

BAB III

METODOLOGI PENELITIAN

3.1 Alat dan Bahan Penelitian

3.1.1 Alat

- a. Perangkat Keras (*Hardware*) yang digunakan oleh penulis ketika membuat aplikasi adalah:
 1. *Personal Computer* (PC) atau laptop.
 - Intel® Core™ i5-4200U *processor*,
 - 8.00 GB RAM,
 - 64 bit *Operating System*,
 - *Windows 10 Pro*.
 2. *Printer* dokumen untuk mencetak laporan.
- b. Perangkat Lunak (*Software*) yang digunakan oleh penulis ketika membuat aplikasi adalah:
 1. *Microsoft Excel 2016*
 2. *Microsoft Word 2016*
 3. *Netbeans IDE 8.1*
 4. *SQL Server 2016*

3.1.2 Bahan

Peneliti memperoleh bahan penelitian data aset dari perusahaan yang masih manual dan dalam data *Microsoft Excel* untuk dikomputerisasi. Maka aplikasi disajikan untuk perancangan dan Pengembangan Aplikasi Pendataan Aset Angkasa Pura I Yogyakarta Berbasis Java.

3.2 Teknik Pengumpulan Data

3.2.1 Wawancara

Wawancara merupakan percakapan antara dua orang atau lebih antara pewawancara dan narasumber dengan tujuan mendapatkan informasi yang tepat, jelas dan akurat dari narasumber yang dipercaya. Melalui proses tanya jawab antara pewawancara dan narasumber.

Proses wawancara ini dilakukan secara langsung kepada admin kantor Angkasa Pura I, Mas Fauzan Dwi Ananda Luthfi dan Mas Yuldi Satria guna mendapatkan info yang akurat dengan berbagai pertanyaan. Terkait keamanan data, penulis bertanya seputar: Pertama, yang bisa dilakukan oleh siapa saja dan kendala apa saja yang dihadapi saat mengelola data. Kedua, apa yang dilakukan saat data yang dimasukkan tidak sesuai dengan aset yang tersedia, dan buku atau kertas pendataan hilang, terselip atau terkena cairan. Serta masukan untuk sistem yang dibangun agar membantu mempermudah penulis dalam mengerjakan tugas akhir dengan judul Pembangunan Aplikasi Pendataan Aset Aset Angkasa Pura I Yogyakarta Berbasis Java.

3.2.2 Alur Penelitian

Dalam perancangan dan pembangunan aplikasi pendataan aset ini, peneliti menggunakan metode *Software Development Life Cycle* (SDLC). Metode ini dipakai untuk proses pengembangan sistem dalam cara yang terstruktur dan metodis yang mengharuskan tahap life cycle dari awal ide sampai sistem final dan dilakukan secara beraturan.

SDLC yang digunakan adalah model *software prototyping*, yaitu salah satu metode siklus hidup (*life cycle*) yang berdasarkan konsep *working model*. Dengan tujuan membangun model sampai menjadi sistem final. Sistem akan dibangun lebih cepat dari pada metode tradisional dan dengan biaya terjangkau.

Salah satu ciri dari metodologi ini adalah *system developer*, klien dan pengguna dapat bereksperimen dan melihat cara kerja sistem sejak awal. Dalam pembuatan aplikasi ini, tahapan yang dilalui oleh penulis yaitu:

1. Pengumpulan Kebutuhan

Proses pertama yang dilakukan adalah mengumpulkan apa saja yang dibutuhkan oleh pengguna. Dalam penulisan ini, penulis mengidentifikasi kebutuhan pengguna untuk sistem yang akan dibangun serta gambar sistem yang akan dibangun.

2. Membangun *Prototyping*

Dalam tahap *prototyping*, penulis menggambarkan cara kerja sistem yang sesuai dengan kebutuhan pengguna agar dapat memberikan gambaran tentang proses atau tampilan aplikasi yang dibangun, sekaligus pengguna paham tentang fitur apa saja yang terdapat dalam aplikasi tersebut. Serta jelas *input* dan *outputnya*.

3. Evaluasi *Prototyping*

Pengguna menyetujui evaluasi sistem yang telah dirancang untuk kemudian dilanjutkan pada tahap pembuatan sistem. Namun pada kasus ini, jika pengguna tidak setuju dengan hasil *prototyping*, maka *developer* dapat mengulang kembali proses 1 dan 2.

4. Mengkodekan Sistem

Selanjutnya, penulis mengkodekan sistem sesuai dengan gambaran *prototyping* yang sudah disetujui oleh pengguna. Semua fitur yang menjadi kebutuhan pengguna untuk sebuah sistem telah direalisasikan.

5. Menguji Sistem

Pada tahap ini, aplikasi berhasil dibangun dengan cara kerja sistem yang sesuai dengan kebutuhan pengguna dan sudah tidak ditemukan *error* atau *bug*.

6. Evaluasi Sistem

Tahap evaluasi sistem dilakukan oleh pengguna untuk memastikan bahwa sistem sudah sesuai atau belum. Pada tahap ini, pengguna masih menginginkan ada tambahan fitur pada aplikasi agar lebih sempurna. Maka penulis mengulang melakukan tahapan 4 dan 5 untuk memastikan bahwa aplikasi sudah sesuai keinginan pengguna.

7. Menggunakan Sistem

Sistem siap digunakan dan semua fitur sudah sesuai dengan kebutuhan pengguna dalam memakai aplikasi.

3.3 Analisa Penelitian

3.3.1 Analisa Kebutuhan

Dari hasil wawancara, terdapat masalah yang dihadapi oleh admin. Dalam pendataan aset manual, admin mendapat kendala saat mengelola data, seperti data bisa disalahgunakan oleh pihak yang tidak bertanggung jawab tanpa seijin admin, kesalahan penulisan data aset. Selanjutnya, admin juga menemukan kendala saat melakukan pembukuan, misalnya kertasnya hilang, terselip atau terkena cairan.

Menindaklanjuti masalah yang telah diuraikan, penulis menawarkan solusi atas masalah yang dihadapi. Pertama, dengan membuat fitur *login* agar tidak ada yang dapat mengelola aplikasi selain admin. Kedua, dalam mendata aset, tentu terjadi *typo* atau kesalahan data, maka solusinya adalah dengan mengadakan fitur mengelola data. Dalam fitur mengelola data, terdapat *button* tambah, *button* edit, dan *button* hapus.

Selanjutnya, dalam proses mengelola data, admin juga membutuhkan fitur *search*. Sesuai namanya, fitur *search* berfungsi untuk mencari data-data yang ada pada tabel. Karena, ketika *searching* manual harus mencari data perbaris, maka akan lebih baik jika *searching* dilakukan dengan fitur *search*. Serta adanya fitur *print* demi kepentingan arsip data perusahaan. Sebelumnya, arsip data dilakukan dengan cara menulis ulang data yang sama agar di arsipkan untuk perusahaan dan sisanya sebagai data duplikat admin.

Dalam mendata aset-aset yang ada di PT. Angkasa Pura I (Persero), hal-hal yang harus diperhatikan adalah:

1. Adanya nomor pendataan sebagai nomor urut.
2. Adanya nama peralatan, yaitu nama aset.
3. Adanya id jenis sebagai nama bahwa aset adalah jenis dari suatu perangkat.
4. Adanya serial number yang terdapat pada fisik aset.
5. Adanya data pabrik pembuat. Artinya, aset berasal dari pabrik tersebut.
6. Adanya tahun yang menunjukkan tahun pembelian barang.
7. Adanya daya sebagai pengukur ketahanan aset.
8. Adanya data kondisi yang menunjukkan bahwa barang dalam kondisi terpakai atau tidak.
9. Adanya data penempatan. Artinya tempat dimana aset diletakkan.
10. Adanya data merk dan tipe aset.
11. Adanya status barang yang terpakai atau tidak.
12. Adanya keterangan jika barang terpakai atau tidak.

Peneliti mengumpulkan data terkait kebutuhan pembangunan aplikasi yang secara langsung diterima dari narasumber berdasarkan permasalahan yang ada. Data digunakan untuk menganalisa kebutuhan sistem dan fungsi pada *software* yang akan dibangun. Dalam tahapan ini penulis juga menentukan target pengguna aplikasi. Daftar kebutuhan aplikasi yaitu adanya fitur *login*. Setelah melewati fitur *login*, juga ada fitur *search* dan *print* yang dilengkapi oleh 3 *button*. Berikut analisa kebutuhan aplikasi:

1. Dalam fitur *login*, keaslian data harus dijaga. Maka ditawarkan adanya fitur *login* admin yang berfungsi mengelola data.

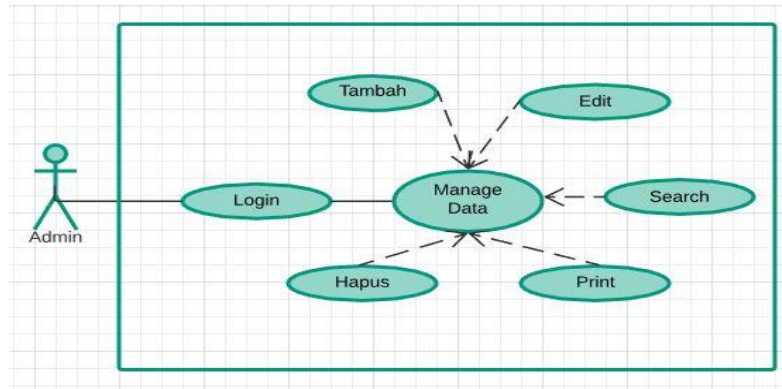
2. Dalam menghadapi kendala-kendala yang diceritakan oleh narasumber pada sesi wawancara, maka yang dibutuhkan selanjutnya adalah penambahan tombol-tombol utama yang nantinya akan sangat membantu meringankan cara kerja admin. Seperti:
 - a. Membuat *button* tambah data, dimana *button* tambah data adalah tombol utama yang dibutuhkan untuk memasukkan data kedalam sistem.
 - b. Kemudian untuk mengatasi kesalahan data yang dialami oleh admin, maka perlu ditambahkan *button* hapus pada sistem.
 - c. Dalam pendataan data aset ditemukan kekeliruan saat melakukan pendataan, maka penulis juga menambahkan *button* edit dalam sistem demi kelancaran pendataan.
3. Fitur *search*. Sebelumnya, admin mendapat kendala kesulitan mencari data atau proses pencarian tidak efisien karena dari sekian banyak data, admin harus mencari satu persatu data dengan cara menyortir data perbaris.
4. Fitur *print*. Dengan adanya fitur *print*, maka akan membantu kepentingan arsip data perusahaan. Fitur *print* membuat kerja admin menjadi lebih singkat, yaitu hanya dengan menekan tombol *print*, data akan tercetak.

3.4 Rancangan Sistem dan Basis Data

Sebelum membangun aplikasi tugas akhir dengan judul Pengembangan Aplikasi Pendataan Aset Angkasa Pura I Yogyakarta Berbasis *Java* maka perancangannya dituangkan terlebih dahulu kedalam diagram, agar jalannya aplikasi dan *database* ini jelas sesuai kebutuhan. Metode yang digunakan dalam pembangunan aplikasi *desktop* ini adalah *Unified Model Language* (UML). Model yang dipakai dalam aplikasi ini adalah *Entity Relationship Diagram* (ERD).

3.4.1 Usecase Diagram

Gambar 3.1 menjelaskan tentang *Usecase Diagram* yang digambarkan pada rancangan aplikasi data aset Angkasa Pura I Yogyakarta.

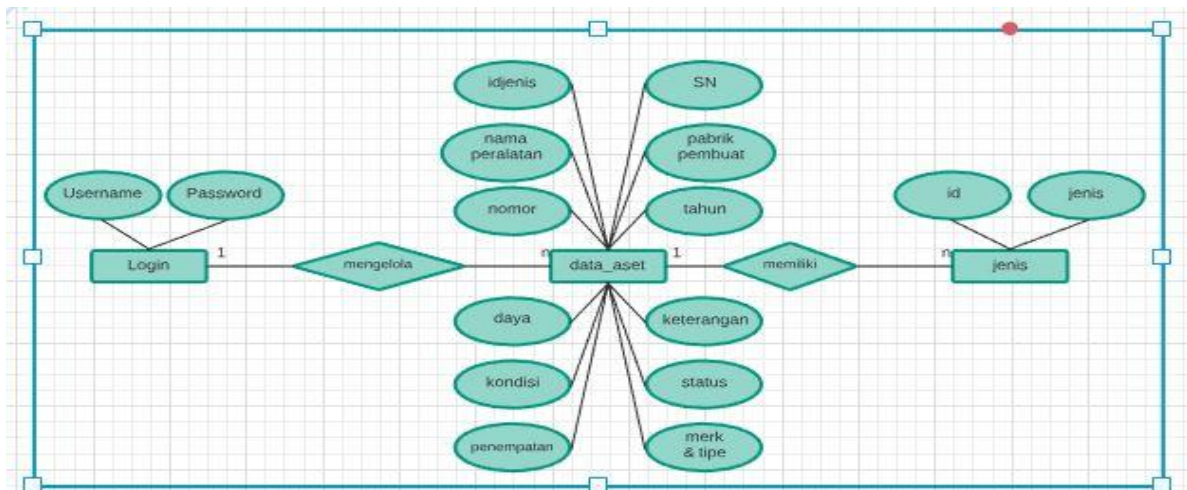


Gambar 3. 1 Usecase Diagram

Pada gambar 3.1 menjelaskan tentang *usecase* diagram, dimana sistem hanya dikelola sepenuhnya oleh admin. Admin bertanggung jawab atas keakuratan data, maka dalam penggunaan sistem, admin diharuskan melakukan proses *login* terlebih dahulu untuk dapat mengelola data. Dalam mengelola data, admin bisa menambah data dengan cara menekan tombol tambah pada sistem. Selanjutnya, jika data keliru dimasukkan maka admin bisa mengedit data serta menghapus data dengan cara menekan tombol hapus yang tersedia pada sistem. Kemudian, daftar data ini harus di arsip, maka tombol *print* berfungsi untuk mencetak data yang akan diarsip. Pada sistem juga tersedia tombol *search* dimana data bisa dicari dengan memasukkan nama jenis pada *textbox search*.

3.4.2 ER Diagram

Gambar 3.2 menjelaskan tentang *Entity Relationship Diagram* (ERD) yang digambarkan pada rancangan aplikasi data aset Angkasa Pura I Yogyakarta.

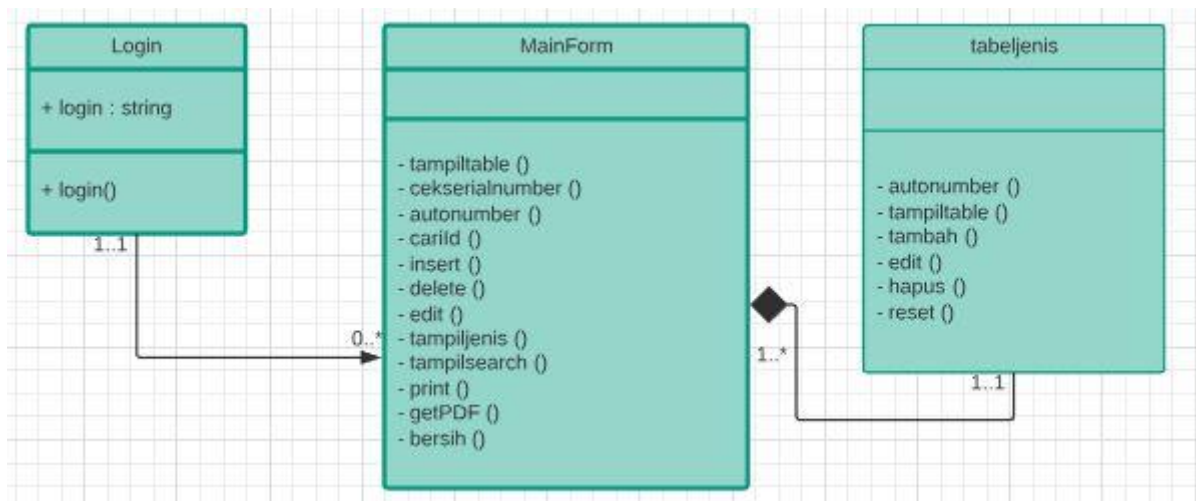


Gambar 3. 2 ER Diagram

Pada gambar 3.2 terdapat tiga (3) entitas yang tertera dalam *Entity Relationship Diagram*, yaitu:

1. *Login*, entitas *login* pada *Entity Relationship Diagram* ini memiliki 2 atribut, yaitu *username* dan *password* sebagai syarat untuk dapat masuk ke sistem dan mengelola data, sehingga fungsi admin sangat berpengaruh pada data sistem. Oleh karena itu, *username* dan *password* hanya menjadi rahasia admin.
2. Entitas *data_aset* dalam cara kerja *Entity Relationship Diagram* ini memiliki 12 atribut yaitu nomor peralatan, nama peralatan dan jenis peralatan, serial number, status, pabrik pembuat, merk dan tipe, tahun, daya, kondisi penempatan dan keterangan. Admin mengelola peralatan dengan cara memasukkan nomor peralatan sebagai id pada barang, dilanjutkan dengan nama dan jenis peralatan.
3. Entitas *jenis* pada perancangan ini dimaksudkan untuk menambah data jenis pada tabel sistem.

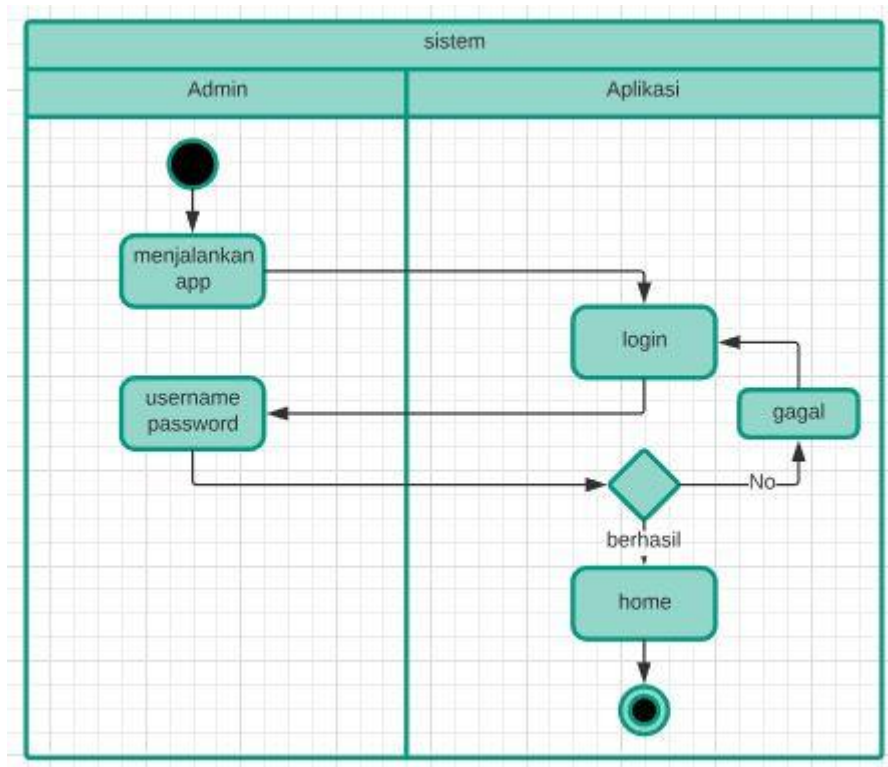
3.4.3 Class Diagram



Gambar 3. 3 Class Diagram

Seperti pada gambar 3.3 *class* diagram data aset, *class* diagram memberikan gambaran luas dari suatu sistem dengan menggambarkan hubungan apa yang terjadi, bukan apa yang terjadi jika mereka berhubungan.

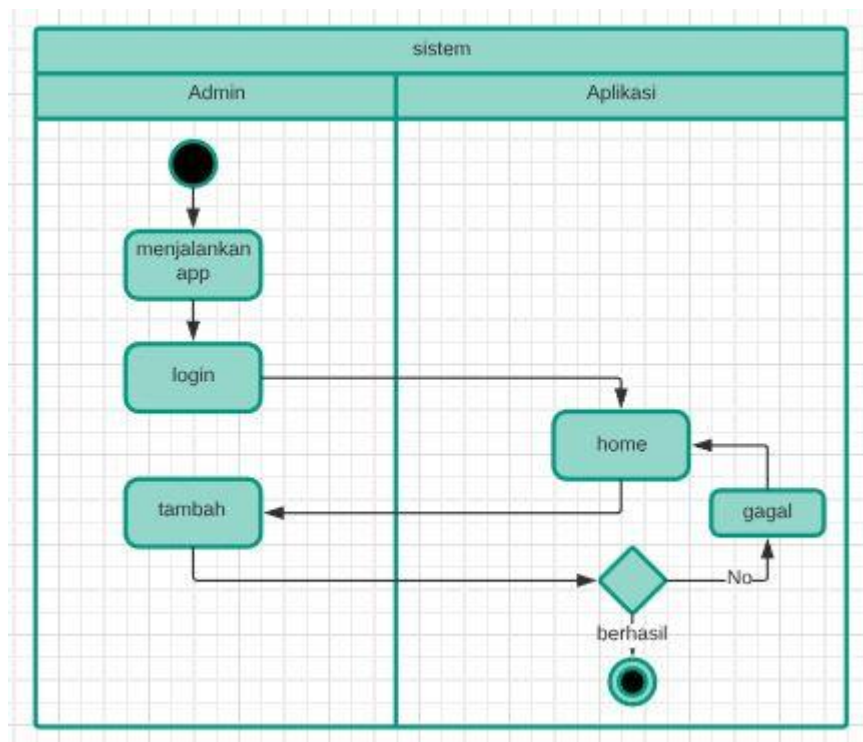
3.4.4 Activity Diagram Login



Gambar 3. 4 Activity Diagram Login Admin

Gambar 3.4 menceritakan tentang *activity* diagram *login* admin. Dimana ketika admin akan mendata, maka hal pertama yang dilakukan setelah menjalankan aplikasi adalah *login*, dengan proses memasukkan *username* dan *password* admin yang sesuai. Namun jika admin salah memasukkan salah satu *textbox* antara *username* dan *password*, maka admin harus kembali memasukkan *username* dan *password* yang benar.

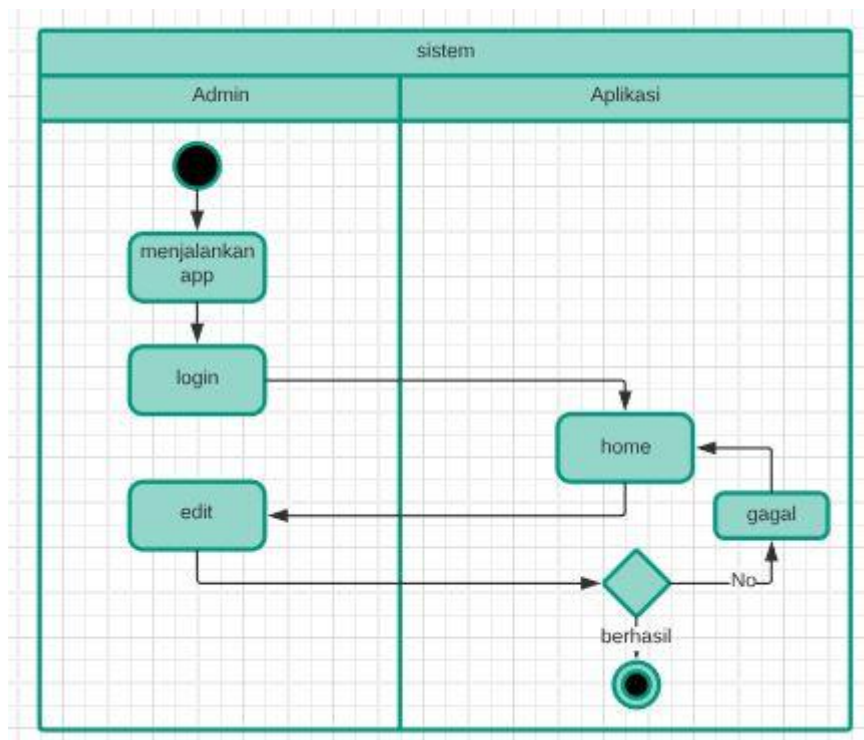
3.4.5 Activity Diagram Input Data Aset



Gambar 3.5 Activity Diagram Input Data

Gambar 3.5 menceritakan tentang *activity* diagram data aset, dimana ketika admin sudah *login*, maka akan keluar tampilan tabel untuk memasukkan data satu persatu kedalam tabel sistem. Data harus dimasukkan sesuai dengan nama *textbox* yang tersedia, setelah semua data dimasukkan kedalam *textbox*, selanjutnya admin dapat menekan tombol tambah agar data masuk ke tabel yang sudah tersedia.

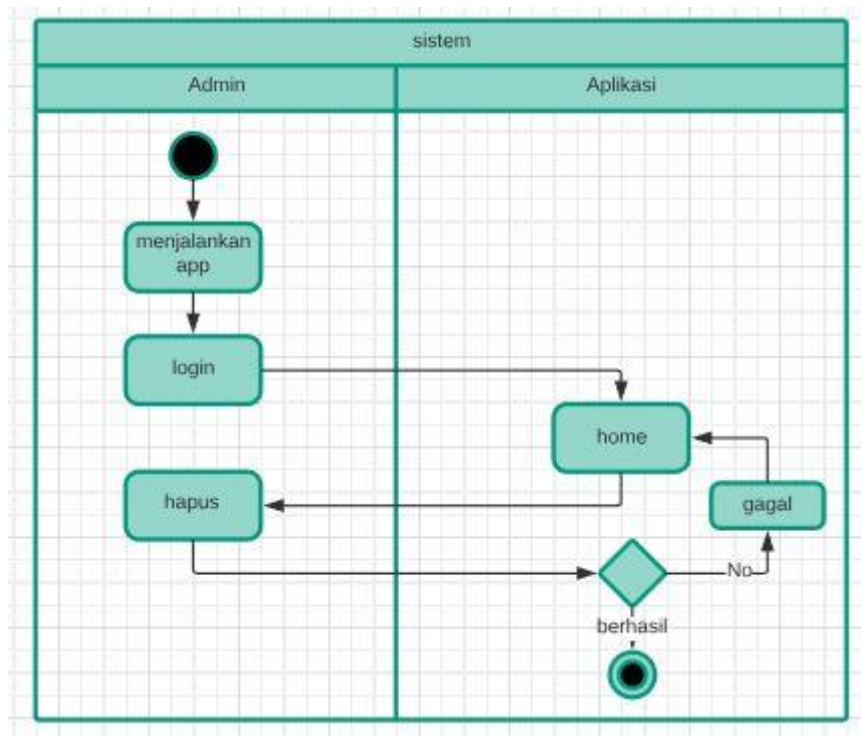
3.4.6 Activity Diagram Edit Data Aset



Gambar 3. 6 Activity Diagram Edit

Gambar 3.6 menceritakan tentang *activity* diagram ketika admin sudah *login*, maka akan keluar tampilan tabel untuk mengedit data yang diketahui terdapat kekeliruan pada saat pendataan sebelumnya. Cara mengedit tabel adalah dengan melakukan klik pada data yang akan di edit, kemudian semua data akan masuk kedalam *textbox*, maka admin bisa merubah data pada *textbox* yang tersedia tentang data apa saja yang keliru, setelah data selesai diubah, selanjutnya tekan tombol edit pada layar, maka dengan otomatis tampilan tabel sudah berubah sesuai yang terakhir.

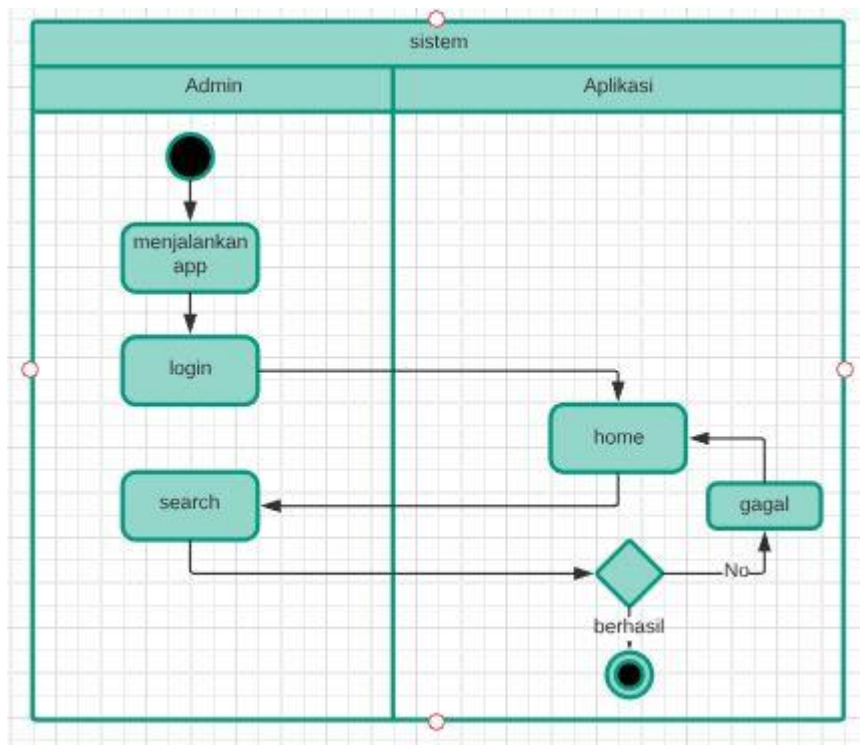
3.4.7 Class Diagram Hapus Data Aset



Gambar 3. 7 Activity Diagram Hapus

Gambar 3.7 menceritakan tentang *activity* diagram ketika admin sudah *login*, maka akan keluar tampilan tabel untuk menghapus data yang salah atau keliru pada saat pendataan sebelumnya, dengan cara yang hampir sama dengan tombol edit. Dimulai dengan menekan data yang akan dihapus, lalu klik tombol hapus yang ada pada sistem. Data berhasil dihapus.

3.4.8 Class Diagram *Search Data Aset*



Gambar 3. 8 Activity Diagram search

Selanjutnya, pada gambar 3.8 menceritakan tentang bagaimana sistem melakukan pencarian data. setelah admin berhasil *login*, maka tampilan selanjutnya adalah tampilan daftar tabel yang memiliki tombol *search*. Cara kerja tombol *search* adalah dengan memasukkan data yang akan dicari pada *textbox*, kemudian klik tombol cari, maka barang yang dicari akan ditampilkan pada tabel sistem.

3.4.9 Kamus *Database*

Tabel 3.1, tabel 3.2 dan tabel 3.3 adalah tabel *database* pada aplikasi Pengembangan Aplikasi Pendataan Aset Angkasa Pura I Yogyakarta Berbasis Java.

Tabel 3. 1 Kamus Basis Data Admin

Nama Kolom	Tipe Data	Key	Keterangan
<i>Username</i>	Varchar (50)	<i>Primary Key</i>	<i>Username</i> admin
<i>Password</i>	Varchar (50)	-	<i>Password</i> admin

Tabel 3. 2 Basis Data Aset

Nama Kolom	Tipe Data	Key	Keterangan
Nomor	Varchar (50)	<i>Primary key</i>	<i>Auto Increment</i>
Nama peralatan	Varchar (50)	-	
IdJenis	Integer	-	
Serial Number	Integer	-	
Pabrik Pembuat	Varchar (50)	-	
Merk dan tipe	Varchar (50)	-	
Tahun	Varchar (50)	-	
Daya	Varchar (50)	-	
Kondisi	Varchar (50)	-	
Penempatan	Varchar (50)	-	
Status	Varchar (50)	-	
Keterangan	Varchar (50)	-	

Tabel 3. 3 Kamus Basis Data Jenis

Nama Kolom	Tipe Data	Key	Keterangan
Id	Integer	<i>Primary Key</i>	<i>Auto Increment</i>
Jenis	Varchar (50)	-	

3.5 Rancangan Antar Muka

Rancangan antar muka dimaksudkan sebagai penghubung antara pengguna dan sistem aplikasi yang dibangun. Rancangan antar muka juga merupakan salah satu sarana untuk menyampaikan bahwa cara kerja sistem adalah seperti yang tergambar, sehingga pengguna akan lebih memahami apa yang ada pada sistem yang dibangun.

3.5.1 Rancangan Antar Muka *Login*

Rancangan antar muka *login* adalah tampilan awal pada sistem data aset saat admin mulai mengakses aplikasi Data Aset Angkasa Pura 1 Yogyakarta. Tampilan *login* admin berfungsi untuk menjaga keaslian data. Tampilan *login* dapat dilihat pada gambar 3.9.



Gambar 3. 9 Rancangan Login

3.5.2 Rancangan Antar Muka Daftar *Textbox*

Rancangan daftar tabel adalah halaman tampilan. Dimana admin memasukkan satu persatu aset data yang ada di Angkasa Pura I Yogyakarta. Dengan nomor urut yang *auto increment* untuk menghindari data yang *double* seperti pada gambar 3.10.

NOMOR	<input type="text" value="4"/>
NAMA PERALATAN	<input type="text"/>
JENIS	<input type="text" value="FIREWALL"/>
MERK DAN TIPE	<input type="text"/>
SERIAL NUMBER	<input type="text"/>
PABRIK PEMBUAT	<input type="text"/>
TAHUN	<input type="text"/>
DAYA	<input type="text"/>
KONDISI	<input type="text"/>
PENEMPATAN	<input type="text"/>
STATUS	<input type="text"/>
KETERANGAN	<input type="text"/>

Gambar 3. 10 rancangan antar muka textbox

3.5.3 Rancangan Antar Muka Tabel

Pada rancangan antar muka terdapat tabel yang berfungsi sebagai tempat penyimpanan data yang sudah dimasukkan oleh admin pada daftar tabel sebelumnya. Maka data-data yang masuk akan disesuaikan didalam tabel yang terlihat pada gambar 3.11. Tabel ini hanya berisi data yang sudah dimasukkan oleh admin dengan benar.

nomor	nama_pe...	jenis	serial_nu...	pabrik_pe...	merk_type	tahun	daya	kondisi	penempa...	status	keterangan
1	Fortigate	Firewall	89632547	North Am...	300D	2017	106/220V	Baik	Server Gr...	Digunakan	
2	MikroTik	Router	2923480	Latvia	CCR1036	2016	110/220V	Baik	Server Gr...	Tidak dig...	Diganti fo...
3	Mikrotik	Router	23498654	Latvia	RB1100AH	2013	110/220V	Kurang B...	Server Gr...	Tidak dig...	Diganti s...
4	Mikrotik	Router	98365	Latvia	RB1100AH	2013	110/220V	Baik	Ruang Inf...	Tidak dig...	Diganti s...
5	Mikrobit	Router	8396548	Latvia	Aneto Lev...	2013	110/220V	Kurang B...	Server Gr...	Tidak dig...	Diganti s...
6	Mikrotik	Router	3865487	Latvia	RB2011	2014	110/220V	Baik	Ruang A...	Digunakan	
7	Cisco	Switch	235468	3650G	North Am...	2017	110/220V	Baik	Server Gr...	Digunakan	
8	Cisco	Switch	456789	North Am...	3560	2015	110/220V	Baik	Server Gr...	Digunakan	
9	HP	Switch	28396478	Pro Curve	America	2011	110/220V	Baik	Server Gr...	Digunakan	

Gambar 3. 11 rancangan antar muka tabel

3.5.4 Rancangan Antar Muka *Button*

Rancangan antar muka *button* meliputi *button* tambah, *button* edit, *button* hapus serta *button* print. Masing-masing dari *button* tersebut memiliki fungsi yang berbeda-beda, admin dapat menggunakan *button* tersebut sesuai fungsi seperti pada gambar 3.12.



Gambar 3. 12 Rancangan Antar Muka Button

3.5.5 Rancangan Antar Muka Jenis

Pada rancangan antar muka antar jenis terdapat tabel yang berfungsi menambah data jenis dimana data tersebut dipakai didalam data tabel aset untuk menentukan aset dari jenis apa saja. Seperti yang terlihat pada gambar 3.13.

ID Jenis	5	ID	JENIS
Jenis	<input type="text"/>	1	FIREWALL
		2	SWITCH
		3	SERVER
		4	ACCESS POINT

Tambah Hapus

Edit Reset

Kembali

Gambar 3. 13 Rancangan Antar Muka Jenis