

KATA PENGANTAR

Pengelolaan inventarisasi berpengaruh terhadap semua fungsi sistem operasi. Selayaknya proses manajerial, inventarisasi sangat diperlukan dalam serangkaian proses perencanaan hingga penghapusan suatu barang inventaris. Akan tetapi, realita yang terjadi di lapangan sebagian besar instansi kurang memperhatikan pentingnya inventarisasi.

Dalam hal ini "*Basis Data Manajemen Aset*" yang bisa digunakan untuk solusi dalam pengelolaan inventarisasi salah satunya di Jurusan Teknik Informatika Unesa. Tujuan diciptakannya Basis Data ini bisa mempermudah pencarian barang, pengecekan barang, pencatatan ruang yang dipakai, pencatatan barang yang dipinjam dan mempercepat peminjaman barang. Hal ini bisa memperhatikan dan menerapkan pertimbangan seminimum mungkin biaya yang dikeluarkan; baik berkaitan dengan finansial, material, waktu.

Surabaya, 16 Desember 2019

Penulis

DAFTAR ISI

DAFTAR ISI.....	2
DAFTAR GAMBAR.....	3
DAFTAR TABEL.....	5
BAB I PENDAHULUAN.....	9
A. Profil Instansi.....	11
B. Tujuan Aplikasi Basis Data Manaje men Aset.....	11
C. Manfaat Aplikasi Basis Data Manajemen Aset.....	11
D. Ruang Lingkup Basis Data Manajemen Aset.....	12
BAB II PERANCANGAN BASIS DATA.....	13
A. <i>Conceptual Data Model</i> (CDM).....	13
B. <i>Physical Data Model</i> (PDM).....	16
C. Pengujian Normalisasi Tabel.....	22
D. Algoritma Query.....	27
BAB III IMPLEMENTASI SISTEM BASIS DATA.....	36
A. <i>Graphical User Inteface</i> (GUI).....	36
B. Report.....	69
BAB IV HASIL UJI COBAS SISTEM BASIS DATA.....	74
A. SEKENARIO 1.....	74

DAFTAR GAMBAR

Gambar 1 . CDM Basis Data Manajemen Aset TU Jurusan Teknik Informatika.....	15
Gambar 2 . Relationship Ms. Accsess pada Basis Data Manajemen Aset TU Jurusan Teknik Informatika	19
Gambar 3 . Layout PDM Basis Data Manajemen Aset TU Jurusan Teknik Informatika	20
Gambar 4 . GUI Form Login Database.....	36
Gambar 5 . GUI Form Home Database.....	38
Gambar 6 . GUI Form Transaksi Database.....	41
Gambar 7 . GUI Form Peminjam (Pendaftaran Peminjam).....	47
Gambar 8 . GUI Detail Transaksi Database.....	49
Gambar 9 . GUI Halaman Admin Database.....	51
Gambar 10 . GUI Form Barang (Tambah Barang) Database.....	56
Gambar 11 . GUI Jenis Barang (tambah jenis barang) Database.....	62
Gambar 12 . GUI Form Ruangan (Menambah Ruangan) DatabaseBu.....	66
Gambar 13 . Report 1: Berapa Kali Barang X dipinjamkan ?.....	70
Gambar 14 . Report 2 : Status Ruangan.....	71
Gambar 15 . Report 3 : Transaksi Berdasarkan Tanggal.....	72
Gambar 16 . Report 3 : Cetak Nota.....	73
Gambar 17 . Sekenario 1 : Form Login.....	74
Gambar 18 . Sekenario 1 : Halaman Utama.....	75
Gambar 19 . Sekenario 1 : Halaman Utama 2.....	76
Gambar 20 . Sekenario 1 : Tampilan Transaksi Peminjaman.....	76
GambA R 21.Sekenario1:Tampilan sebelum dan setelah InputdataForm Pendaftaran.....	77
Gambar 22.Sekenario 1 : Input Rincian peminjaman Form Transaksi Peminjaman	79
Gambar 23 Sekenario 1 : Tampilan detail peminjaman (barang yang dipinjam)	

Pertama setelah tekan Button Meminjam.....	80
Gambar 24 . Sekenario 1 : Tampilan record Detail Peminjaman Pertama setelah InputData.....	80
Gambar 25 . Tampilan detail peminjaman (barang yang dipinjam) Ke -2 setelah tekan Button Meminjam.....	81
Gambar 26 . Tampilan record Detail Peminjaman Kedua setelah InputData.....	82
Gambar 27 . Tampilan Button pada Form Transaksi Prminjaman.....	82
Gambar 28 . Tampilan Cetak Nota.....	83

DAFTAR TABEL

Tabel 1. Barang

Kode_Baran	Id_Jenis	Nama_Barang	Tahun	Click to Add
1900	6	ENC 211	2016	
1901	6	View Sonic 233	2018	
1902	4	Broco 321'	2018	
1903	9	L-Ion 78	2016	
1904	8	Senin-Matem	2019	
1905	4	Broco X78	2017	
1906	6	View Sonic Bla	2019	
1907	4	Broco 11	2017	
1908	8	Selasa-Ipa		
*	0	0		

Tabel 2. Jenis Barang

Id_Jenis	Nama_Jenis	Click to Add
4	KABEL_OLOR	
6	LCD	
8	JURNAL	
9	SPEAKER	
12	NULL	
	(New)	

Tabel 3. Meminjam

No_Meminj	No_Transak	Kode_Baran	Jumlah	Click to Add
4	4	1904	1	
5	15	1906	1	
6	7	1906	1	
9	15	1905	0	
10	15	1901	11111	
11	3	1902	1	
13	16	1900	12	
14	16	1901	13	
*	(New)	0	0	

Tabel 4. Peminjam

NIM	Nama_Mahasiswa	KELAS	PRODI	Jenis_Kelamin	No_Hp	Click to Add
123424321	Hahahaha	2019B	MI	L	123	
161204023	Fahri M	2016	TI	P	0876-8902-722	
170974028	Novita	2017A	PTI	P	0870-7655-897	
171214017	Kevin H.	2017A	SI	L	0821-3252-979	
180121212	qada	2017	SI	L	12999999	
181204001	Suci Wahyunir	2018A	TI	P	0832-7411-876	
181214016	Salman Alfariz	2017	PTI	L	0873-7411-905	
190397011	Kemala Aprilla	2019B	MI	P	0879-7411-909	
191214013	Yanuar Putra	2019A	TI	L	0821-3211-767	

Tabel 5. Petugas

	Id_Petugas	Nama_Petu	Jenis_Kelar	No_Hp	Click to Add
[+]	11901	ILMA	P	082230051291	
[+]	11902	PIPIT	P	082239386182	
[+]	11903	AUFA	L	089728272291	
[+]	11904	INTAN	P	081283817839	
*	0				

Tabel 6. Ruangan

	Id_Ruangan	Nama_Ruan	STATUS	Click to Add
[+]	A10.01.01	KELAS	DIPAKAI	
[+]	A10.01.02	KELAS	TIDAK DIPAKAI	
[+]	A10.01.03	KELAS	DIPAKAI	
[+]	A10.01.04	KELAS	TIDAK DIPAKAI	
[+]	A10.01.05	KELAS	DIPAKAI	
[+]	A10.01.06	KELAS	DIPAKAI	
[+]	A10.01.07	KELAS	TIDAK DIPAKAI	
[+]	A10.01.08	KELAS	DIPAKAI	
[+]	A10.01.09	KELAS	TIDAK DIPAKAI	
[+]	A10.01.10	KELAS	TIDAK DIPAKAI	
[+]	A10.03.01	LABORATORIU	TIDAK DIPAKAI	
[+]	A10.03.02	LABORATORIU	DIPAKAI	
[+]	A10.03.03	LABORATORIU	TIDAK DIPAKAI	
[+]	A10.03.04	LABORATORIU	TIDAK DIPAKAI	
[+]	A10.04.01	AUDITORIUM	TIDAK DIPAKAI	
*				

Tabel 7. Transaksi Peminjaman

	No_Transak	NIM	TANGGAL	JAM_PINJAM	JAM_KEMBALI	ID_RUANGA	ID_PETUGAS	Click to Add
[+]	3	123424321	08/11/2019	12.00.00	15.00.00	A10.01.07	11903	
[+]	4	171214017	08/12/2019	10.00.00	11.00.00	A10.01.05	11904	
[+]	5	190397011	08/12/2019	11.00.00	12.00.00	A10.01.04	11904	
[+]	6	190397011	13/12/2019	11.00.00	13.00.00	A10.01.04	11904	
[+]	7	181204001	06/12/2019	05.00.00	06.00.00	A10.01.01	11902	
[+]	15	171214017	10/12/2019	11.11.00	12.12.00	A10.01.06	11903	
[+]	16	191214013	16/12/2019	11.11.00	11.23.00	A10.03.02	11902	
[+]	17	171214017	06/12/2019				11903	
[+]	21	180121212	20/12/2019			A10.01.04	11902	
*	(New)	0						

Pengelolaan inventarisasi berpengaruh terhadap semua fungsi sistem operasi. Tujuan pengelolaan inventarisasi adalah menyediakan persediaan peralatan kantor dengan jumlah dan waktu yang sesuai dengan kebutuhan. Melihat realita yang ada di lapangan, maka diperlukan suatu sistem pencatatan peralatan kantor yang mencatat seluruh persediaan barang. Pencatatan tersebut diawali dari perencanaan pengadaan barang, pencatatan jumlah dan spesifikasi barang yang masuk instansi, perawatan dan pengelolaan barang tersebut ketika sudah didistribusikan di masing-masing unit, serta penanganan yang dilakukan ketika barang tersebut sudah tidak layak pakai.

Inventarisasi idealnya dilakukan secara terperinci pada interval sejumlah tahun tertentu. Pertimbangan tersebut melatarbelakangi penulis untuk melakukan membuat “Basis Data Manajemen Aset” yang mengenai perencanaan atas pengelolaan serangkaian kegiatan inventarisasi melalui pemanfaatan sistem teknologi informasi dan komunikasi pada Jurusan Teknik Informatika Unesa. Jurusan Teknik Informatika Unesa merupakan salah satu Jurusan di suatu instansi pendidikan perguruan tinggi yang dalam melaksanakan tugas dan kewajibannya memerlukan barang inventaris kantor tata usaha untuk kelancaran tugas dan pencapaian tujuan.

Keberadaan barang-barang inventaris kantor tata usaha memerlukan proses pengelolaan, pencatatan, atau pelaporan barang untuk mengetahui jumlah barang, penambahan barang yang ada maupun untuk mengetahui laporan mutasi atau penghapusan barang. Barang inventaris harus dikelola dengan baik agar dapat dimanfaatkan secara

maksimal. Rincian-rincian data yang bersangkutan dibawah ini :

A. Profil Instansi

Nama Instansi	:Jurusan Teknik Informatika Universitas Negeri Surabaya
Tempat Instansi	:Gedung A10 Jurusan Teknik Informatika Universitas Surabaya
Tahun Berdiri	:2015
Bidang Instansi	:Pendidikan
Kebutuhan Basis Data	:Manajemen aset Tata Usaha meliputi barang dan ruangan berbasis aplikasi

B. Tujuan Aplikasi Basis Data Manajemen Aset

1. Mempermudah pencarian barang
2. Pengecekan barang
3. Pencatatan ruang yang dipakai
4. Pencatatan barang yang dipinjam
5. Pengefensiensian waktu dalam peminjaman barang

C. Manfaat Aplikasi Basis Data Manajemen Aset

1. Dapat menghemat waktu dalam meminjam barang
2. Menghemat kertas untuk mencatat asesmen yang akan dipinjam
3. Meminimalkan rekап peminjaman yang hilang

BAB II

PERANCANGAN BASIS DATA

A. *Conceptual Data Model (CDM)*

Pada rancangan CDM Transaksi Peminjaman merupakan pemegang entitas paling penting dalam sistem basis data, karena setiap peminjaman yang dilakukan akan ter *record* pada tabel entitas tersebut. Selain itu, Transaksi Peminjaman memiliki riwayat nama-naman peminjam berdasarkan gambar CDM yang ada, **gambar.1**, yang bertujuan untuk memudahkan *User* (petugas) dalam mencatat setiap transaksi yang dilakukan. Entitas beserta atribut yang dimiliki pada CDM antara lain:

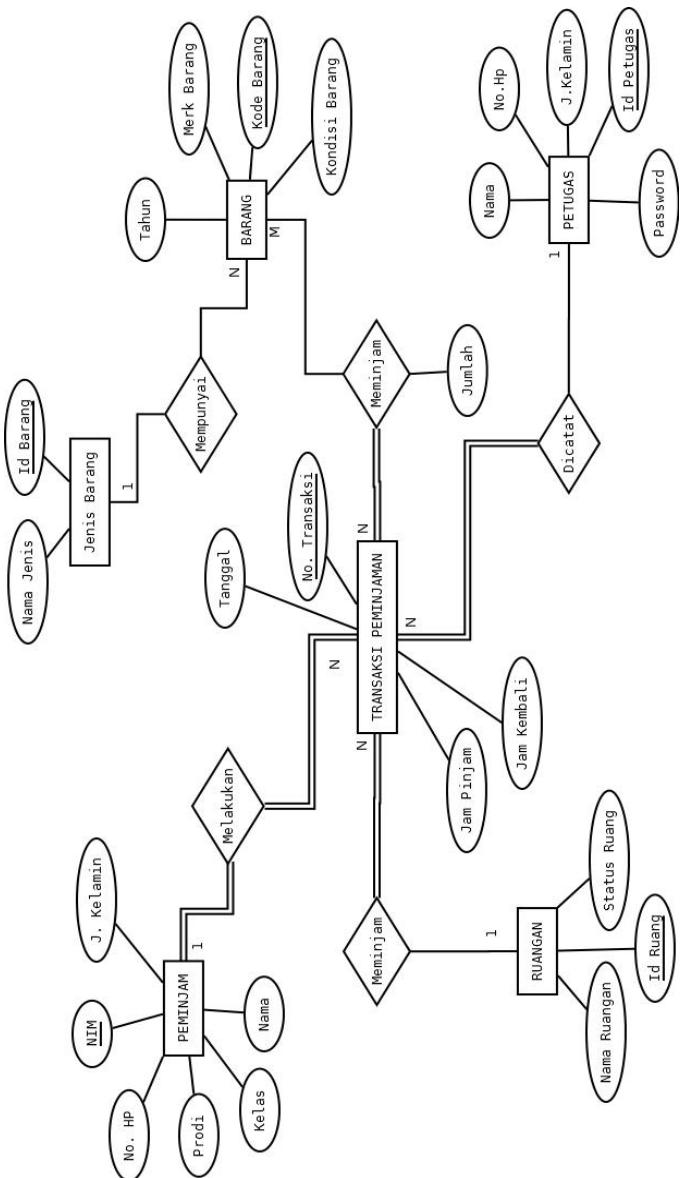
1. **Transaksi Peminjaman**, atribut : no.transaksi, tanggal, jam pinjam dan jam kembali.
2. **Peminjam**, atribut : NIM, nama, kelas, prodi, jenis kelamin dan no.hp
3. **Barang**, atribut : kode barang, merk barang, tahun dan kondisi barang
4. **Jenis Barang**, atribut : Id jenis, nama jenis
5. **Ruangan**, atribut : Id ruangan, nama ruangan dan status.
6. **Petugas**, atribut : Id petugas, nama petugas, jenis kelamin, no.hp dan password petugas.

Pertama, pada entitas Transaksi Peminjaman dan Peminjam memiliki relasi “melakukan” yang bersifat *one to many*, yang berarti satu peminjam hanya dapat melakukan

sekali transaksi disetiap no.transaksi dan begitu sebaliknya.Sedangkan pada relasi entitas Transaksi Peminjaman dan Barang mempunyai relasi “meminjam” yang bersifat *many to many*, yang berarti dalam setiap transaksi, peminjam dapat meminjam beberapa barang namun tidak sebaliknya. Selain itu pada entitas Barang juga memiliki relasi dengan entitas Jenis Barang, sehingga dapat disimpulkan bahwa setiap barang memiliki jenisnya masing-masing yang dibedakan melalui Id Jenis barang, dan relasi yang tepat antar keduanya adalah “mempunya” yang bersifat *one to many*.

Kemudian untuk mengidentifikasi di ruang mana barang tersebut dipinjam, maka pada entitas Transaksi Peminjaman dan Ruangan harus saling berelasi, sehingga relasi yang tepat yang dimiliki antara dua entitas tersebut adalah “meminjam” yang bersifat *one to many*, maknanya dalam setiap peminjaman barang harus menyertakan letak ruangan yang digunakan untuk meminjam barang tersebut.

Selanjutnya, dalam setiap transaksi peminjaman tentu ada namanya Petugas yang berfungsi sebagai penanggung jawab dalam setiap transaksi peminjaman yang dilakukan, tentu diantara entitas Transaksi Peminjaman dan Petugas memiliki relasi “dicatat” yang bersifat *one to many*, yang berarti dalam beberapa transaksi akan atau dapat dicatat oleh satu petugas, namun tidak sebaliknya. Berikut merupakan gambaran CDM pada basis data manajemen aset yang telah kami buat :



**Gambar 1. CDM Basis Data Manajemen Aset TU Jurusan
Teknik Informatika**

B. *Physical Data Model (PDM)*

Fungsi dari pemetaan dari CDM ke *physical data model* (PDM) adalah agar kerja basis data dapat terlihat lebih mudah dan sesuai alur, dan dapat dijadikan *relationship* pada Microsoft Access. Berikut merupakan teknik pemetaan :

1. Semua entitas menjadi tabel.
2. Semua atribut *single value* akan menjadi field.
3. Jika ada atribut *multivalue* maka akan menjadi tabel baru yg berisi atribut kunci dari entitasnya dan atribut itu sendiri.
4. Jika ditemukan relasi *one to one*. Atribut kunci pada entitas sisi parsial akan menjadi *foreign key* pada entitas sisi total, diikuti dgn atribut yg ada pada relasi.
5. Jika ada relasi *one to many*. Atribut kunci dari sisi *one* akan jadi *foreign key* pada sisi N (*many*).
6. Jika ada relasi *many to many*, maka akan jadi tabel baru yg berisi semua atribut kunci entitas yg bernilai dan atribut yang ada pada relasi.

Pertama semua entitas akan menjadi tabel, sesuai dengan CDM yang ada (enam entitas), sehingga terdapat enam tabel pula pada PDM (Transaksi Peminjaman, Peminjam, Barang, Jenis Barang, Ruangan dan Petugas).

Berhubung pada CDM tidak ditemukan atribut yang *single value*, *multivalue* dan relasi entitas yang bersifat *one to one*, maka teknik pemetaan PDM yang kedua hingga

keempat tidak perlu digunakan. Sehingga secara otomatis pemetaan selanjutnya dilakukan pada teknik ke lima.

Selanjutnya jika ada relasi *one to many*, maka atribut kunci pada sisi *one* menjadi *foreign key* pada sisi *many*. Sehingga *key attribute* tabel peminjam (NIM), tabel ruangan (Id ruangan) dan tabel petugas (Id petugas) menjadi *foreign key* pada tabel transaksi peminjaman dan untuk *key attribute* pada tabel jenis barang (Id jenis) menjadi *foreign key* pada tabel barang.

Sesuai dengan teknik keenam dalam pemetaan CDM ke PDM, maka relasi “meminjam” akan menjadi tabel baru yang berisi atribut berharga pada tabel transaksi peminjaman (no.transaksi), tabel barang (kode barang) dan atribut dari tabel tersebut yakni jumlah.

Berikut gambaran *Physical Data Model* (PDM) :

1. Peminjam

NIM + Nama + Kelas + Prodi + No. Hp

2. Petugas

Id Petugas + Nama + Jenis Kelamin + No. Hp + Password

3. Ruangan

Id Ruangan + Nama Ruangan + Status

4. Transaksi Peminjaman

No. Transaksi + Tanggal + Jam Pinjam + Jam Kembali + #NIM + #Id Petugas + #Id Ruangan

5. Jenis Barang

Id Jenis + Nama Jenis

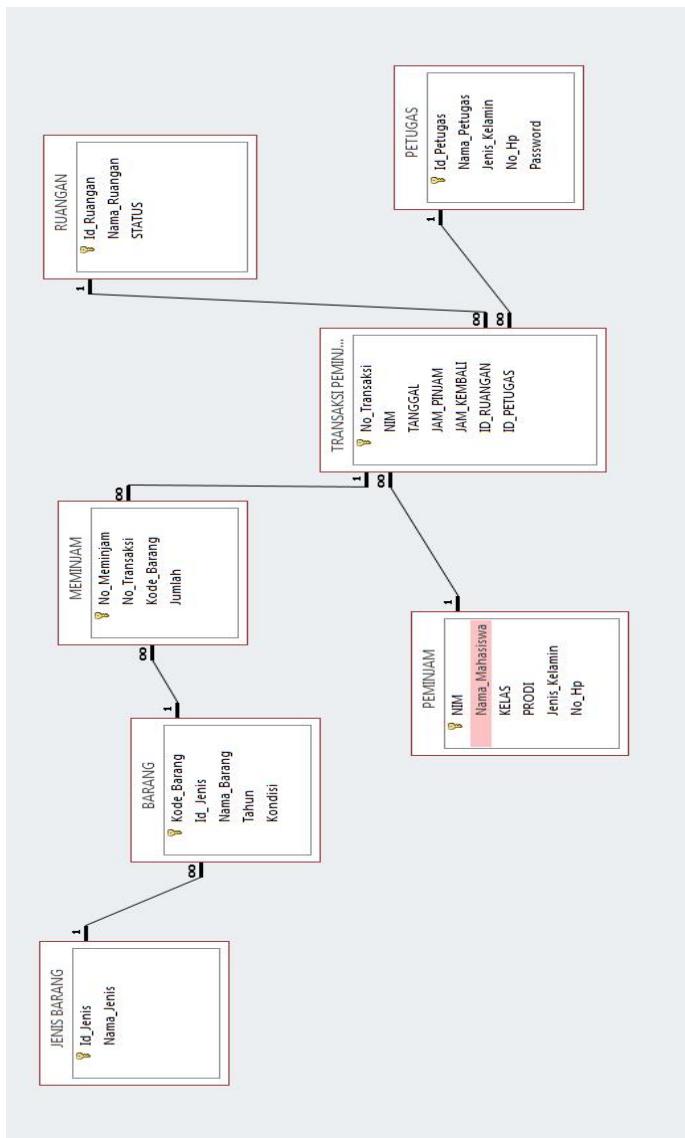
6. Barang

Kode Barang + Merk + Tahun + Kondisi + #Id Jenis

7. Meminjam

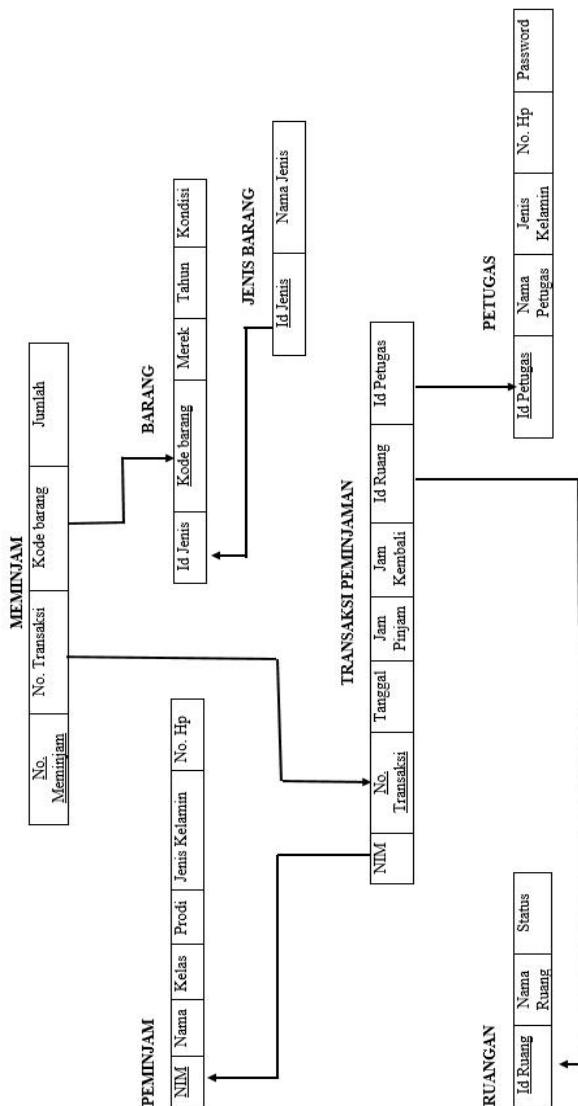
No. Peminjam + Jumlah + #No. Transaksi + #Kode Barang

Oleh karena itu, pemetaan ke *physical data model* (PDM) tentu sangat dibutuhkan untuk pembentukan *relationship* pada Ms. Access, berikut akan ditunjukkan bentuk *relationship* nya :



Gambar 2. Relationship Ms. Access pada Basis Data Manajemen Aset TU Jurusan Teknik Informatika

Layout *Physical Data Model*(PDM) :



Gambar 3. Layout PDM Basis Data Manajemen Aset TU Jurusan Teknik Informatika

C.Pengujian Normalisasi Tabel

Peminjam

NIM	Nama	Kelas	No.HP	Prodi	J.Kelamin
1NF		2NF		3NF	
<ul style="list-style-type: none">-Untuk normalisasi tahap pertama, field Peminjam sudah lolos-Alasan : Tidak ada field yang composite atau multivalue. Field Nama,Kelas, No._HP,NIM, Prodi, Jenis_Kelamin bersifat single value.-Tidak ada duplikasi antar data-Tidak ada penggandaan field		<ul style="list-style-type: none">-Untuk normalisasi tahap kedua, field Peminjam sudah lolos-Alasan : Sudah lolos dan sudah memenuhi normalisasi tahap pertama		<ul style="list-style-type: none">-Untuk normalisasi tahap ketiga, field Peminjam sudah lolos.-Alasan : Sudah lolos dan sudah memenuhi normalisasi tahap kedua	

Petugas

ID_Petugas	Nama	Jenis_Kelamin	No._HP	Password
1NF		2NF		3NF
<p>-Untuk normalisasi tahap pertama, field Petugas sudah lolos</p> <p>-Alasan : Tidak ada field yang composite atau multivalue. Bukti : field Nama,Jenis_Kelamin, No._HP dan Password bersifat single value / bernilai tungga</p>		<p>-Untuk normalisasi tahap kedua, field Petugas sudah lolos</p> <p>-Alasan : Sudah lolos dan sudah memenuhi normalisasi tahap pertama</p>		<p>-Untuk normalisasi tahap ketiga, field Petugas sudah lolos</p> <p>-Alasan : Sudah lolos dan sudah memenuhi normalisasi tahap kedua.</p>
		<p>-Sudah terdapat primary key didalam field "Petugas" yaitu ID_Petugas. Setiap field yang bukan primary key sudah bergantung penuh pada primary key.</p>		<p>-Setiap field sudah bergantung pada primary key utama field dan tidak bergantung pada field lain yang bukan primery key. Field Jenis_Kelamin, No._HP dan Password hanya bergantung pada primary key ID_Petugas.</p>

Barang

Kode_barang	Kondisi Barang	Merk_barang	Tahun
1NF	2NF	3NF	
<ul style="list-style-type: none">-Untuk normalisasi tahap pertama, field Barang sudah lolos-Alasan : Tidak ada field yang composite atau multivalue. Bukti : field Kode_barang, Kondisi_barang, Merk_Barang dan Tahun bersifat single value.-Tidak ada duplikasi antar data dan tidak ada penggandaan field	<ul style="list-style-type: none">-Normalisasi tahap kedua, field sudah lolos-Alasan : Sudah lolos dan sudah memenuhi normalisasi tahap pertama-Sudah terdapat primary key didalam field "Barang" yaitu Kode_barang.-Setiap field yang bukan primary key sudah bergantung penuh pada primary key.	<ul style="list-style-type: none">-Untuk normalisasi tahap ketiga, field sudah lolos.-Alasan : Sudah lolos dan sudah memenuhi normalisasi tahap kedua.-Setiap field yang bukan primary key dan tidak bergantung pada field lain yang bukan primary key.-Setiap field sudah bergantung pada primary key utama field dan tidak bergantung pada field lain yang bukan primery key.	

Ruangan

ID_Ruang	Jenis_Ruang	Status_Ruang
1NF	2NF	3NF
<p>-Untuk normalisasi tahap pertama, field Pegawai sudah lolos</p> <p>-Alasan : Tidak ada field yang composite atau multivalue. Bukti : field ID_Ruang, Jenis_Ruang dan Status_Ruang single value</p> <p>-Tidak ada duplikasi antar data dan tidak ada penggandaan field.</p>	<p>-Untuk normalisasi tahap kedua, field sudah lolos</p> <p>-Alasan : Sudah lolos dan sudah memenuhi normalisasi tahap pertama</p> <p>-Sudah terdapat primary key didalam field "Ruangan" yaitu ID_Ruang.</p> <p>-Setiap field yang bukan primary key sudah bergantung penuh pada primary key.</p>	<p>-Untuk normalisasi tahap ketiga, field sudah lolos.</p> <p>-Alasan : Sudah lolos dan sudah memenuhi normalisasi tahap kedua.</p> <p>-Setiap field yang bukan primary key dan tidak bergantung pada field lain yang bukan primary key.</p> <p>-Setiap field sudah bergantung pada primary key utama field dan tidak bergantung pada field lain yang bukan primary key.</p> <p>Bukti : field Jenis_Ruang dan Status_Ruang bergantung penuh pada ID_Ruang yang berposisi sebagai Primary Key</p>

Transaksi Peminjam

No._Transaksi	Tanggal	Jam_Pinjam	Jam_Kembali
1NF	2NF	3NF	
<p>-Untuk normalisasi tahap pertama, field Transaksi Peminjam sudah lolos</p> <p>-Alasan : Tidak ada field yang composite atau multivalue. Bukti : field No._Transaksi, Tanggal, Jam_Pinjam dan Jam_Kembali bersifat single value</p> <p>-Tidak ada duplikasi antar data.Tidak ada penggandaan field.</p>	<p>-Untuk normalisasi tahap kedua, field sudah lolos</p> <p>-Alasan : Sudah lolos dan sudah memenuhi normalisasi tahap pertama Sudah terdapat primary key didalam field "Transaksi Peminjam" yaitu No._Transaksi</p> <p>-Setiap field yang bukan primary key sudah bergantung penuh pada primary key</p>	<p>-Untuk normalisasi tahap ketiga, field sudah lolos.</p> <p>-Alasan : Sudah lolos dan sudah memenuhi normalisasi tahap kedua.</p> <p>-Setiap field yang bukan primary key dan tidak bergantung pada field lain yang bukan primary key.</p> <p>-Setiap field sudah bergantung pada primary key utama field dan tidak bergantung pada field lain yang bukan primary key. Bukti : Tanggal, Jam_Pinjam dan Jam_Kembali hanya bergantung pada satu primary key yaitu No._Transaksi</p>	

Jenis Barang

Id_Barang	Nama_Jenis	
1NF	2NF	3NF
<ul style="list-style-type: none">-Untuk normalisasi tahap pertama field Jenis_Barang sudah lolos-Alasan : Tidak ada field yang composite atau multivalue. Bukti : Nama_Jenis, Id_Barang bersifat single value / bernilai tunggal .-Tidak ada pengulangan maupun duplikasi data pada field transaksi.	<ul style="list-style-type: none">-Untuk normalisasi tahap kedua, field Jenis_Barang sudah lolos-Alasan : Sudah lolos dan sudah memenuhi normalisasi tahap pertama Sudah terdapat primary key didalam field "Jenis Barang" yaitu No_Resi .-Setiap field yang bukan primary key sudah bergantung penuh pada primary key.	<ul style="list-style-type: none">-Untuk normalisasi tahap ketiga, field Jenis_Barang sudah lolos.-Alasan : Sudah lolos dan sudah memenuhi normalisasi tahap kedua.-Setiap field sudah bergantung pada primary key utama field dan tidak bergantung pada field lain yang bukan primery key. Field Nama_Jenis hanya bergantung pada primary key ID_Barang

D. Algoritma Query

1. Berapa Kali Barang dipinjam

Aljabar Relasional

$$Q1 \leftarrow \pi_{Kode_{Barang}=[Forms]![Barang]![Combo65]}(MEMINJAM)$$
$$Result_{Tot} \leftarrow \text{Jumlah } F_{Sum}(Q1)$$

SQL

```
SELECT Sum(MEMINJAM.Jumlah) AS Tot
FROM MEMINJAM
GROUP BY MEMINJAM.Kode_Barang
HAVING
(((MEMINJAM.Kode_Barang)=[Forms]![BARANG]![Combo65]));
```

Misal [Forms]![BARANG]![Combo65]=K0A.2

Tot
3

2. Daftar Transaksi

Aljabar Relasional

$$Q1 \leftarrow \pi_{Kode_{Barang}, Nama_{Barang}, Jumlah, No_{Transaksi}}(MEMINJAM * BARANG)$$

Q2←

$$\pi_{Nama_{Mhs}, Nama_{Ruang}, Nama_{Petugas}, No_{Transaksi}, NIM, Prodi, Kelas} (((Transaksi_Peminjaman * Peminjam) * Ruangan) * Petugas)$$
$$\text{Result} \leftarrow \pi_{Q2, Kode_{Barang}, Nama_{Barang}, Jumlah}(Q1 * Q2)$$

SQL

```
SELECT [TRANSAKSI PEMINJAMAN].No_Transaksi, [TRANSAKSI PEMINJAMAN].TANGGAL, [TRANSAKSI PEMINJAMAN].NIM, PEMINJAM.Nama_Mahasiswa, PEMINJAM.KELAS, PEMINJAM.PRODI, BARANG.Kode_Barang, BARANG>Nama_Barang, MEMINJAM.Jumlah, [TRANSAKSI PEMINJAMAN].JAM_PINJAM, [TRANSAKSI PEMINJAMAN].JAM_KEMBALI, RUANGAN.Id_Ruangan, PETUGAS>Nama_Petugas, PETUGAS.Id_Petugas  
FROM (RUANGAN INNER JOIN (PETUGAS INNER JOIN (PEMINJAM INNER JOIN [TRANSAKSI PEMINJAMAN] ON PEMINJAM.NIM = [TRANSAKSI PEMINJAMAN].NIM) ON PETUGAS.Id_Petugas = [TRANSAKSI PEMINJAMAN].ID_PETUGAS) ON RUANGAN.Id_Ruangan = [TRANSAKSI PEMINJAMAN].ID_RUANGAN) INNER JOIN (BARANG INNER JOIN MEMINJAM ON BARANG.Kode_Barang = MEMINJAM.Kode_Barang)  
ON [TRANSAKSI PEMINJAMAN].No_Transaksi = MEMINJAM.No_Transaksi;
```

Output

No_Tra	TANGGAL	NIM	Nama_Mahasiswa	KELAS	PRODI	Kode_Baran	Nama_Barang	Jumlah	JAM_P	JAM_KE	Id_Ruan	Nama_P	Id_P
3	08/11/2019	1705097428	Novita	2017A	PTI	KOD2.2	IPS 18A	1	12.00.00	15.00.00	A10.01.07	AUFA	11903
16	16/12/2019	1905121413	Yanuar Putra	2019A	TI	KOA.2	View Sonic 233	2	11.11.00	11.23.00	A10.03.02	PIPIT	11902
33	22/12/2019	1805120401	Suci Wahyuning	2018A	TI	KOC.1	JBL 601	2	18.35.00	22.30.00	A10.04.01	AUFA	11903
34	22/12/2019	1805120413	Jazaal Aufa	2018A	TI	KOA.2	View Sonic 233	1	10.00.00	12.00.00	A10.01.04	AUFA	11903
34	22/12/2019	1805120413	Jazaal Aufa	2018A	TI	KOD1.2	Matematika 18B	1	10.00.00	12.00.00	A10.01.04	AUFA	11903
35	23/12/2019	1705121417	Kevin H.	2017A	SI	KOB.1	L-Ion 78	2	09.00.00	11.00.00	A10.01.08	INTAN	11904
38	23/12/2019	1805121416	Salman Alfarizi	2017	PTI	KOB.1	L-Ion 78	1	07.00.00	08.00.00	A10.01.04	AUFA	11903
38	23/12/2019	1805121416	Salman Alfarizi	2017	PTI	KOD1.3	RPL 2017 A	1	07.00.00	08.00.00	A10.01.04	AUFA	11903
39	23/12/2019	1805120412	Chindy A.S	2018B	TI	KOB.2	Broco 11	1	07.45.00	08.30.00	A10.01.10	PIPIT	11902
39	23/12/2019	1805120412	Chindy A.S	2018B	TI	KOD1.2	Matematika 18B	1	07.45.00	08.30.00	A10.01.10	PIPIT	11902
42	23/12/2019	1805120412	Chindy A.S	2018B	TI	KOB.2	Broco 11	0	09.00.00	11.10.00	A10.01.08	ILMA	11901
42	23/12/2019	1805120412	Chindy A.S	2018B	TI	KOD2.2	IPS 18A	1	09.00.00	11.10.00	A10.01.08	ILMA	11901

3. Detail Transaksi Ajabar Relasional

$Q1 \leftarrow \pi_{Kode_{Barang}, Nama_{Barang}, Jumlah, No_{Transaksi}} (MEMINJAM * BARANG)$

$Q2 \leftarrow$

$\pi_{Nama_{Mhs}, Nama_{Ruang}, No_{Transaksi}, NIM, Prodi, Kelas} (((Transaksi_Peminjaman * Peminjam) * Ruangan))$

$Result \leftarrow \pi_{Q2*, Kode_{Barang}, Nama_{Barang}, Jumlah} (Q1 * Q2)$

SQL

```
SELECT [TRANSAKSI PEMINJAMAN].NIM, [TRANSAKSI PEMINJAMAN].No_Transaksi, MEMINJAM.Jumlah,
BARANG>Nama_Barang, [JENIS BARANG].Nama_Jenis,
[TRANSAKSI PEMINJAMAN].JAM_PINJAM, [TRANSAKSI PEMINJAMAN].JAM_KEMBALI
FROM [TRANSAKSI PEMINJAMAN] INNER JOIN ((JENIS BARANG) INNER JOIN BARANG ON [JENIS BARANG].Id_Jenis = BARANG.[Id_Jenis]) INNER JOIN MEMINJAM ON BARANG.Kode_Barang = MEMINJAM.Kode_Barang) ON [TRANSAKSI PEMINJAMAN].No_Transaksi = MEMINJAM.No_Transaksi;
```

Output :

NIM	No_Transak	Jumlah	Nama Barang	Nama_Jenis	JAM_PINJAM	JAM_KEMBALI
1805121416	38	1 RPL 2017 A	JURNAL-SENIN	07.00.00	08.00.00	
1805120412	39	1 Broco 11	KABEL OLOR	07.45.00	08.30.00	
1805120412	42	0 Broco 11	KABEL OLOR	09.00.00	11.10.00	
1805120413	34	1 Matematika 18B	JURNAL-SENIN	10.00.00	12.00.00	
1805120412	39	1 Matematika 18B	JURNAL-SENIN	07.45.00	08.30.00	
1705097428	3	1 IPS 18A	JURNAL-SELASA	12.00.00	15.00.00	
1805120401	40	1 IPS 18A	JURNAL-SELASA			
1805120412	42	1 IPS 18A	JURNAL-SELASA	09.00.00	11.10.00	
1805120401	33	2 JBL 601	SPEAKER	18.35.00	22.30.00	
1905121413	16	2 View Sonic 233		11.11.00	11.23.00	
1805120413	34	1 View Sonic 233		10.00.00	12.00.00	
1705121417	35	2 L-Ion 78	KABEL OLOR	09.00.00	11.00.00	
1805121416	38	1 L-Ion 78	KABEL OLOR	07.00.00	08.00.00	

4. Jumlah Pelayanan Petugas Aljabar Relasional

$Q1 \leftarrow \sigma_{ID_{Petugas}=[\text{Input ID_Petugas} :]}(Q\text{DaftarTransaksi})$

$Result_{Selama} \leftarrow \text{No_Transaksi } F_{Count}(Q1)$

SQL

```
SELECT Count([DAFTAR TRANSAKSI].No_Transaksi) AS Selama
FROM [DAFTAR TRANSAKSI]
GROUP BY [DAFTAR TRANSAKSI].Id_Petugas
HAVING ((([DAFTAR TRANSAKSI].Id_Petugas)=[ID_Petugas :]));
Misal ID_Petugas = 2
```

Output

Selama	▼
	2

5. Jumlah Transaksi per MHS

Aljabar Relasional

$$Q1 \leftarrow \sigma_{NIM=[\text{Input NIM}]}(Q\text{DaftarTransaksi})$$
$$\text{Result}_{CountOfNo,NIM,NamaMHS} \leftarrow \text{No_Transaksi } F_{Count}(Q1)$$

SQL

```
SELECT Count([DAFTAR TRANSAKSI].No_Transaksi) AS  
CountOfNo_Transaksi, [DAFTAR TRANSAKSI].NIM, [DAFTAR  
TRANSAKSI].Nama_Mahasiswa  
FROM [DAFTAR TRANSAKSI]  
GROUP BY [DAFTAR TRANSAKSI].NIM, [DAFTAR  
TRANSAKSI].Nama_Mahasiswa  
HAVING ((([DAFTAR TRANSAKSI].NIM)=[Masukkan NIM :]));
```

Misal NIM : 1805121416

Output :

CountOfNo_	NIM	Nama_Mahasiswa
2	1805121416	Salman Alfarizi

6. Berapa Kali Kelas DiPinjam

Aljabar Relasional

$$Q1 \leftarrow \sigma_{Kelas=[\text{Input Kelas}]}(Q\text{DaftarTransaksi})$$
$$Result_{Jumlah,Kelas} \leftarrow \text{No_Transaksi } F_{Count}(Q1)$$

SQL

```
SELECT Count([DAFTAR TRANSAKSI].No_Transaksi) AS
Jumlah, [DAFTAR TRANSAKSI].KELAS
FROM [DAFTAR TRANSAKSI]
GROUP BY [DAFTAR TRANSAKSI].KELAS
HAVING ((([DAFTAR TRANSAKSI].KELAS)=[Masukkan Kelas
Sesuai Prodi (2018A)]));
```

[Masukkan Kelas Sesuai Prodi (2018A) :] = 2017A

Output

Jumlah	KELAS
2	2017A

7. Letak Barang Dipinjam

Aljabar Relasional

$$Q1 \leftarrow \sigma_{KodeBarang=[\text{Input KodeBarang}]}(Q\text{DaftarTransaksi})$$
$$Result \leftarrow \pi_{Jumlah,Nama_{Barang},Id_{Ruang},Kode_{Barang}}(Q1)$$

SQL

```
SELECT [DAFTAR TRANSAKSI].Kode_Barang, [DAFTAR TRANSAKSI].Nama_Barang, [DAFTAR TRANSAKSI].Jumlah, [DAFTAR TRANSAKSI].Id_Ruangan  
FROM [DAFTAR TRANSAKSI]  
WHERE ((([DAFTAR TRANSAKSI].Kode_Barang)=[Masukkan Kode Barang :]));
```

Misal : Kode Barang = K0C.1

Output :

Kode_Baran	Nama_Barang	Jumlah	Id_Ruan
K0C.1	JBL 601	2	A10.04.01
*			

8. Transaksi Sesuai Tanggal

Aljabar Relasional

$$Q1 \leftarrow \sigma_{Tanggal=Date()}(QDaftarTransaksi)$$
$$Result \leftarrow \pi_{No_Ts, Tgl, NIM, Nama_MHS, Nama_Brg, Jmlh, Id_Ruang, Nama_Petugas}(Q1)$$

SQL

```
SELECT [DAFTAR TRANSAKSI].No_Transaksi, [DAFTAR TRANSAKSI].TANGGAL, [DAFTAR TRANSAKSI].NIM, [DAFTAR TRANSAKSI].Nama_Mahasiswa, [DAFTAR TRANSAKSI].Nama_Barang, [DAFTAR TRANSAKSI].Jumlah, [DAFTAR TRANSAKSI].Id_Ruangan, [DAFTAR TRANSAKSI].Nama_Petugas  
FROM [DAFTAR TRANSAKSI]  
WHERE ((([DAFTAR TRANSAKSI].TANGGAL)=Date()));
```

Output :

No_Tra	TANGGAL	NIM	Nama_Mahas	Nama_Barang	Jumlah	Id_Ruan	Nama_P
35	23/12/2019	1705121417	Kevin H.	L-Ion 78	2	A10.01.08	INTAN
38	23/12/2019	1805121416	Salman Alfarizi	L-Ion 78	1	A10.01.04	AUFA
38	23/12/2019	1805121416	Salman Alfarizi	RPL 2017 A	1	A10.01.04	AUFA
39	23/12/2019	1805120412	Chindy A. S	Broco 11	1	A10.01.10	PIPIT
39	23/12/2019	1805120412	Chindy A. S	Matematika 18B	1	A10.01.10	PIPIT
42	23/12/2019	1805120412	Chindy A. S	Broco 11	0	A10.01.08	ILMA
42	23/12/2019	1805120412	Chindy A. S	IPS 18A	1	A10.01.08	ILMA
*	(New)						

9. Status Kelas

Aljabar Relasional

Result $\leftarrow \sigma_{ID_Ruang=[\text{Input ID_Ruang}]}(Ruangan)$

SQL

```
SELECT RUANGAN.Id_Ruangan, RUANGAN.Nama_Ruangan,  
RUANGAN.STATUS  
FROM RUANGAN  
WHERE (((RUANGAN.Id_Ruangan)=[Masukkan Id Ruangan :]));  
Misal ID_Ruangan = A10.01.07
```

Id_Ruangan	Nama_Ruangan	STATUS
A10.01.07	KELAS	TIDAK DIPAKAI

10. Jumlah Transaksi Hari Ini

Aljabar Relasional

Q1 $\leftarrow \sigma_{Tanggal=Date_0}(QDaftarTransaksi)$

Result_{Jmlh,Tgl} \leftarrow No_Transaksi F_{Count}(Q1)

SQL

```
SELECT Count([DAFTAR TRANSAKSI].No_Transaksi) AS [Jmlh  
transaksi Hari ini], [DAFTAR TRANSAKSI].TANGGAL  
FROM [DAFTAR TRANSAKSI]  
GROUP BY [DAFTAR TRANSAKSI].TANGGAL  
HAVING ((([DAFTAR TRANSAKSI].TANGGAL)=Date()));
```

Jmlh transal	TANGGAL
7	23/12/2019

BAB III

IMPLEMENTASI BASIS DATA

A. *Graphical User Interface (GUI)*

Pada bagian ini akan dijelaskan *Interface* pada semua form yang ada di database, selain itu akan dijelaskan spesifikasi dari masing-masing form. Berikut ini penjelasannya :

b. Form Login



Gambar 4. GUI Form Login Database

Form Log in merupakan tampilan awal bagi *User* (petugas), pada form ini petugas diminta untuk memasukkan Id dan Password yang telah dimiliki agar dapat masuk ke menu utama. Berikut spesifiknya :

- **Button Login**

Jenis Button : Open Form

Event : On Click [Event Procedure]

```
Option Compare Database

Private Sub Command1_Click()
Dim o As Object
Set o = CurrentDb.OpenRecordset("SELECT * FROM PETUGAS WHERE Id_Petugas ='" & user & "' AND Password ='" & sandi & "'")
If o.RecordCount > 0 Then
    'Data Ada
    DoCmd.Close
    DoCmd.OpenForm "Home", acNormal
    Forms!Home!Id.Caption = o!ID_PETUGAS
    Forms!Home!nama.Caption = o!Nama_Petugas

    MsgBox "Selamat datang"

    Dim db As DAO.Database
    Set db = CurrentDb

    Else
        'Data Tidak ada
        MsgBox "User tidak ada / data ada yang salah"
    End If
End Sub
```

- **Button Cancel**

Jenis Button : Close Window

Event : On Click [Embedded Macro]

Close Window

- Object Type : -
- Object Name : -
- Save : Yes

Quit Access

- Option : Save All

c. Form HOME

Form Home merupakan tampilan selanjutnya apabila *user* (petugas), berhasil log in menggunakan Id dan Password yang dimiliki. Kemudian user akan disajikan pilihan halaman utama berupa tambah transaksi (menambah peminjaman) dan admin untuk mengolah data aset yang dipunya.



Gambar 5. GUI Form Home Database

Kemudian apabila *user* telah berhasil Login, maka secara otomatis Id dan Nama *user* akan muncul pada Home (**pada pojok kanan bawah**), hal tersebut akibat *Event Procedure* pada form sebelumnya. Selain itu *user* juga dapat melihat hari, tanggal dan tahun pada tampilan pojok kiri atas serta jam yang akan selalu *update* setiap waktunya. Berikut spesifikasinya :

- **Button Logout**

Jenis Button : Open Form
Event : On Click [Embededeb Macro]
Close Window
- Object Type : Form
- Object Name : Home
- Save : Yes
Open Form
- Form Name : Form Login
- View : Form
- Data Mode : Add
- Window Mode : Normal

- **Button Tambah Transaki**

Jenis Button : Open Form
Event : On Click [Embededeb Macro]
Open Form
- Form Name : FORM_TRANSAKSI
- View : Form
- Data Mode : Add
- Window Mode : Normal

- **Button Admin**

Jenis Button : Open Form
Event : On Click [Embededeb Macro]

Open Form

- Form Name : Form Halaman

Admin

- View : Form
- Data Mode : Add
- Window Mode : Normal

Close Window

- Object Type : Form
- Object Name : Home
- Save : Yes

d. FORM_TRANSAKSI

Seperti yang sudah dijelaskan pada bab sebelumnya, FORM_TRANSAKSI merupakan form yang paling penting, sebab pada form ini akan menjadi media utama pada proses peminjaman dan *redord* data.

Pada form ini, *user* diminta untuk memasukkan data diri yang sudah dimiliki melalui KTM (NIM), kemudian *user* (petugas) yang akan memproses transaksi yang ada.

The screenshot shows a user interface for a database transaction system. At the top left is the logo of Universitas Negeri Semarang (Unesa) followed by the title "Transaksi Peminjaman". Below the title is the subtitle "Tata Usaha Jurusan Teknik Informatika Unesa". On the right side, there is a "Back" button and a small profile picture. The main area is divided into two sections: a form on the left and a grid on the right.

Form Section:

- Header: NO. (New)
- Fields: NIM, NAMA, TANGGAL, KELAS, PRODI, JAM PINJAM, JAM KEMBALI, RUANGAN. Each field has a dropdown arrow icon.
- A blue "+" button is located next to the NIM field.

Grid Section:

- Header: Daftar Pinjaman Barang
- Fields: Nama_Barang, Jenis, Jumlah.
- Data: A single row is visible with an asterisk (*) in the Nama_Barang column.
- Record navigation: Record: 1 of 1, Previous, Next, Last, No Filter, Search.

Buttons:

- Pinjam Barang ? (A blue button)
- Action buttons: New, Save, Refresh, Delete.

Bottom Left:

ii9o3 AUFA

Gambar 6. GUI Form Transaksi Database

Panel dengan heading bertuliskan “Trasaksi Peminjaman Tata Usaha dan Logo Jurusan Teknik Informatika” , foto profil user dan terdapat :

Button Back

Close Window

- Object Type : -
- Object Name : -
- Save : Yes

Selain itu, didalam Form Terdapat **Label** yang bertuliskan :

- | | |
|-----------------|---------------|
| - No. Transaksi | - Jama Pinjam |
| - NIM | - Jam Kembali |
| - Nama | - Ruangan |
| - Tanggal | |
| - Kelas | |
| - Prodi | |

Didalam Form Terdapat **Text Box** yang bertuliskan :

- | | |
|------------------|---------------|
| - No. Transaksi | - JAM_PINJAM |
| - NIM | - JAM_KEMBALI |
| - Nama_Mahasiswa | - ID_RUANGAN |
| - TANGGAL | |
| - KELAS | |
| - PRODI | |

Kemudian terdapat Button yakni :

Button Tambah Peminjam

Jenis Button : Open Form

Event : On Click [Embededeb Macro]

Open Form

- Form Name : Form Peminjam
- View : Form
- Data Mode : Add
- Window Mode : Dialog

Close Window

- Object Type : Form
- Object Name : FORM_TRANSAKSI
- Save : Yes

Button Detail Peminjaman Barang (Pinjam Barang?)

Jenis Button : Open Form

Event : On Click [Embededeb Macro]

Open Form

- Form Name : Form MEMINJAM
- View : Form
- Data Mode : Add
- Window Mode : Dialog

Button Tambah Pinjaman Baru (New)

Jenis Button : Add
Event : On Click [Embededeb Macro]
OnError
 Go to : Next
GoToRecord
 Object Type : -
 Object Name : -
 Record : New
If [MacroError]<>0 **Then**
 MessageBox
 Mesangge : [MacroError].[Description]
 Beep : Yes
 Type : None
End If

Button Simpan Data Pinjaman (Save)

Jenis Button : Save
Event : On Click [Embededeb Macro]
OnError
 Go to : Next
 Macro Name : -
RunMenuCommand
 Command : Save Record
If [MacroError]<>0 **Then**
 MessageBox
 Mesangge : [MacroError].[Description]
 Beep : Yes
 Type : None
End If

Button Hapus Data Pinjaman (Delete)

Jenis Button : Delete
Event : On Click [Embededeb Macro]
OnError
 Go to : Next
 Macro Name : -
GoToControl
 Control Name :
 =[Screen].[PreviousControl].[Name]
ClearMacroError
If Not [Form].[NewRecord] **Then**
 RunMenuCommand
 Command :DeleteRecord
End If
If [Form].[NewRecord] And Not [Form].[Dirty]
Then
 Beep
End If

If [Form].[NewRecord] And Not [Form].[Dirty]
Then
 RunMenuCommand
 Command : UnderRecord
End If

If [MacroError]<>0 **Then**
MessageBox
 Mesangge :[MacroError].[Description]
 Beep : Yes
 Type : None
End If

Button Refresh Data Pinjaman (Refresh)

Jenis Button : Refresh
Event : On Click [Embededeb Macro]

OnError

Go to : Next

Macro Name : -

RunMenuCommand

Command : Save Record

If [MacroError]<>0 Then

MessageBox

Mesangge : [MacroError].[Description]

Beep : Yes

Type : None

End If

e. Form Peminjam

Pada saat proses transaksi tentu tidak semua Peminjam (Mahasiswa) pernah melakukan transaksi. Oleh karena itu, pada form Peminjam sangat berguna bagi peminjam baru jika ingin melakukan transaksi (peminjaman), maka tugas *user* untuk melayani pendaftaran peminjam.

The screenshot shows a Java Swing application window titled "Pendaftaran Peminjam". At the top left is the logo of the Faculty of Computer Science at Unesa, which is a stylized blue 'f' with a yellow dot above it, followed by the text "INFORMATIKA UNESA". To the right of the title is the subtitle "Tata Usaha Jurusan Teknik Informatika Unesa". The main area contains six input fields arranged vertically. Each field has a label on the left and a text input box on the right. The labels are: "NIM", "NAMA", "KELAS", "PRODI", "GENDER", and "NO HP". The "KELAS", "PRODI", and "GENDER" fields include small dropdown arrows to the right of the input boxes. At the bottom of the window are two buttons: "DAFTAR" on the left and "BACK" on the right.

Gambar 7. GUI Form Peminjam (Pendaftaran Peminjam)

Panel dengan heading bertuliskan “Trasaksi Peminjaman Tata Usaha dan Logo Jurusan Teknik Informatika”. Selain itu, didalam Form Terdapat **Label** yang bertuliskan :

- | | |
|---------|----------|
| - NIM | - Gender |
| - Nama | - No. Hp |
| - Kelas | - Prodi |

Didalam Form Terdapat **Text Box yang bertuliskan :**

- NIM - Jenis_Kelamin
- Nama_Mahasiswa - No_Hp
- TANGGAL
- KELAS
- PRODI

Button Daftar

Jenis Button : Add

Event : On Click [Event Procedure]

```
Private Sub Command23_Click()  
  
Me.Refresh  
  
Forms!FORM_TRANSAKSI!Combo216 = Me.NIM  
  
DoCmd.Close  
  
End Sub
```

Button Back

Jenis Button : Open Form

Event : On Click [Embedded Macro]

Close Window

- Object Type : Form
- Object Name : PEMINJAM
- Save : No

f. Form Meminjam (Detail Peminjaman)

Dalam proses peminjaman, setiap peminjaman wajib menambahkan detail barang yang akan dipinjam, yang selanjutnya data tersebut secara otomatis akan ter *record* pada Query Daftar Transaksi. Berikut *Graphical User Interface (GUI)* nya :

The screenshot shows a Java Swing application window titled "Detail Peminjaman". In the top left corner is the logo of the Faculty of Computer Science at Unesa, which consists of a stylized blue 'f' with a yellow dot above it, followed by the text "INFORMATIKA UNESA". To the right of the logo is the title "Detail Peminjaman" in a large, bold, white font. Below the title, the subtitle "Tata Usaha Jurusan Teknik Informatika Unesa" is displayed in a smaller, italicized white font. The main content area has a teal background. It contains four input fields: "BARANG" (a dropdown menu), "JENIS" (a text field), "TAHUN" (a text field), and "JUMLAH" (a text field). Below these fields is a blue "SIMPAN" button.

Gambar 8. GUI Detail Transaksi Database.

Panel dengan heading bertuliskan “Trasaksi Peminjaman Tata Usaha dan Logo Jurusan Teknik Informatika”. Selain itu, didalam Form Terdapat **Label** yang bertuliskan :

- Barang (Combo Box) - Tahun
- Jenis - Jumlah

Didalam Form Terdapat **Text Box yang bertuliskan :**

- Kode_Barang - Tahun
- Nama_Jenis - Jumlah

Selain itu, erdapat button simpan yang berfungsi untuk menyimpan data detail peminjaman, setelah data tersimpan maka secara otomatis tampilan akan kembali pada tampilan awal (Form_Transaksi) dan secara otomatis detail pinjaman akan terekam pada daftar pinjaman barang, berikut spesifikasinya :

Button Simpan

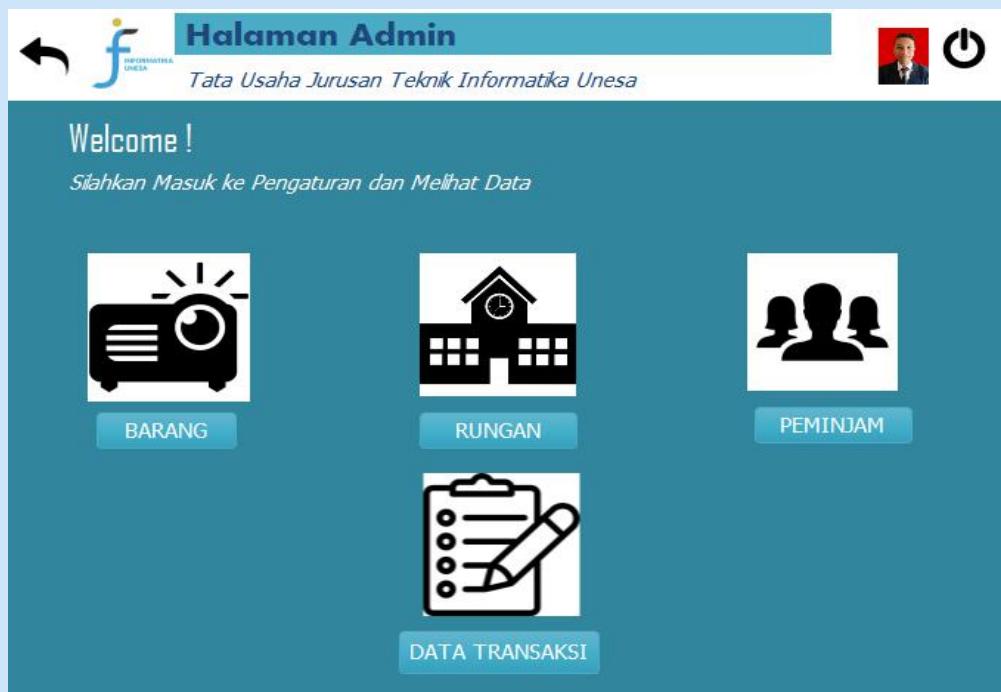
Jenis Button : Open Form

Event : On Click [Event Procedure]

```
Private Sub Command31_Click()  
  
Me.No_Transaksi = Forms!FORM_TRANSAKSI!No_Transaksi  
  
DoCmd.Close  
  
Forms!FORM_TRANSAKSI!subform.Form.Requery  
  
End Sub
```

g. Form Halaman Admin

Pada Form Halaman Utama (Home) terdapat dua pilihan button yang disajikan untuk *user* (petugas), salah satunya ialah Admin. Dalam tugasnya Admin memiliki tugas sebagai pengolah data dan mendata keluar masuknya data dalam aplikasi database ini. Berikut ini tampilan yang akan muncul apabila *user* menekan button Admin :



Gambar 9. GUI Halaman Admin Database

Dalam tampilan disajikan empat pilihan yang berguna untuk mengelola dan memantau data yang keluar masuk pada database, yakni :

1. Button Barang (Menambah Barang)
2. Button Ruangan (Menambah dan Mengubah status ruangan)
3. Button Peminjam (Melihat transaksi berdasarkan NIM peminjam)
4. Data Transaksi (Melihat dan Mengubah transaksi yang pernah dilakukan)

Untuk spesifikasi pada tabel Form Halaman Admin hanya terdapat tombol button terbagi menjadi enam bagian, antara lain :

Button Back

Jenis Button : Open Form

Event : On Click [Embedded Macro]

Open Form

- Form Name : Form Home
- View : Form
- Data Mode : Read Only
- Window Mode : Dialog

Button Logout

Jenis Button : Open Form

Event : On Click [Embeded Macro]

Close Window

- Object Type : Form
- Object Name : Halaman Admin
- Save : No

Open Form

- Form Name : Form Login
- View : Form
- Data Mode : Read Only
- Window Mode : Normal

Button Barang

Berfungsi untuk menambah data barang pada database.

Jenis Button : Open Form

Event : On Click [Embeded Macro]

Open Form

- Form Name : Form BARANG
- View : Form
- Data Mode : Add
- Window Mode : Normal

Button Ruangan

Berfungsi untuk menambah dan mengubah data ruangan.

Jenis Button : Open Form

Event : On Click [Embedded Macro]

Open Form

- Form Name : Form RUANGAN
- View : Form
- Data Mode : Add
- Window Mode : Normal

Button Peminjam

Berfungsi untuk mengetahui detail transaksi yang dilakukan peminjam berdasarkan NIM.

Jenis Button : Open Query

Event : On Click [Embedded Macro]

Query Form

- Query Name : Jumlah Transaksi per Mahasiswa
- Data Mode : Datasheet
- Window Mode : Edit

Button Daftar Transaksi

Bertujuan agar *user* dapat mengetahui dan memantau transaksi yang telah dilakukan setiap waktunya.

Jenis Button : Open Form

Event : On Click [Embedded Macro]

Query Form

- Form Name : DAFTAR TRANSAKSI subform
- View : Datasheet
- Data Mode : Edit
- Window Mode : Dialog

h. Form Barang

Form ini diperuntukkan bagi *user* (Admin) yang ingin menambah data barang yang ada dalam database. Selain itu, pada form ini juga mempunyai data barang yang meliputi : Kode_Barang, Id_Jenis, Nama_Barang dan Tahun. Berikut tampilannya :

The screenshot shows a Windows application window titled "BARANG". At the top left is the logo of the Faculty of Computer Science (Fakultas Informatika) with the text "Tata Usaha Jurusan Jurusan Teknik Informatika Unesa". The main area has a teal header bar with the title "BARANG". Below the header are four input fields: "KODE", "JENIS" (with a dropdown arrow), "MERK", and "TAHUN" (with a dropdown arrow). To the right of these fields is a button with a plus sign (+). To the right of the input fields is a grid table titled "DAFTAR BARANG" containing the following data:

Kode_Barang	Id_Jenis	Nama_Barang	Tahun
KOD1.2	KOD1	Matematika 18B	2018
KOD1.3	KOD1	RPL 2017 A	2019
KOD2.1	KOD2	Matematika 19 B	2019
KOD2.2	KOD2	IPS 18A	2018

At the bottom of the grid are buttons for "Record: 14 4 8 of 13" and "No Filters". Below the grid are four buttons: "Tambah", "Simpan", "Refresh", and "Hapus".

Gambar 10. GUI Form Barang (Tambah Barang) Database

Panel dengan heading bertuliskan “Trasaksi Peminjaman Tata Usaha dan Logo Jurusan Teknik Informatika”. Selain itu, didalam Form Terdapat **Label** yang bertuliskan :

- | | |
|---------------------|---------------------|
| - Kode | - Merk |
| - Jenis (Combo Box) | - Tahun (Combo Box) |

Didalam Form Terdapat **Text Box** yang bertuliskan :

- | | |
|---------------|--------------|
| - Kode_Barang | - Nama_Jenis |
| - Id_Jenis | - Tahun |

Pada tampilanm ini terdapat beberapa button untuk mengelola data terutama dalam hal menambah barang, antara lain :

Button Back

Jenis Button : Open Form

Event : On Click [Embeded Macro]

Open Form

- Form Name : Form Home
- View : Form
- Data Mode : Read Only
- Window Mode : Dialog

Button Tambah Jenis Barang

Jenis Button : Open Form

Event : On Click [Embeded Macro]

Open Form

- Form Name : Form Jenis Barang
- View : Form
- Data Mode : Add
- Window Mode : Dialog

Button ini berfungsi untuk menambah jenis barang baru pada database, setelah *user* berhasil memasukkan Id Jenis baru, maka *user* secara otomatis dapat memasukkan Kode dan Nama Barang sesuai dengan Id Jenis baru yang telah dimasukkan.

Apabila proses penambahan Id Jenis sudah selesai, maka akan kembali pada pemabahan barang. Jika hendak menambahkan barang, maka *user* wajib mengisi **label** kosong yang telah disediakan.

Ketika data sudah diisi, maka *user* menekan **button simpan** (untuk menyimpan penambahan barang) dan **button refresh** untuk merefresh data agar dapat ter *record* pada subform DATA BARANG. Berikut spesifikasinya :

Button Simpan Data Pinjaman (Save)

Jenis Button : Save

Event : On Click [Embeded Macro]

OnError

Go to : Next

Macro Name : -

RunMenuItem

Command : Save Record

If [MacroError]<>0 **Then**

MessageBox

Mesangge : [MacroError].[Description]

Beep : Yes

Type : None

Title : -

Button Refresh Data Pinjaman (Refresh)

Jenis Button : Refresh

Event : On Click [Embeded Macro]

OnError

Go to : Next

Macro Name : -

RunMenuCommand

Command : Save Record

If [MacroError]<>0 Then

MessageBox

Mesangge : [MacroError].[Description]

Beep : Yes

Type : None

End If

Kemudian untuk mereset proses penambahan data barang maka user dapat menekan **Button Tambah**.

Button Tambah Barang

Jenis Button : Add

Event : On Click [Embededeb Macro]

OnError

Go to : Next

GoToRecord

Object Type :-

Object Name :-

Record : New

If [MacroError]<>0 Then

MessageBox

Mesangge : [MacroError].[Description]

Beep : Yes

Type : None

End If

Selain itu, apabila data yang sudah terlanjur dimasukkan dan ter *record* pada daftar barang namun kurang sesuai dengan data sebenarnya ingin diinputkan, maka *user* dapat menekan **Button Delete** untuk mengahpus data barang yang sudah direkam (ter *record*).

Button Hapus Data Barang (data yang sudah terekam)

Jenis Button : Delete

Event : On Click [Embededeb Macro]

OnError

Go to : Next

Macro Name : -

GoToControl

Control Name :

= [Screen].[PreviousControl].[Name]

ClearMacroError

If Not [Form].[NewRecord] Then

RunMenuCommand

Command : DeleteRecord

End If

If [Form].[NewRecord] And Not [Form].[Dirty]

Then

Beep

End If

If [Form].[NewRecord] And Not [Form].[Dirty]

Then

RunMenuCommand

Command : UnderRecord

End If

If [MacroError]<>0 Then

MessageBox

Mesangge : [MacroError].[Description]

Beep : Yes

Type : None
End If

i. Form Id Jenis (Menambah Jenis Barang Baru)

Form ini bertujuan untuk menambahkan jenis barang yang belum ada pada database sebelumnya, yang harus diinputkan ialah Id Jenis (jenis baru) dan Nama Jenis (nama jenis baru). Berikut tampilannya :



Gambar 11. GUI Jenis Barang (tambah jenis barang) Database

Pada Form ini *user* diminta untuk memasukkan Jenis Barang baru yang dibutuhkan dalam Instansi dengan cara menginputkan (**button add**) data pada **text box** yang telah tersedia. Apabila *user* sudah masukkan data yang diminta, maka selanjutnya bagi *user* untuk menyimpan data yang telah diinputkan (**button save**) dan tekan **button refresh** agar data dapat muncul dan ter *record* pada subform DATA BARANG yang terdapat

pada Form Barang (Form Sebelumnya). Berikut spesifikasi masing-masing buttonya :

Button Tambah Jenis Baru (Add)

Jenis Button : Add

Event : On Click [Embeded Macro]

OnError

Go to : Next

GoToRecord

Object Type : -

Object Name : -

Record : New

If [MacroError]<>0 **Then**

MessageBox

Mesangge : [MacroError].[Description]

Beep : Yes

Type : None

End If

Button Back

Jenis Button : Open Form

Event : On Click [Embeded Macro]

Open Form

- Form Name : Form Home
- View : Form
- Data Mode : -
- Window Mode : Dialog

Button Simpan Data Jenis Barang baru (Save)

Jenis Button : Save
Event : On Click [Embededeb Macro]
OnError
 Go to : Next
 Macro Name : -
RunMenuCommand
 Command : Save Record

 If [MacroError]<>0 **Then**
 MessageBox
 Mesangge : [MacroError].[Description]
 Beep : Yes
 Type : None
 Title : -
 End If

Button Refresh Data Jenis Barang Baru (Refresh)

Jenis Button : Refresh
Event : On Click [Embededeb Macro]
OnError
 Go to : Next
 Macro Name : -
RunMenuCommand
 Command : Save Record
 If [MacroError]<>0 **Then**
 MessageBox
 Mesangge : [MacroError].[Description]
 Beep : Yes
 Type : None
 End If

j. Form Ruangan (Menambah dan Mengubah data ruangan)

Dapat dilihat pada form Halaman Admin sebelumnya, bahwa terdapat **button ruangan** yang fungsinya untuk menambahkan dan mengubah status pada ruangan, maka hal yang harus dilakukan oleh *user* ialah menginputkan data-data yang dibutukan antara lain :

1. Id Ruangan
2. Nama Ruangan
3. Status

Berikut tampilan Form Ruangan, yang ditunjukkan pada **Gambar.12** :

The screenshot shows a web-based application for managing room data. At the top, there is a header bar with the logo 'f INFORMATIKA UNESA' and the title 'RUANGAN'. Below the header, the subtext 'Tata Usaha Jurusan Jurusan Teknik Informatika Unesa' is visible. The main area contains three input fields: 'ID RUANGAN', 'NAMA RUANG', and 'STATUS', each with a dropdown arrow. To the right of these fields are three buttons: 'ADD', 'SAVE', and 'REFRESH'. Below these fields is a section titled 'DAFTAR RUANGAN' containing a grid table. The table has columns: 'Nama_Ruangan', 'Id_Ruangan', and 'STATUS'. The data in the grid is as follows:

Nama_Ruangan	Id_Ruangan	STATUS
KELAS	A10.01.03	DIPAKAI
KELAS	A10.01.04	TIDAK DIPAKAI
KELAS	A10.01.05	DIPAKAI
KELAS	A10.01.06	DIPAKAI

At the bottom of the grid, there are navigation buttons for 'Record: 14 < 1 of 15 > >> No Filter Search'.

Gambar 12. GUI Form Ruangan (Menambah Ruangan) Database *Bu*

Panel dengan heading bertuliskan “Trasaksi Peminjaman Tata Usaha dan Logo Jurusan Teknik Informatika”. Selain itu, didalam Form Terdapat **Label** yang bertuliskan :

- Id Ruangan
- Status
- Nama Ruangan

Didalam Form Terdapat **Text Box** yang bertuliskan :

- Id_Ruangan (Combo Box)
- STATUS (Combo Box)
- Nama Ruangan (Combo Box)

Button Tambah (Add)

Jenis Button : Add

Event : On Click [Embeded Macro]

OnError

Go to : Next

GoToRecord

Object Type :-

Object Name :-

Record : New

If [MacroError]<>0 Then

MessageBox

Mesangge : [MacroError].[Description]

Beep : Yes

Type : None

Button Simpan (Save)

Jenis Button : Save
Event : On Click [Embededeb Macro]
OnError
 Go to : Next
 Macro Name : -
RunMenuCommand
 Command : Save Record
 If [MacroError]<>0 **Then**
MessageBox
 Mesangge : [MacroError].[Description]
 Beep : Yes
 Type : None
 Title : -
End If

Button Refresh (Refresh)

Jenis Button : Refresh
Event : On Click [Embededeb Macro]
OnError
 Go to : Next
 Macro Name : -
RunMenuCommand
 Command : Save Record
 If [MacroError]<>0 **Then**
MessageBox
 Mesangge : [MacroError].[Description]
 Beep : Yes
 Type : None

Button Back

Jenis Button : Open Form

Event : On Click [Embedded Macro]

Open Form

- Form Name : Form Home
- View : Form
- Data Mode : -
- Window Mode : Dialog

Button Back

Jenis Button : Open Form

Event : On Click [Embedded Macro]

Open Form

- Form Name : Form Halaman Admin
- View : Form
- Data Mode : -
- Window Mode : Normal

B. Report

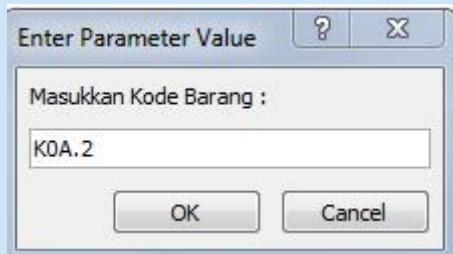
Pada database Manajemen Aset ini ada banyak yang yang perlu dijadikan laporan bagi *user*, tujuannya agar dapat memudahkan hal hal yang dibutuhkan *user* dan mengatasi hal hal yang tidak diinginkan. Oleh karena itu, perlu dilakukannya report.

a. Report 1 : Berapa kali barang tersebut dipinjamkan ?

Query pada kesempatan kali ini sangat diperlukan untuk menampilkan berapa kali barang Kode X dipinjam.

Query yang dibutuhkan adalah **Berapa kali barang tersebut dipinjam**, berikut hasil reportnya :

Masukkan Kode Barang yang pernah dipinjamkan :



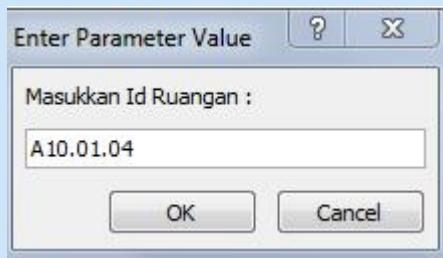
NO. TRANSAKSI	TANGGAL	NIM	KODE BARANG	MERK	Jumlah
16	16/12/2019	1905121413	KOA.2	View Sonic 233	2
34	22/12/2019	1805120413	KOA.2	View Sonic 233	1

Gambar 13. Report 1: Berapa Kali Barang X dipinjamkan ?

b. Report 2 : Status Ruangan

Dalam setiap peminjaman *user* perlu mengetahui status ruangan yang akan dipinjamkan, karena jika ingin meninjam barang, seorang peminjam wajib menyertakan letak ruangan. Untuk membuat report status ruangan, yang dibutuhkan ialah **Query Status Ruangan**, berikut hasil reportnya :

Masukkan Id Ruangan :



Maka secara langsung pada report akan menampilkan status pada ruangan berdasarkan Id Ruangan yang telah dicari. Output :



STATUS RUANGAN

Tata Usaha Jurusan Jurusan Teknik Informatika

23 Desember 2019
4:02:40

ID RUANGAN	NAMA RUANGAN	STATUS
A10.01.04	KELAS	TIDAK DIPAKAI

Gambar 14. Report 2 : Status Ruangan

c. Report 3 : Transaksi Berdasarkan Tanggal

Pada report ini bertujuan untuk mengetahui transaksi yang dilakukan berdasarkan tanggal transaksi, tentu ini sangat memudahkan *user* dalam mengetahui transaksi di hari-hari tertentu. Untuk memfungsikan Report ini pada database, maka digunakan **Query Transaksi Sesuai Tanggal** sebagai pemanggilnya. Berikut hasil reportnya :

Masukkan Tanggal Transaksi yang dicari

Enter Parameter Value

Masukkan Tanggal Transaksi (dd/mm/yyyy) :

OK Cancel

Maka pada report akan muncul detail transaksi yang dilakukan pada tanggal itu, berdasarkan transaksi yang pernah dilakukan sebelumnya. Output :



TRANSAKSI BERDASARKAN TANGGAL		23 Desember 2019
Tata Usaha Jurusan Teknik Informatika		
NO.	16	PETUGAS
TANGGAL	16/12/2019	MERK
NIM	1905121413	JUMLAH
NAMA	Yanuar Putra	ID RUANGAN

Gambar 15. Report 3 : Transaksi Berdasarkan Tanggal

d. Report 3 : cetak Nota

Dalam setiap transaksi yang dilakukan pengguna tentu ada bukti transaksi yang akan ditujuan kepada peminjam. Sehingga *user* dapat mengetahui detail Transaksi yang dilakukan oleh peminjam.

Syarat agar nota transaksi dapat dicetak adalah harus memasukkan dan menyimpan data Transaksi peminjaman terlebih dahulu. Berikut tampilan Reportnya :

CETAK TRANSAKSI			
NO	43	TANGGAL	23/12/2019
ID PETUGAS	LIMA	JAM PINJAM	12:00:00
NIM	1805120413	JAM KEMBALI	15:00:00
NAMA	Jazeel Aufa	ID RUANG	A10.01.08
KELAS	2018A		

Gambar 16. Report 3 : Cetak Nota

BAB IV

HASIL UJI COBA SISTEM BASIS DATA

A. SEKENARIO 1

Studi Kasus

Seorang mahasiswa baru jurusan teknik informatika bernama **Chindy A. S TI 2018A**, ingin meminjam barang yaitu jurnal Matakuliah Matematika 18B untuk absensi kelas dan satu buah kabel olor. Akhirnya segera Ia pergi ke tata usaha jurusan teknik infomatika, walaupun pada saat itu Ia hanya memiliki KTMS (Kartu Tanda Mahasiswa Sementara) dan ini kali pertama Ia melakukan peminjaman

Hasil Uji Coba Aplikasi

Pertama *user* (petugas) masuk pada form log in, untuk memasukkan Id_Petugas dan Password yang telah dimiliki sebelumnya. ***Id Petugas : 11903*** atas Nama : PIPIT dengan ***Password : ab123***. Kemudian tekan **LOGIN** untuk masuk Menu Utama (Home) dan **Cancel** untuk keluar.



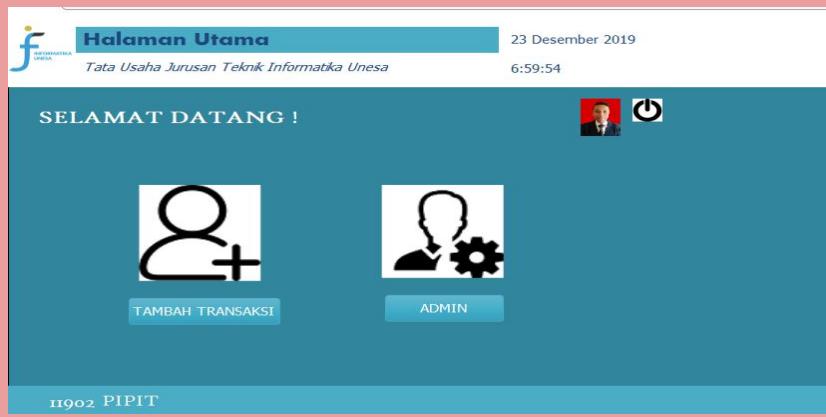
Gambar 17. Sekeranrio 1 : Form Login

Setelah *user* berhasil masuk pada form Login maka *user* akan secara otomatis masuk pada Form Home.



Gambar 18. Skenario 1 : Halaman Utama

Maka secara otomatis *Id Petugas* dan *Nama Petugas* akan muncul pada Halaman Utama (HOME). Kemudia apabila ingin melanjutkan transaksi maka **Tekan Button OK'**



Gambar 19. Skenario 1 : Halaman Utama 2

Sesuai studi kasus diatas, *user* akan melayani suatu peminjaman pada mahasiswa. Sehingga tekan **button Tambah Transaksi** untuk melakukan peminjama. Sehingga tampilan selanjutnya ialah :

The screenshot shows a web-based application titled "Transaksi Peminjaman" (Loan Transaction) from "Tata Usaha Jurusan Teknik Informatika Unesa". The main interface has a teal header and a white body. On the left, there's a form with fields for "NO." (labeled "(New)" with a plus sign), "NIM", "NAMA", "TANGGAL", "KELAS", "PRODI", "JAM PINJAM", "JAM KEMBALI", and "RUANGAN". Below the form are two buttons: "MEMINJAM" and "CETAK NOTA". At the bottom are four buttons: "New", "Simpan", "Refresh", and "Hapus". On the right, there's a "Daftar Pinjaman Barang" (List of Borrowed Items) table with columns for "Nama_Barang", "Jenis", and "Jumlah". The table currently shows one record: an asterisk (*) in the first column. At the bottom of the table are buttons for "Record", "No Filter", and "Search". A small black arrow points to the "plus" sign in the "NO." field.

Gambar 20. Skenario 1 : Tampilan Transaksi Peminjaman

Meskipun berlanjut pada page selanjutnya, data *user* yang diinputkan pada form Login sebelumnya tetap ter *record*. Perlu diketahui, Peminjam merupakan mahasiswa baru dan belum pernah melakukan transaksi sebelumnya.

Sehingga dapat disimpulkan bahwa mahasiswa tersebut adalah pengguna baru. Sehingga *user* harus menekan **button tambah** untuk menambah peminjam, berikut tampilan nya :

Pendaftaran Peminjaman
Tata Usaha Jurusan Teknik Informatika Unesa

NIM	
NAMA	
KELAS	
PRODI	
GENDER	
NO HP	

DAFTAR **BACK**



Kemudian InputData Peminjam baru pada Form Pendaftaran Peminjam (perlu diingat NIM harus diisi dengan baik dan sesuai).

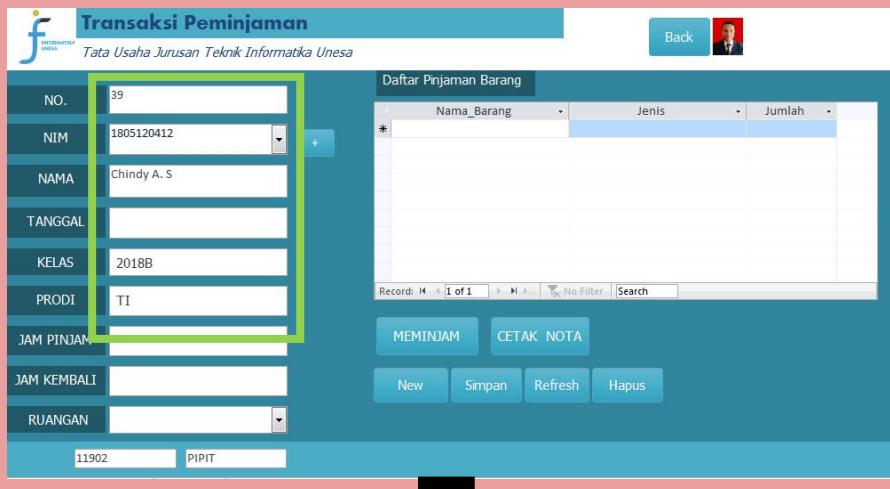
Pendaftaran Peminjaman
Tata Usaha Jurusan Teknik Informatika Unesa

NIM	1805120412
NAMA	Chindy A. S
KELAS	2018B
PRODI	TI
GENDER	P
NO HP	0890-8890-8999

DAFTAR **BACK**

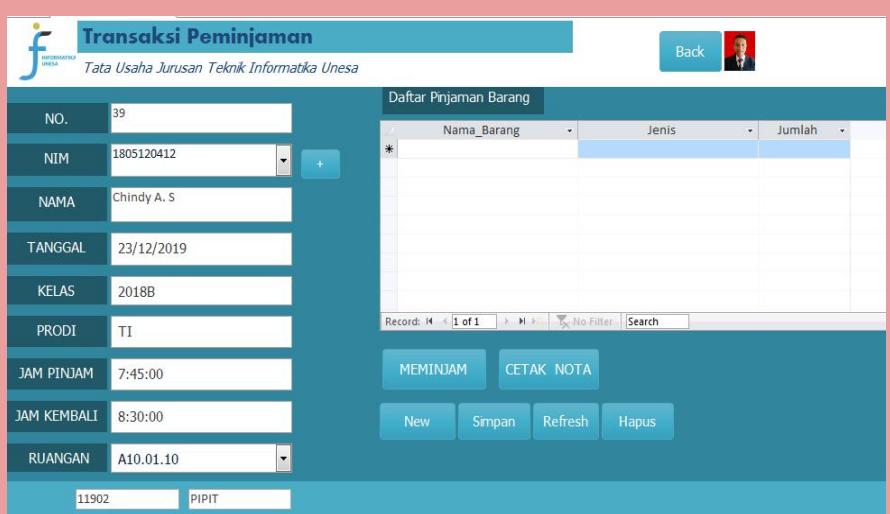
Gambar 21. Skenario 1 : Tampilan sebelum dan setelah InputdataForm Pendaftaran

Maka pada tampilan selanjutnya data diri peminjam baru dapat masuk pada **Label NIM**, hal ini berarti peminjam baru **sudah resmi terdaftar** dan dapat melakukan transaksi peminjaman dengan cara **InputData** dan **Choosoedata**. Berikut proses tampilannya :



The screenshot shows the 'Transaksi Peminjaman' application interface. On the left, there is an input form with fields: NO. (39), NIM (1805120412), NAMA (Chindy A. S), TANGGAL (empty), KELAS (2018B), PRODI (TI), JAM PINJAM (empty), JAM KEMBALI (empty), and RUANGAN (empty). Below the form are two status bars: 11902 and PIPIT. On the right, there is a data grid titled 'Daftar Pinjaman Barang' with columns: Nama_Barang, Jenis, and Jumlah. The grid shows one record: * (empty). At the bottom of the screen are several buttons: MEMINJAM, CETAK NOTA, New, Simpan, Refresh, and Hapus.

↓



The second screenshot shows the same application interface after data entry. The input form now includes the entered data: NO. (39), NIM (1805120412), NAMA (Chindy A. S), TANGGAL (23/12/2019), KELAS (2018B), PRODI (TI), JAM PINJAM (7:45:00), JAM KEMBALI (8:30:00), and RUANGAN (A10.01.10). The status bars remain at 11902 and PIPIT. The data grid on the right still shows one record: * (empty). The bottom buttons are the same: MEMINJAM, CETAK NOTA, New, Simpan, Refresh, and Hapus.

Gambar 22. Sekenario 1 : Input Rincian peminjaman Form Transaksi Peminjaman

Setelah petugas selesai **InputData** transaksi peminjaman yang dibutuhkan dalam Form Transaksi Peminjaman, maka tugas dari *user* ialah menekan **button MEMINJAM** untuk menambahkan detail peminjaman (barang yang akan dipinjam).

Sesuai dengan studi kasus yang ada, maka detail peminjaman (barang) pertama yang dipinjam ialah **berupa kabel olor**, sehingga *user* akan **Select** data barang yang dibutuhkan, kemudian tekan **button Simpan** untuk me *record* detail peminjaman (barang) pada form Transaksi

The screenshot shows a Windows application window titled "Detail Peminjaman". At the top left is the logo of "f INFORMATIKA UNESA". The main title bar says "Detail Peminjaman" and the subtitle below it is "Tata Usaha Jurusan Teknik Informatika Unesa". The subform displays the following data:

BARANG	Broco 11
JENIS	KABEL OLOR
TAHUN	2017
JUMLAH	1

At the bottom right of the subform is a blue "SIMPAN" button.

Gambar 23. Skenario 1 : Tampilan detail peminjaman (barang yang dipinjam) Pertama setelah tekan Button Meminjam

Setelah berhasil, maka secara otomatis detail pemijaman (barang) akan ter *record* pada subform **DataPinjamanBarang**. Berikut tampilannya :

Daftar Pinjaman Barang			
Nama Barang	Jenis	Jumlah	
Broco 11	KABEL OLOR	1	
*			
Record: 14	1 of 1	>	No Filter Search

Gambar 24. Sekenario 1 : Tampilan record Detail Peminjaman Pertama setelah
InputData

Selain kabel olor, peminjam (**Chindy**) juga ingin meminjam barang lain yakni **Jurnal Matakuliah Matematika 18B** untuk absensi kelas, maka *user* harus menekan **button Meminjam** nya kembali untuk menambah peminjamannya (barang) yakni **Matematika 18B**.

Detail Peminjaman	
Tata Usaha Jurusan Teknik Informatika Unesa	
BARANG	Matematika 18B
JENIS	JURNAL-SENIN
TAHUN	2018
JUMLAH	1
SIMPAN	

Gambar 25. Tampilan detail peminjaman (barang yang dipinjam) Ke -2 setelah tekan
Button Meminjam

Secara otomatis juga detail peminjaman terrecord pada subform **DetailPinjamanBarang** pada Form Transaksi.

Daftar Pinjaman Barang		
Nama_Barang	Jenis	Jumlah
Broco 11	KABEL OLOR	1
Matematika 18B	JURNAL-SENIN	1
*		

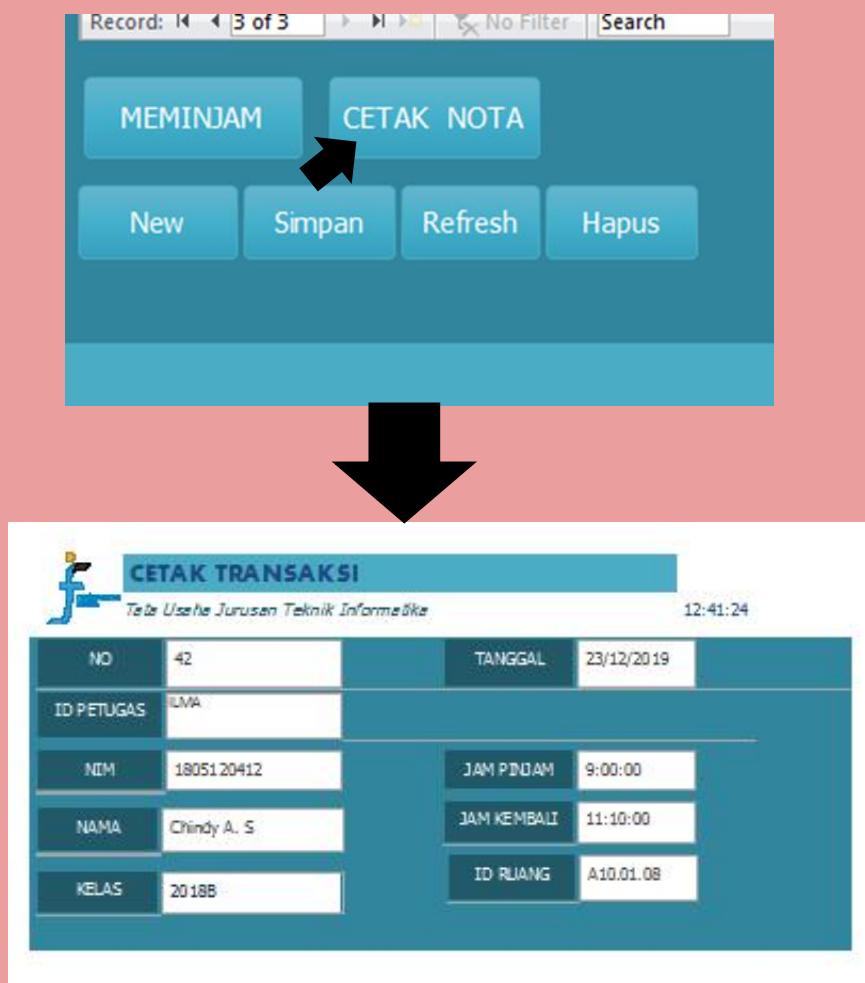
Gambar 26. Tampilan record Detail Peminjaman Kedua setelah InputData

Berhubung barang yang dipinjam hanya dua jenis barang, maka *user* wajib untuk menekan button **Simpan** untuk menyimpan datas selanjutnya dari proses Transaksi Peminjaman yang telah dilakukan.



Gambar 27. Tampilan Button pada Form Transaksi Prminjaman

Setelah data dan detail peminjaman sudah terisi dan sudah tersimpan, tugas *user* yang terakhir ialah mencetak nota (report) dari hasil transaksin yang dilakukan.



Gambar 28. Tampilan Cetak Nota

PENUTUP

1. Kekurangan

- data peminjam yang meminjam lebih dari 1 kali belum bisa menjadi satu
- aplikasi ini kurang simpel

2. Kelebihan

- lebih praktis dalam pendataan
- menghemat waktu dalam meminjam barang
- Menghemat kertas untuk mencatat aset yang akan dipinjam
- Meminimalkan rekап peminjaman yang hilang
- Memanfaatkan teknologi tepat guna

DAFTAR ISI

https://dspace.uii.ac.id/bitstream/handle/123456789/10073/SIMA_Bappeda.pdf?sequence=11&isAllowed=y. Diakses pada tanggal 20 Desember 2019

<https://lib.unnes.ac.id/17616/1/7101409095.pdf>. Diakses pada tanggal 11 Desember.

<https://www.maxmanroe.com/vid/manajemen/manajemen-aset.html>. Diakses pada tanggal 10 Desember

<https://www.gurupendidikan.co.id/pengertian-manajemen-aset/>. Diakses pada tanggal 12 Desember 2019

<https://sis.binus.ac.id/2014/10/06/tentang-manajemen-aset/>

<http://rahmatunnisya.web.ugm.ac.id/2017/11/12/membuat-database-dengan-microsoft-access/>