahan

10) Skenario Pengujian Menghapus Lokasi Aset

Tabel 12 Skenario Pengujian Menghapus Lokasi Aset

ID	Deskripsi	Skenario	Hasil yang Diharapkan
P-10	Admin ingin menghapus suatu lokasi aset dari sistem	Klik <i>icon</i> hapus aset	Menampilkan pesan konfirmasi apakah ingin menghapus aset atau tidak
		ID lokasi aset valid	Data lokasi aset berhasil dihapus dari sistem
		ID lokasi aset tidak valid	Menampilkan pesan kesalahan

11) Skenario Pengujian Mengajukan Peminjaman Aset

Tabel 13 Skenario Pengujian Mengajukan Peminjaman Aset

ID	Deskripsi	Skenario	Hasil yang Diharapkan
P-11	Staf aset ingin mengajukan permohonan peminjaman aset	Klik halaman peminjaman aset	Menampilkan formulir pengajuan peminjaman aset
		Input data pengajuan peminjaman aset valid	Sistem menyimpan data pengajuan peminjaman aset baru ke dalam basis data
		Input data pengajuan peminjaman aset tidak valid	Menampilkan pesan kesalahan

13) Skenario Pengujian Melaporkan Kerusakan Aset

Tabel 14 Skenario Pengujian Melaporkan Kerusakan Aset

ID	Skenario	Skenario	Hasil yang Diharapkan
P-13	Staf aset ingin melaporkan kerusakan aset	Klik halaman laporan kerusakan aset	Menampilkan halaman formulir laporan kerusakan aset
		Input data laporan kerusakan aset valid	Sistem menyimpan data laporan kerusakan aset baru ke dalam basis data
		Input data laporan kerusakan aset tidak valid	Menampilkan pesan kesalahan

14) Skenario Pengujian Menyetujui Pengajuan Peminjaman Aset

Tabel 15 Skenario Pengujian Menyetujui Pengajuan Peminjaman Aset

ID	Deskripsi	Skenario	Hasil yang Diharapkan
P-14	Admin ingin menyetujui permohonan peminjaman aset yang telah diajukan.	Klik halaman peminjaman aset	Menampilkan halaman peminjaman aset
		Klik <i>icon</i> centang untuk menyetujui pengajuan peminjaman aset	Menampilkan pesan sukses

15) Skenario Pengujian Menolak Pengajuan Peminjaman Aset

Tabel 16 Skenario Pengujian Menolak Pengajuan Peminjaman Aset

ID	Deskripsi	Skenario	Hasil yang Diharapkan
P-15	Admin ingin menolak permohonan peminjaman aset yang telah diajukan.	Klik halaman peminjaman aset	Menampilkan halaman peminjaman aset
		Klik <i>icon</i> silang untuk menolak pengajuan peminjaman aset	Menampilkan pesan sukses

16) Skenario Pengujian Menambah Data Perbaikan Aset

Tabel 17 Skenario Pengujian Menambah Data Perbaikan Aset

ID	Deskripsi	Skenario	Hasil yang Diharapkan
P-16	Admin ingin menambah data perbaikan yang telah dilakukan pada aset.	Klik <i>button</i> tambah data	Menampilkan halaman formulir untuk menambah data perbaikan aset
		Input data perbaikan valid	Sistem menyimpan data laporan kerusakan aset baru ke dalam basis data
		Input data perbaikan tidak valid	Menampilkan pesan kesalahan

3.3.3 Instalasi ke Server

Untuk membangun virtualisasi server, terdapat beberapa langkah yang harus dilakukan, yaitu:

- 1) Mempersiapkan server VMware vSphere
- 2) Melakukan instalasi Ubuntu 23.04
- 3) Instalasi Apache
- 4) Transfer Proyek Laravel ke Server
- 5) Instalasi Composer
- 6) Instalasi PHP
- 7) Konfigurasi Environment Laravel
- 8) *Generate Key* Aplikasi Laravel
- 9) Konfigurasi Web Server
- 10) Migrasi *Database*
- 11) Menjalankan Web

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil yang telah dicapai dalam tugas akhir ini, terdapat beberapa kesimpulan sebagai berikut:

1. Penggunaan *framework* Laravel dalam pengembangan sistem informasi manajemen aset telah berhasil. Keseluruhan penggunaan fitur-fitur seperti login, tampilan data aset, peminjaman, pelaporan kerusakan, perbaikan, dan riwayat aset, telah memenuhi kebutuhan fungsional yang dibutuhkan pengguna.
2. Penggunaan metode Kanban dalam sistem informasi manajemen aset telah berhasil. Dengan menerapkan prinsip visualisasi alur kerja, manajemen tugas, dan pengelolaan waktu, memudahkan pemantauan progres setiap tugas dan meningkatkan efisiensi pengembangan sistem.

5.2 Saran

Saran untuk penelitian berikutnya yang melibatkan pengembangan atau peningkatan sistem dapat berfokus pada penerapan fungsi generasi ke PDF, Excel atau QR *code* untuk data aset. Fungsi ini dapat membantu memperkaya fitur web manajemen aset, serta memberikan kemudahan dalam pelacakan dan identifikasi aset secara cepat menggunakan teknologi pemindaian.

DAFTAR PUSTAKA

- [1] P., Wahyuni, S., & Rifki Khoirudin, Me. (n.d.). Pengantar Manajemen Aset.
- [2] Luke Welling, & Laura Thomson. (2001). *PHP and MySQL Web Development*. SAMS.
- [3] Achmad Solichin. (n.d.). Pemrograman Web dengan PHP dan MySQL.
- [4] Devi Purnama Sari, & Rony Wijanarko. (2019). Implementasi Framework Laravel pada Sistem Informasi Penyewaan Kamera (Studi Kasus Di Rumah Kamera Semarang). *Informatika Dan RPL*, 2, 32–36.
- [5] Yuniar Supardi, & Sulaeman. (2019). Semua Bisa Menjadi Programmer Laravel Basic. Elex Media Komputindo.
- [6] Tanari, A., Handojo, A., & Andjarwirawan, J. (n.d.). Aplikasi Pencarian Jurnal Ilmiah dengan Term Frequency-Inverse Document Frequency.
- [7] Canggih Ajika Pamungkas. (2017). Pengantar dan Implementasi Basis Data. DEEPUBLISH.
- [8] Rahimi Fitri, S. Kom., M. Kom. (2020). Pemrograman Basis Data Menggunakan MySQL. Deepublish.
- [9] Erawan, L. (n.d.). The Systems Development Life Cycle.
- [10] Andrew Stellman, & Jennifer Greene. (2014). Learning Agile: Understanding Scrum, XP, Lean, and Kanban. O'Reilly Media, Inc.
- [11] Sumarudin, A., & Suheryadi, A. (2021). Penerapan Sistem Informasi Penelitian Internal Di Politeknik Negeri Indramayu Menggunakan Metode Kanban (Vol. 2, Issue 4).
- [12] HAVILUDDIN. (2011). Memahami Penggunaan UML (Unified Modelling Language). *Jurnal Informatika Mulawarman*.
- [13] Khasanah, S. N., Kuryanti, S. J., Informasi, J. S., Nusa, S., & Jakarta, M. (n.d.). Rancangan Virtualisasi Server Menggunakan VMWare Vsphere (Vol. 7).
- [14] Ni Made Dwi Febriyanti, A.A. Kompiang Oka Sudana, & I Nyoman Piarsa. (2021). Implementasi Black Box Testing pada Sistem Informasi Manajemen Dosen. *JITTER*, 2.
- [15] Priyaungga, B. A., Aji, D. B., Syahroni, M., Aji, N. T. S., & Saifudin, A.

- (2020). Pengujian Black Box pada Aplikasi Perpustakaan Menggunakan Teknik Equivalence Partitions. *Jurnal Teknologi Sistem Informasi Dan Aplikasi*, 3(3), 150.
- [16] Sujatmiko, F., & Fatrianto Suyatno, D. (n.d.). Rancang Bangun Sistem Informasi Manajemen Aset Alat Kantor Berbasis Website menggunakan Framework Laravel dan Metode LIFO. *JEISBI*, 02, 2021.
- [17] Afrody, H., Prima Mustika, W., & Sanjaya, A. (2023). Sistem Informasi Manajemen Aset (SIMASET) Berbasis Web (Vol. 4, Issue 2).
- [18] Fifit Alfiah, Fuad Muhammad Nur, & Dedi Gunawan. (2022). Aplikasi Sistem Manajemen Aset Laboratorium Komputer Pada SMKN2 Kab.Tangerang Menggunakan Framework Laravel8. *Creative Education of Research in Information Technology and Artificial Informatics*, 8.
- [19] Juliana Roshinta. (2022). Pengembangan Sistem Informasi Manajemen Aset Dan Keuangan Untuk Meningkatkan Akuntabilitas Manajemen Masjid.
- [20] Danang Ardiyanto, & IGL. Putra Eka Prisma. (2022). Implementasi Rancang Bangun Manajemen Aset Berbasis Website di PT Petrokimia Gresik. *Journal of Informatics and Computer Science*, 03.
- [21] Ditha Nurcahya Avianty, Andy Hidayat Jatmika, & Lalu Miftahul Ulum. (2020). Sistem Informasi Manajemen Aset Pada Bpkad Provinsi Nusa Tenggara Barat Berbasis Website Dengan Framework Laravel. *Jurnal Begawe Teknologi Informasi*, 1.
- [22] Andry Prayoga, Indrarini Dyah Irawati, & Muhammad Zero Fernandes. (2023). Perancangan Dan Implementasi Website Manajemen Pendataan Dan Detail Penggunaan Aset Jaringan Iconnet Studi Kasus Di Pt Indonesia Comnets Plus Regional Jawa Barat. 9.
- [23] Junita, R. R. (2021). *Asset Management Information System at The Health Service of Aceh Province. International Journal Software Engineering and Computer Science (IJSECS)*, 1(2), 49–57.
- [24] Rahmat, A., & Hanifiah, N. A. M. (2020, December 14). *Usability Testing in Kanban Agile Process for Club Management System. 6th International Conference on Interactive Digital Media, ICIDM 2020*.

- [25] Kirovska, N., & Koceski, S. (n.d.). *Usage Of Kanban Methodology At Software Development Teams*.
- [26] Chaerul, R. A., Hayuhardhika, W., Putra, N., & Hanggara, B. T. (2021). *Utilizing of the Trello API Within the Development of a Monitoring Information System Recording of Project Activities Using a Website-Based Kanban System (Case Study: Electrical Project of PT. XYZ)*. In *Journal of Information Technology and Computer Science* (Vol. 6, Issue 2).
- [27] Nugraha, U., & Sianturi, T. (2021). *Blackbox Testing On E-Commerce System Web-Based Evermos (Feature: Registration Experiment & Revamp)*. In *Turkish Journal of Computer and Mathematics Education* (Vol. 12, Issue 8).
- [28] Powell, D. J. (2018). *Kanban for Lean Production in High Mix, Low Volume Environments*. 51(11), 140–143.
- [29] Yulistiyanti, D., Akhirina, T. Y., Afrizal, T., Paramita, A., & Farkhatin, N. (2022). *Testing Learning Media for English Learning Applications Using BlackBox Testing Based on Equivalence Partitions*. *Scope: Journal of English Language Teaching*, 6(2).
- [30] Kemenkeu Learning Center. Mengenal Penggolongan dan Kodefikasi Barang Milik Negara. <https://klc2.kemenkeu.go.id/kms/knowledge/klc1-pusknpk-mengenal-penggolongan-dan-kodefikasi-barang-milik-negara/detail/>.
- [31] Peraturan Menteri Keuangan Nomor:29/PMK.06/2010 tentang Penggolongan dan Kodefikasi Barang Milik Negara