



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI

POLITEKNIK NEGERI SRJWIJAYA
JURUSAN MANAJEMEN INFORMATIKA

Jalan Srijaya Negara Bukit Besar - Palembang 30139 Telepon (0711) 353414
Laman : <http://polsri.ac.id> Pos El : info@polsri.ac.id

KESEPAKATAN BIMBINGAN TUGAS AKHIR

Kami yang bertanda tangan dibawah ini,

Pihak Pertama

Nama : Nina Yuliana
NIM : 061940830326
Jurusan : Manajemen Informatika
Program Studi : D-IV Manajemen Informatika

Pihak Kedua

Nama : Dr. Indri Ariyanti, SE.,M.Si.
NIP : 197306032008012008
Jurusan : Manajemen Informatika
Program Studi : D-IV Manajemen Informatika

Pada hari ini Selasa tanggal 28 Maret 2023 telah sepakat untuk melakukan konsultasi bimbingan Tugas Akhir.

Konsultasi bimbingan sekurang-kurangnya 2 (dua) kali dalam satu minggu. Pelaksanaan bimbingan pada setiap hari Selasa dan Kamis ber-tempat di Politeknik Negeri Sriwijaya.

Demikianlah kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Tugas Akhir.

Pihak Pertama

Nina Yuliana
NIM 061940830326

Palembang , 28 Maret 2023

Pihak Kedua

Dr. Indri Ariyanti, SE.,M.Si.
NIP 197306032008012008

Mengetahui,
a.n. Ketua Jurusan Manajemen Informatika

Mevi Kusnandar, S.Kom.,M.Kom.
NIP 197407052002121014





KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI

POLITEKNIK NEGERI SRIWIJAYA
JURUSAN MANAJEMEN INFORMATIKA

Jalan Sriwijaya Negara Bukit Besar - Palembang 30139 Telepon (0711) 353414

Laman : <http://polsri.ac.id>, Pos El : info@polsri.ac.id

KESEPAKATAN BIMBINGAN TUGAS AKHIR

Kami yang bertanda tangan dibawah ini,

Pihak Pertama

Nama : Nina Yuliana
NIM : 061940830326
Jurusan : Manajemen Informatika
Program Studi : D-IV Manajemen Informatika

Pihak Kedua

Nama : Sony Oktapriandi, S.Kom., M.Kom
NIP : 197510272008121001
Jurusan : Manajemen Informatika
Program Studi : D-VI Manajemen Informatika

Pada hari ini Senin tanggal 27 Maret 2023 telah sepakat untuk melakukan konsultasi bimbingan Tugas Akhir.

Konsultasi bimbingan sekurang-kurangnya 2 (dua) kali dalam satu minggu. Pelaksanaan bimbingan pada setiap hari Senin dan Kamis pukul 09.00 WIB tempat di Politeknik Negeri Sriwijaya.

Demikianlah kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Tugas Akhir.

Palembang, 27 Maret 2023

Pihak Pertama

Pihak Kedua

Nina Yuliana
NIM 06194083026

Sony Oktapriandi, S.Kom., M.Kom
NIP 197510272008121001

Mengetahui,
Ketua Jurusan Manajemen Informatika

Dr. Indri Ariyanti, SE., M.Si.
NIP 197306032008012008





KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI

POLITEKNIK NEGERI SRIWIJAYA
JURUSAN MANAJEMEN INFORMATIKA

Jalan Sriwijaya Negara Bukit Besar - Palembang 30139 Telepon (0711) 353414
Laman : <http://polsri.ac.id>, Pos El : info@polsri.ac.id

PENGAJUAN JUDUL TUGAS AKHIR

Kepada Yth,

Dosen Pembimbing I

Dr. Indri Ariyanti, SE., M.Si.

Di Politeknik Negeri Sriwijaya

Dengan Hormat,

Dalam rangka memenuhi syarat penyusunan Tugas Akhir, maka saya:

Nama : Nina Yuliana

NIM : 061940830326

Jurusan / Program Studi : Manajemen Informatika / D-IV Manajemen Informatika

Bermaksud mengajukan judul Tugas Akhir. Adapun judul Tugas Akhir yang saya ajukan yaitu:

- a c c
1. Penerapan Metode K-Means Clustering dalam Aplikasi Inventarisasi Fasilitas Kantor Berbasis Web Mobile pada PT PLN (Persero) UP3 Palembang
 2. Implementasi Metode PIECES pada Aplikasi Monitoring Inventarisasi Fasilitas Kantor Berbasis Website (Studi Kasus: PT PLN (Persero) UP3 Palembang)
 3. Penerapan Algoritma FIFO pada Aplikasi Stok Material Gudang PT PLN (Persero) UP3 Palembang Berbasis Website

Demikian surat pengajuan judul Tugas Akhir ini dibuat, sekiranya judul diatas dapat disetujui.

Atas persetujuan Ibu kami ucapkan terima kasih.

Palembang, 04 April 2023

Menyetujui,
Dosen Pembimbing I,


Dr. Indri Ariyanti, SE., M.Si.
NIP 197306032008012008

Hormat Saya,
Mahasiswa


Nina Yuliana
NIM 061940830326





KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI

POLITEKNIK NEGERI SRIWIJAYA
JURUSAN MANAJEMEN INFORMATIKA

Jalan Srijaya Negara Bukit Besar - Palembang 30139 Telepon (0711) 353414
Laman : <http://polsri.ac.id>, Pos El : info@polsri.ac.id

PENGAJUAN JUDUL TUGAS AKHIR

Kepada Yth,

Dosen Pembimbing II

Sony Oktapriandi, S.Kom., M.Kom.

Di Politeknik Negeri Sriwijaya

Dengan Hormat,

Dalam rangka memenuhi syarat penyusunan Tugas Akhir, maka saya:

Nama : Nina Yuliana

NIM : 061940830326

Jurusan / Program Studi : Manajemen Informatika / D-IV Manajemen Informatika

Bermaksud mengajukan judul Tugas Akhir. Adapun judul Tugas Akhir yang saya ajukan yaitu:

1. Penerapan Metode K-Means Clustering dalam Aplikasi Inventarisasi Fasilitas Kantor Berbasis Web Mobile pada PT PLN (Persero) UP3 Palembang
2. Implementasi Metode PIECES pada Aplikasi Monitoring Inventarisasi Fasilitas Kantor Berbasis Website (Studi Kasus: PT PLN (Persero) UP3 Palembang)
3. Penerapan Algoritma FIFO pada Aplikasi Stok Material Gudang PT PLN (Persero) UP3 Palembang Berbasis Website

Demikian surat pengajuan judul Tugas Akhir ini dibuat, sekiranya judul diatas dapat disetujui.

Atas persetujuan Bapak kami ucapkan terima kasih.

Palembang, 04 April 2023

Menyetujui,
Dosen Pembimbing II,

Sony Oktapriandi, S.Kom., M.Kom.
NIP 197510272008121001

Hormat Saya,
Mahasiswa

Nina Yuliana
NIM 061940830326





KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI

POLITEKNIK NEGERI SRIWIJAYA
JURUSAN MANAJEMEN INFORMATIKA

Jalan Srijaya Negara Bukit Besar - Palembang 30139 Telepon (0711) 353414
Laman : <http://polsri.ac.id>, Pos El : info@polsri.ac.id

LEMBAR PENGESAHAN JUDUL TUGAS AKHIR (TA)

Nama : Nina Yuliana
NIM : 061940830326
Jurusan / Program Studi : Manajemen Informatika/ DIV Manajemen Informatika
Judul Laporan Akhir : Penerapan Metode K-Means Clustering dalam Aplikasi Inventarisasi Fasilitas Kantor Berbasis Web Mobile pada PT PLN (Persero) UP3 Palembang

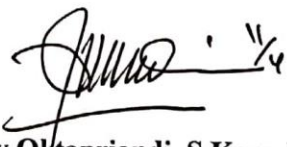
Palembang, 4 April 2023

Tim Pembimbing:

Pembimbing I

Pembimbing II


Dr. Indri Ariyanti, SE., M.Si
NIP 197306032008012008


Sony Oktapriandi, S.Kom., Kom
NIP 197510272008121001

Mengetahui,
a.n. Ketua Jurusan Manajemen Informatika


Meivi Kusnandar, S.Kom., M.Kom
NIP 197407052002121014





Palembang, 5 April 2023

Hal: Permohonan Surat Pengantar Pengambilan Data

Kepada Yth.
Wakil Direktur I
Politeknik Negeri Sriwijaya

Dengan Hormat,

Sesuai dengan kurikulum Jurusan Manajemen Informatika Politeknik Negeri Sriwijaya bahwa penyusunan Tugas Akhir merupakan mata kuliah pada semester VIII (delapan) yang diwajibkan bagi seluruh mahasiswa untuk dapat menyelesaikan pendidikan Diploma IV di Politeknik Negeri Sriwijaya, untuk itu saya yang bertanda tangan di bawah ini:

No	Nama	NIM	Kelas	Jurusan
1	Nina Yuliana	061940830326	8 MIA	Manajemen Informatika

Bermaksud untuk mengajukan surat pengantar pengambilan data yang ditunjukkan kepada:

Yth.

Manager PT PLN (Persero) UP3 Palembang

Jl. Kapten A. Rivai No.37, 24 Ilir, Kec. Bukit Kecil, Kota Palembang, Sumatera Selatan 30127

Untuk pengambilan data di **PT PLN (Persero) UP3 Palembang** dalam pembuatan aplikasi yang berjudul **"Penerapan Metode K-Means Clustering dalam Aplikasi Inventarisasi Fasilitas Kantor Berbasis Web Mobile pada PT PLN (Persero) UP3 Palembang"**

Demikianlah surat permohonan ini saya buat, atas bantuan dan kerjasamanya saya ucapkan terima kasih.

Mengetahui,
Ketua Jurusan Manajemen Informatika


Dr. Indri Andriyanti, SE., M.Si.
NIP 197306032008012008

Hormat saya,


Nina Yuliana
NIM 061940830326





KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI SRIWIJAYA

Jalan Sriwijaya Negara Bukit Besar – Palembang 30139 Telepon (0711) 353414
Laman : <http://polsri.ac.id>, Pos El : info@polsri.ac.id

Nomor : 3656/PL6.3.1/SP/2023
Perihal : Izin Pengambilan Data

17 April 2023

Yth. Manager
PT. PLN (Persero) UP3 Palembang
Jalan Kapten A. Rivai No.37
Kelurahan 24 Ilir Kecamatan Bukit Kecil
Palembang 30127

Dengan hormat,

Sesuai dengan kurikulum Program Sarjana Terapan (D-IV) Manajemen Informatika Jurusan Manajemen Informatika pada Politeknik Negeri Sriwijaya tahun akademik 2022/2023, bahwa Tugas Akhir merupakan mata kuliah wajib pada semester 8 (delapan). Untuk itu mahasiswa kami memerlukan data untuk penyusunan Tugas Akhir tersebut.

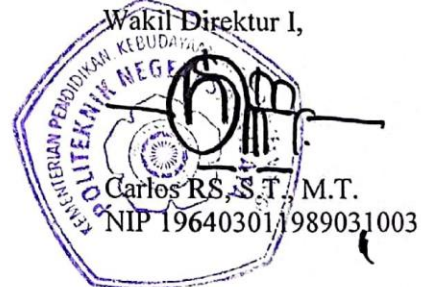
Sehubungan dengan hal tersebut di atas, kami mohon Bapak/Ibu dapat memberikan izin dan membantu mahasiswa kami ini untuk pengambilan data.

Mahasiswa kami yang akan melaksanakan pengambilan data tersebut adalah :

No	Nama	NPM	Kelas	Jurusan/ Program Studi
1	Nina Yuliana	0619 4083 0326	8 MI.A	Manajemen Informatika

Atas perhatian dan bantuannya diucapkan terima kasih.

a.n. Direktur
Wakil Direktur I,



Tembusan:

1. Ketua Jurusan Manajemen Informatika
2. Yang bersangkutan

Ms.Word/E/AD/Dw



Nomor : 1195 /STH.01.04/F11050000/2023
Lampiran : 1 Lembar
Sifat : Biasa
Perihal : Penerimaan Pengambilan Data

04 Mei 2023

Kepada

Yth. POLITEKNIK NEGERI SRIWIJAYA
Jl. Srijaya Negara Bukit Besar
Palembang

Menanggapi surat dari Politeknik Negeri Sriwijaya Nomor : 3656/PL6.3.1/SP/2023 tanggal 17 April 2023 perihal permohonan pengambilan data dengan ini kami informasikan pada prinsipnya PT. PLN (Persero) UP3 Palembang dapat mengizinkan Mahasiswa Saudara :

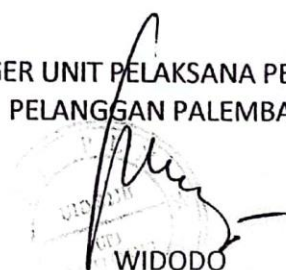
NO	NAMA	NIM	JURUSAN/PROGRAM STUDI
1	Nina Yuliana	0619 4083 0326	Manajemen Informatika

Kegiatan pengambilan data tersebut dilaksanakan mulai tanggal 03 Mei 2023 s.d 23 Mei 2023 setiap hari kerja (Senin s.d Jum'at) Pukul 07.30 s.d 16.00 WIB selama 14 (empat belas) hari kerja di PT. PLN (Persero) UP3 Palembang dengan mengikuti ketentuan sebagai berikut :

1. PLN tidak menanggung fasilitas dan biaya Mahasiswa selama pengambilan data.
2. Bersedia mendownload aplikasi PLN Mobile melalui play store/APP store.
3. Kecelakaan akibat kelalaian adalah diluar tanggung jawab PLN.
4. Mematuhi Protokol Kesehatan dengan memakai masker dan sudah bervaksinasi lengkap, vaksin 1 dan vaksin 2 selama berada dilingkungan kantor PT. PLN (Persero) UP3 Palembang.
5. Melaksanakan tugas dan tanggung jawab yang diberikan sesuai lokasi pengambilan data.

Demikian kami sampaikan, atas perhatian dan kerjasamanya diucapkan terima kasih.

MANAGER UNIT PELAKSANA PELAYANAN
PELANGGAN PALEMBANG


WIDODO

Tembusan :

- ASMAN KEUANGAN DAN UMUM PLN



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI

POLITEKNIK NEGERI SRIWIJAYA
JURUSAN MANAJEMEN INFORMATIKA

Jalan Srijaya Negara Bukit Besar - Palembang 30139 Telepon (0711) 353414
Laman : <http://polsri.ac.id>, Pos El : info@polsri.ac.id

LEMBAR BIMBINGAN TUGAS AKHIR (TA)

Lembar : 1

Nama : Nina Yuliana
NIM : 061940830326
Jurusan/Program Studi : Manajemen Informatika / DIV
Judul : Penerapan Metode K-Means Clustering dalam Aplikasi Inventarisasi Fasilitas Kantor Berbasis Web Mobile pada PT PLN (Persero) UP3 Palembang
Pembimbing I : Dr. Indri Ariyanti, SE.,M.Si.

No.	Tanggal	Uraian Bimbingan	Tanda Tangan Pembimbing
1.	4/4/2023	Pengertian Judul	
2.	7/4/2023	ACC judul	
3.	11/5/2023	Pengantar BAB I	
4.	11/5/2023	Revisi BAB I	
5.	15/5/2023	ACC BAB I	
6.	25/5/2023	Pengantar BAB II	
7.	23/5/2023	Revisi BAB II	
8.	9/6/2023	ACC BAB III	
9.	12/6/2023	Pengantar BAB III	



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI

POLITEKNIK NEGERI SRIWIJAYA
JURUSAN MANAJEMEN INFORMATIKA

Jalan Srijaya Negara Bukit Besar - Palembang 30139 Telepon (0711) 353414
Laman : <http://polsri.ac.id>, Pos El : info@polsri.ac.id

LEMBAR BIMBINGAN TUGAS AKHIR (TA)

Lembar : 2

10.	12/2023	Revisi BAB <u>IV</u>	2
11.	12/2023	ACC BAB <u>IV</u>	2
12.	17/2023	Pengajuan Bab <u>IV</u>	2
13.	17/2023	Pengajuan Bab <u>V</u>	2
14.	20/2023	Revisi BAB <u>IV</u> & <u>R</u>	2
15.	26/2023	ACC BAB <u>IV</u>	2
16.	26/2023	ACC BAB <u>V</u>	2
17.			
18.			

Palembang, 26 April 2023

a.n. Ketua Jurusan Manajemen Informatika

Meivi Kusnandar, S.Kom., M.Kom.
NIP 197407052002121014





KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI

POLITEKNIK NEGERI SRIWIJAYA
JURUSAN MANAJEMEN INFORMATIKA

Jalan Srijaya Negara Bukit Besar - Palembang 30139 Telepon (0711) 353414
Laman : <http://polsri.ac.id>, Pos El : info@polsri.ac.id

LEMBAR BIMBINGAN TUGAS AKHIR (TA)

Lembar : 1

Nama : Nina Yuliana
NIM : 061940830326
Jurusan/Program Studi : Manajemen Informatika / DIV
Judul : Penerapan Metode K-Means Clustering dalam Aplikasi Inventarisasi Fasilitas Kantor Berbasis Mobile pada PT PLN (Persero) UP3 Palembang
Pembimbing II : Sony Oktapriandi, S.Kom., M.Kom.

No.	Tanggal	Uraian Bimbingan	Tanda Tangan Pembimbing
1.	4/4/2013	Pengantar judul	
2.	4/5/2013	Bab I edit	
3.	8/5/2013	Bab I edit	
4.	10/5/2013	Bab I ok	
5.		Langkah bab II	
6.	12/5/2013	Bab II edit	
7.	28/5/2013	Bab II ok	
8.		Langkah bab III	
9.	15/6/2013	Bab III edit	





KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI

POLITEKNIK NEGERI SRIWIJAYA
JURUSAN MANAJEMEN INFORMATIKA

Jalan Srijaya Negara Bukit Besar - Palembang 30139 Telepon (0711) 353414
Laman : <http://polsri.ac.id>, Pos El : info@polsri.ac.id

LEMBAR BIMBINGAN TUGAS AKHIR (TA)

Lembar : 2

10.	24/6 2023	Bab III di lanjut bab IV	f
11.	20/7 2023	Bab IV, cari aplikasi	lv
12.	29/7 2023	Bab IV Tambah output	f
13.	04/8 2023	Bab IV di lanjut bab V	f
14.	5/8 2023	aplikasi oke, ppt ac	1
15.		Bab I & II + lanjut +	lv
16.		ber. Pustaka di	
17.		Boleh lanjut ujian TA.	
18.			

Palembang, 5 April 2023

Ketua Jurusan Manajemen Informatika

Dr. Indri Ariyanti, S.E., M.Si.
NIP 197306032008012008





**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN,
RISET, DAN TEKNOLOGI**
POLITEKNIK NEGERI SRIWIJAYA

Jalan Srijaya Negara Bukit Besar - Palembang 30139 Telepon (0711) 353414

Laman : <http://polsri.ac.id>, Pos El : info@polsri.ac.id

REKOMENDASI UJIAN LAPORAN TUGAS AKHIR (TA)

Pembimbing Laporan Tugas Akhir memberikan rekomendasi kepada,

Nama : Nina Yuliana
NPM : 061940830326
Jurusan/Program Studi : Manajemen Informatika/D-IV Manajemen Informatika
Judul Laporan Tugas Akhir : Penerapan Metode K-Means Clustering Dalam Aplikasi
Inventarisasi Fasilitas Kantor Berbasis Web Mobile Pada PT
PLN (Persero) UP3 Palembang

Mahasiswa tersebut telah memenuhi persyaratan dan dapat mengikuti Ujian Laporan Tugas Akhir (TA) pada Tahun Akademik 2022/2023

Palembang, 27 Juli 2023

Pembimbing I,


Dr. Indri Ariyanti, SE., M.Si
NIP. 197306032008012008

Pembimbing II,


Sony Oktapriandi, S.Kom., M.Kom.
NIP. 197510272008121001

Mengetahui,
a.n. Ketua Jurusan Manajemen Informatika


Meivi Kusnandar, S.Kom., M.Kom
NIP 197407052002121014



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

POLITEKNIK NEGERI SRIWIJAYA

Jalan Srijaya Negara Bukit Besar - Palembang 30139 Telepon (0711) 353414

Laman : <http://polsri.ac.id> Pos El : info@polsri.ac.id

LEMBAR PELAKSANAAN REVISI TUGAS AKHIR

Mahasiswa berikut,

Nama : Nina Yuliana
 NPM : 061940830326
 Jurusan/Program Studi : Manajemen Informatika/DIV
 Judul Tugas Akhir : Penerapan Metode *K-Means Clustering* Dalam Aplikasi Inventarisasi Fasilitas Kantor Berbasis *Web Mobile* Pada PT PLN (Persero) UP3 Palembang

Telah melaksanakan revisi terhadap Tugas Akhir yang diujikan pada hari Kamis tanggal 10 bulan Agustus tahun 2023. Pelaksanaan revisi terhadap Tugas Akhir tersebut telah disetujui oleh Dosen Penguji yang memberikan revisi :

No.	Komentar	Nama Dosen Penguji *)	Tanggal	Tanda Tangan
1.	Tambahkan laporan berapa jumlah barang yang baik dan buruk	Rika Sadariawati, SE.,M.Si	21/8-2023	
2.	Perbaiki sistem yang sedang berjalan Perbaiki perancangan Bab IV	Andre Mariza Putra, S.Kom.,M.Kom	21/08 2023	
3.	Perbaiki perhitungan K-Means Clustering Perbaiki sistem K-Means	Febie Elfaladonna, S.Kom.,M.Kom	21/2023/08	

Palembang, 21 Agustus 2023

Ketua Penguji **)

Rika Sadariawati, SE.,M.Si
 NIP 197302232002122001

Catatan:

- *) 1. Dosen penguji yang memberikan revisi saat ujian Tugas akhir.
- **) 2. Dosen penguji yang ditugaskan sebagai Ketua Penguji saat ujian TA;
- 3. Lembaran pelaksanaan revisi ini harus dilampirkan dalam Laporan Tugas Akhir;

DAFTAR INVENTARIS FASILITAS KANTOR & ATK
PT PLN (PERSERO) WS2JB AREA PALEMBANG

UNIT : Rayon Sekayu

NO.	NAMA BARANG	MERK/TYPE	NO. INVENTARIS	JML	STN	KONDISI	KET
	Ruang Manajer						
1	Kursi Direksi		1 /Sky/Mjr/X/2013	2	Buah	Baik	
2	Kursi Berlengan		2 /Sky/Mjr/X/2013	1	Buah	Baik	
3	Kursi Berlengan		3 /Sky/Mjr/X/2013	1	Buah	Baik	
4	Meja 1 Biro		4 /Sky/Mjr/X/2013	1	Buah	Baik	
5	Meja Kecil Komputer		5 /Sky/Mjr/X/2013	1	Buah	Baik	
6	Printer	Epson	6 /Sky/Mjr/X/2013	1	Buah	Baik	
7	Kulkas	Sharp	7 /Sky/Mjr/X/2013	1	Buah	Baik	
8	Brangkas		8 /Sky/Mjr/X/2013	1	Buah	Baik	
9	Lemari Kayu		9 /Sky/Mjr/X/2013	1	Buah	Baik	
10	AC 1 PK		10 /Sky/Mjr/X/2013	1	Buah	Baik	
11	Komputer		11 /Sky/Mjr/X/2013	1	Buah	Baik	
12	Meja Sofa		12 /Sky/Mjr/X/2013	1	Buah	Baik	
13	TV	LG	13 /Sky/Mjr/X/2013	2	Buah	Baik	
14	White Board		14 /Sky/Mjr/X/2013	1	Buah	Baik	
	Ruang Adm						
1	Meja 1/2 Biro		1 /Sky/Adm/X/2013	1	Buah	Baik	
2	Meja 1/2 Biro		2 /Sky/Adm/X/2013	1	Buah	Baik	
3	Brangkas/Lemari		3 /Sky/Adm/X/2013	1	Buah	Baik	
4	Meja 1 Biro		4 /Sky/Adm/X/2013	1	Buah	Baik	
5	Lemari Kayu		5 /Sky/Adm/X/2013	1	Buah	Baik	
6	Lemari Kayu		6 /Sky/Adm/X/2013	3	Buah	Baik	
7	Kalkulator		7 /Sky/Adm/X/2013	1	Buah	Baik	
8	Kursi Berlengan		8 /Sky/Adm/X/2013	1	Buah	Baik	
9	Kursi Berlengan		9 /Sky/Adm/X/2013	1	Buah	Baik	
10	Scanner	Zyrec	10 /Sky/Adm/X/2013	1	Buah	Baik	
11	Komputer	LG	11 /Sky/Adm/X/2013	1	Buah	Baik	
12	Komputer		12 /Sky/Adm/X/2013	1	Buah	Baik	
13	Kursi Putar	LG	13 /Sky/Adm/X/2013	1	Buah	Baik	
14	AC 1 PK		14 /Sky/Adm/X/2013	1	Buah	Baik	
15	Komputer		15 /Sky/Adm/X/2013	1	Buah	Rusak	
16	Komputer		16 /Sky/Adm/X/2013	1	Buah	Rusak	
17	Komputer		17 /Sky/Adm/X/2013	1	Buah	Rusak	
18	Printer		18 /Sky/Adm/X/2013	2	Buah	Baik	
	Ruang Loby PP						
1	Kursi Tunggu gandeng 3		1 /Sky/PP/X/2013	4	Buah	Baik	
2	Meja 1 Biro		2 /Sky/PP/X/2013	3	Buah	Baik	
3	Kursi putar berlengan		3 /Sky/PP/X/2013	1	Buah	Baik	
4	Kursi berlengan		4 /Sky/PP/X/2013	1	Buah	Baik	
5	Meja kaca kecil		5 /Sky/PP/X/2013	1	Set	Baik	
6	Kursi Tunggu gandeng 4		6 /Sky/PP/X/2013	1	Buah	Baik	
7	Komputer	Acer	7 /Sky/PP/X/2013	1	Set	Baik	
8	Kotak Sampah		8 /Sky/PP/X/2013	1	Buah	Baik	
	Ruang PP						
1	Meja 1 Biro		1 /Sky/PP/X/2013	8	Buah	Baik	
2	Meja 1/2 Biro		1 /Sky/PP/X/2013	5	Buah	Baik	
3	kursi Berlengan		1 /Sky/PP/X/2013	6	Buah	Baik	
4	FilingCabinet 2 pintu		1 /Sky/PP/X/2013	4	Buah	Baik	
5	FilingCabinet 4 Rak		1 /Sky/PP/X/2013	2	Buah	Baik	
6	Komputer		1 /Sky/PP/X/2013	8	Buah	Baik	
7	Printer		1 /Sky/PP/X/2013	9	Buah	Baik	
8	Meja Komputer		1 /Sky/PP/X/2013	5	Buah	Baik	
9	Facsimile		1 /Sky/PP/X/2013	1	Buah	Baik	
10	lemari Arsi Kayu 4 pintu		1 /Sky/PP/X/2013	2	Buah	Baik	
11	Lemari Kayu Kecil		1 /Sky/PP/X/2013	1	Buah	Baik	
12	Ac 1 PK		1 /Sky/PP/X/2013	1	Buah	Baik	
13	AC 2 PK		1 /Sky/PP/X/2013	2	Buah	Baik	
14	Kipas Angin		1 /Sky/PP/X/2013	3	Buah	Baik	
15	LemariKayu 4 Rak		1 /Sky/PP/X/2013	1	Buah	Baik	



NO.	NAMA BARANG	MERK/TYPE	NO. INVENTARIS	JML	STN	KONDISI	KET
16	Kursi Tunggu gandeng 3		16 /Sky/PP/X/2013	2	Buah	Baik	
17	Kursi Berlangan		17 /Sky/PP/X/2013	2	Buah	Baik	
18	Etalasi Hasil Kerja		18 /Sky/PP/X/2013	1	Buah	Baik	
19	Kursi Putar		19 /Sky/PP/X/2013	6	Buah	Baik	
20	Kursi Rotan		20 /Sky/PP/X/2013	1	Buah	Baik	
21	Apar		21 /Sky/PP/X/2013	3	Buah	Baik	
22	Monitor Komputer		22 /Sky/PP/X/2013	1	Buah	Rusak	
	Ruang Jaringan						
1	Meja 1/2 Biro		1 /Sky/JAR/X/2013	2	Buah	Baik	
2	Meja Komputer		2 /Sky/JAR/X/2013	2	Buah	Baik	
3	AC 1 PK		3 /Sky/JAR/X/2013	1	Buah	Baik	
4	Kursi Putar		4 /Sky/JAR/X/2013	3	Buah	Baik	
5	Dispenser		5 /Sky/JAR/X/2013	1	Buah	Baik	
6	Papan Umum Besar		6 /Sky/JAR/X/2013	1	Buah	Baik	
7	Apar		7 /Sky/JAR/X/2013	1	Buah	Baik	
8	Komputer		8 /Sky/JAR/X/2013	1	Buah	Baik	
9	Kursi Tunggu Kayu		9 /Sky/JAR/X/2013	1	Buah	Baik	
10	Meja 1/2 Biro		10 /Sky/JAR/X/2013	4	Buah	Baik	
11	Filing Cabinet 2 Pintu		11 /Sky/JAR/X/2013	2	Buah	Baik	
12	AC 1 PK		12 /Sky/JAR/X/2013	1	Buah	Baik	
13	Kursi Berputar berlangan		13 /Sky/JAR/X/2013	1	Buah	Baik	
14	Meja Komputer		14 /Sky/JAR/X/2013	1	Buah	Baik	
15	HAP	Toshiba	15 /Sky/JAR/X/2013	1	Buah	Baik	
16	Komputer/Printer		16 /Sky/JAR/X/2013	1	Buah	Baik	
17	Papan Umum Kecil		17 /Sky/JAR/X/2013	1	Buah	Baik	
18	Filing Cabinet 4 Rak	Panasonic	18 /Sky/JAR/X/2013	1	Buah	Baik	
19	Lemari Alat rak Kayu		19 /Sky/JAR/X/2013	1	Buah	Baik	
20	Dispenser		20 /Sky/JAR/X/2013	1	Buah	Baik	
	Ruang Gudang						
1	Filing Cabinet		1 /Sky/PP/X/2013	2	Set	Baik	
2	meja 1/2 Biro		2 /Sky/PP/X/2013	2	Set	Baik	
3	Kursi Putar		3 /Sky/PP/X/2013	1	Buah	Baik	
4	Kursi Kayu		4 /Sky/PP/X/2013	1	Buah	Baik	
	Kaja Babat Toman						
1	Meja 1/2 Biro		1 /Sky/BT/X/2013	1	Buah	Baik	
2	Meja 1/2 Biro		2 /Sky/BT/X/2013	1	Buah	Baik	
	Kaja Ngulak						
1	Meja 1/2 Biro		1/Sky/NGLK/X/2013	1	Buah	Baik	
2	Meja 1/2 Biro		2/Sky/NGLK/X/2013	1	Buah	Baik	
	Kaja Tebing Bulang						
1	Meja 1/2 Biro		1 /Sky/TB/X/2013	1	Buah	Baik	
2	Kursi		2 /Sky/TB/X/2013	1	Buah	Baik	

Palembang, Oktober 2013

Tim Inventaris Peralatan & Perlengkapan Kantor

MENGETAHUI;

MANAJER RAYON SEKAYU

ASMAN PP & ADM

1. Umriyanto

2. Mauladi

DANANG SETIAWAN

HARUN A. RONI

3. Widya sari

4. Ayu Lestari

5. Budi Setiawan

UID SUMATERA SELATAN, JAMBI DAN BENGKULU
UP3 PALEMBANG

Nomor : 1253 /STH.01.04/F11050000/2023
Lampiran : 1 Lembar
Sifat : Biasa
Perihal : Pembuatan Aplikasi

08 Mei 2023

Kepada

Yth. Jurusan DIV Manajemen
Informatika
Jl. Srijaya Negara Bukit Besar
Palembang

Menindak lanjuti Surat dari Politeknik Negeri Sriwijaya Nomor : 3656/PL6.3.1/SP/2023 dan surat kami Nomor : 1195/STH.01.04/F11050000/2023 tanggal 04 Mei 2023 Perihal : Izin Pengambilan Data yang berhubungan dengan sistim aplikasi masih manual.

Dengan ini kami sampaikan persetujuan pembuatan Aplikasi Inventarisasi Fasilitas Kantor Berbasis web Mobile pada PT. PLN (Persero) UP3 Palembang melalui penerapan Metode K-mean clustering.

Demikian kami sampaikan, atas perhatian dan kerjasamanya diucapkan terima kasih.

MANAGER UNIT PELAKSANA PELAYANAN
PELANGGAN PALEMBANG


WIDODO

Tembusan :
- ASMAN KEUANGAN DAN UMUM PLN





User.php

```
<?php
```

```
namespace App\Models;
```

```
//          use
```

```
    Illuminate\Contracts\Auth\Must  
    VerifyEmail;
```

```
use
```

```
    Illuminate\Database\Eloquent\F  
    actories\HasFac  
    tory;
```

```
use
```

```
    Illuminate\Foundation\Auth\Us  
    er as  
    Authenticatable  
    ;
```

```
use
```

```
    Illuminate\Notifications\Notifia  
    ble;
```

```
use
```

```
    Laravel\Sanctu  
    m\HasApiToke  
    ns;
```

```
class User extends  
    Authenticatable
```

```
{
```

```
    use HasApiTokens,  
        HasFactory,  
        Notifiable;
```

```
/**
```

```
 * The attributes that  
    are mass  
    assignable.
```

```
 *
```

```
 * @var array<int,  
    string>
```

```
 */
```

```
// protected $fillable =  
    [
```

```
    // 'name',
```

```
    // 'email',
```

```
    // 'password',
```

```
    // ];
```

```
protected $guarded =  
    ['id'];
```

```
protected $with =  
    ['unit'];
```

```
/**
```

```
 * The attributes that  
    should be  
    hidden for  
    serialization.
```

```
 *
```

```
 * @var array<int,  
    string>
```

```
 */
```

```
protected $hidden = [  
    'password',  
    'remember_token',  
];
```

```
/**
```

```
 * The attributes that  
    should be cast.
```

```
 *
```

```
 * @var array<string,  
    string>
```

```
 */
```

```
protected $casts = [  
    'email_verified_at'  
    => 'datetime',  
];
```

```
public function unit()
```

```
{
```

```
    return $this-  
        >belongsTo(Un  
        it::class);
```

```
}
```

```
}
```

Unit.php

```
<?php

namespace App\Models;

use Illuminate\Database\
    \Eloquent\Factories\
    HasFactory;

use Illuminate\Database
    \Eloquent\Model;

class Unit extends Model
{
    use HasFactory;

    protected $guarded = ['id'];

    public function user()
    {
        return $this-
            >hasMany(User::cla
                ss);
    }

    public function
    inventory()
    {
        return $this-
            >hasMany(Inventor
                y::class);
    }
}
```

Type.php

```
<?php

namespace App\Models;

use Illuminate\Database
    \Eloquent\Factories\
    HasFactory;

use Illuminate\Database
    \Eloquent\Model;

class Type extends Model
{
    use HasFactory;

    protected $guarded = ['id'];

    public function
```

```
inventory()
{
    return $this-
        >hasMany(Inventor
            y::class);
}
```

Room.php

```
<?php

namespace App\Models;

use Illuminate\Database
    \Eloquent\Factories\
    HasFactory;

use Illuminate\Database
    \Eloquent\Model;

class Room extends Model
{
    use HasFactory;

    protected $guarded = ['id'];

    public function
    inventory()
    {
        return $this-
            >hasMany(Inventor
                y::class);
    }
}
```

Inventory.php

```
<?php

namespace App\Models;

use Illuminate\Database
    \Eloquent\Factories\
    HasFactory;

use Illuminate\Database
    \Eloquent\Model;

class Inventory extends
    Model
{
    use HasFactory;

    protected $guarded = ['id'];
```

```
protected $with = ['unit',
    'ruangan',
    'sub_jenis'];
```

```
public function unit()
{
    return $this-
        >belongsTo(Unit::cl
            ass);
}

public function ruangan()
{
    return $this-
        >belongsTo(Room::c
            lass, 'room_id');
}

public function sub_jenis()
{
    return $this-
        >belongsTo(Type::c
            lass, 'type_id');
}
```

UserController.php

```
<?php

namespace
    App\Http\Controller
        s;

use App\Models\Unit;
use App\Models\User;
use Illuminate\Http\Request;
use
    Illuminate\Support\
        Facades\Session;

class UserController extends
    Controller
{
    /**
     * Display a listing of the
     * resource.
     *
     * @return
     * \Illuminate\Http\Res
        ponse
     */
    public function index()
    {
        Session::put('index',
            request()-
```

```

        >fullUrl());

        return
        view('users.index', [
            'title' => 'Users',
            'users' =>
                User::latest()-
                >paginate(10)-
                >withQueryString(),
            'units' => Unit::all(),
        ]);
    }

    /**
     * Show the form for
     * creating a new
     * resource.
     *
     * @return
     * \Illuminate\Http\Res
     * ponse
     */
    public function create()
    {
        //
    }

    /**
     * Store a newly created
     * resource in storage.
     *
     * @param \Illuminate\
     * Http\Request $re
     * quest
     *
     * @return
     * \Illuminate\Http\Res
     * ponse
     */
    public function
    store(Request
    $request)
    {
        $rule = [
            'nama' => ['required',
            'min:3'],
            'email' => ['required',
            'email', 'min:3'],
            'unit_id' =>
            ['required'],
            'password' =>
            ['required', 'min:8']
        ];

        $validatedData =

```

```

        $request-
        >validate($rule);

        $validatedData['passwo
        rd'] =
        bcrypt($request-
        >password);

        User::create($validated
        Data);

        return
        redirect(session('ind
        ex'))-
        >with('berhasil',
        'Successfully added
        user data');
    }

    /**
     * Display the specified
     * resource.
     *
     * @param \App\Mod
     * els\User $user
     *
     * @return
     * \Illuminate\Http\Res
     * ponse
     */
    public function show(User
    $user)
    {
        //
    }

    /**
     * Show the form for
     * editing the specified
     * resource.
     *
     * @param \App\Mod
     * els\User $user
     *
     * @return
     * \Illuminate\Http\Res
     * ponse
     */
    public function edit(User
    $user)
    {
        //
    }

    /**

```

```

     * Update the specified
     * resource in storage.
     *
     * @param \Illuminate\
     * Http\Request $re
     * quest
     *
     * @param \App\Mod
     * els\User $user
     *
     * @return
     * \Illuminate\Http\Res
     * ponse
     */
    public function
    update(Request
    $request, User
    $user)
    {
        $rule = [
            'nama' => ['required',
            'min:3'],
            'email' => ['required',
            'email', 'min:3'],
            'unit_id' =>
            ['required'],
        ];

        if ($request->password)
        {
            $rule['password'] =
            ['required', 'min:8'];

            $validatedData =
            $request-
            >validate($rule);

            $validatedData['pass
            word'] =
            bcrypt($request-
            >password);
        } else {
            $validatedData =
            $request-
            >validate($rule);
        }

        User::where('id', $user-
        >id)-
        >update($validated
        Data);

        return
        redirect(session('ind
        ex'))-

```

```

        >with('berhasil',
        'Successfully
        changed user data');
    }

/**
 * Remove the specified
    resource      from
    storage.
 *
 *
 * @param \App\Mod
    els\User $user
 * @return
    \Illuminate\Http\Res
    ponse
 */
    public function
    destroy(User $user)
    {
        User::destroy($user-
        >id);

        return
        redirect(session('ind
        ex'))-
        >with('berhasil',
        'Successfully
        deleted user data');
    }
}

```

UnitController.php

```

<?php

namespace
    App\Http\Controller
    s;

use App\Models\Unit;
use Illuminate\Http\Request;
use
    Illuminate\Support\
    Facades\Session;

class UnitController extends
    Controller
{
    /**
     * Display a listing of the
        resource.
     *
     * @return
        \Illuminate\Http\Res
        ponse
    */
}

```

```

    */
    public function index()
    {
        Session::put('index',
        request()-
        >fullUrl());

        // $units = Unit::all();
        $units =
        Unit::paginate(10)-
        >withQueryString();

        return
        view('units.index', [
        'title' => 'Units',
        'units' => $units,
        ]);
    }

/**
 * Show the form for
    creating a new
    resource.
 *
 * @return
    \Illuminate\Http\Res
    ponse
 */
    public function create()
    {
        //
    }

/**
 * Store a newly created
    resource in storage.
 *
 * @param \Illuminat
    e\Http\Request $re
    quest
 * @return
    \Illuminate\Http\Res
    ponse
 */
    public function
    store(Request
    $request)
    {
        $rule = [
        'nama' => ['required',
        'unique:units'],
        'kode' => ['required',
        'unique:units'],
        ];
    }
}

```

```

        $validatedData =
        $request-
        >validate($rule);

        Unit::create($validated
        Data);

        return
        redirect(session('ind
        ex'))-
        >with('berhasil',
        'Successfully
        created unit data');
    }

/**
 * Display the specified
    resource.
 *
 * @param \App\Mod
    els\Unit $unit
 * @return
    \Illuminate\Http\Res
    ponse
 */
    public function show(Unit
    $unit)
    {
        //
    }

/**
 * Show the form for
    editing the specified
    resource.
 *
 * @param \App\Mod
    els\Unit $unit
 * @return
    \Illuminate\Http\Res
    ponse
 */
    public function edit(Unit
    $unit)
    {
        //
    }

/**
 * Update the specified
    resource in storage.
 *

```

```

        *
        @param \Illuminate\Http\Request $request
        *
        @param \App\Models\Unit $unit
        *
        @return \Illuminate\Http\Response
    */
    public function update(Request $request, Unit $unit)
    {
        $rule = [
            'nama' => ['required'],
            'kode' => ['required'],
        ];

        if ($request->nama == $unit->nama) {
            $rule['nama'] = 'required|unique:units';
        }
        if ($request->kode == $unit->kode) {
            $rule['kode'] = 'required|unique:units';
        }

        $validatedData = $request->validate($rule);

        Unit::where('id', $unit->id)->update($validatedData);

        return redirect(session('index'))->with('berhasil', 'Successfully changed unit data');
    }

    /**
     * Remove the specified resource from storage.
     *

```

```

        *
        @param \App\Models\Unit $unit
        *
        @return \Illuminate\Http\Response
    */
    public function destroy(Unit $unit)
    {
        Unit::destroy($unit->id);

        return redirect(session('index'))->with('berhasil', 'Successfully deleted unit data');
    }
}

TypeController.php
<?php
namespace App\Http\Controllers;

use App\Models\Type;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Session;

class TypeController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     *
     * @return \Illuminate\Http\Response
    */
    public function index()
    {
        Session::put('index', request()->fullUrl());

        // $types = Type::all();
        $types = Type::paginate(10)-

```

```

        >withQueryString();

        return view('types.index', [
            'title' => 'Types',
            'types' => $types,
        ]);
    }

    /**
     * Show the form for creating a new resource.
     *
     *
     * @return \Illuminate\Http\Response
    */
    public function create()
    {
        //
    }

    /**
     * Store a newly created resource in storage.
     *
     *
     * @param \Illuminate\Http\Request $request
     *
     * @return \Illuminate\Http\Response
    */
    public function store(Request $request)
    {
        $rule = [
            'nama' => ['required', 'unique:types'],
            'kode' => ['required', 'unique:types'],
        ];

        $validatedData = $request->validate($rule);

        Type::create($validatedData);

        return redirect(session('index'))->withQueryString();
    }
}

```

```

        ex'))-
        >with('berhasil',
        'Successfully
        created type data');
    }

/**
 * Display the specified
    resource.
 *
 *
 * @param \App\Mod
    els\Type $type
 * @return
    \Illuminate\Http\Res
    ponse
 */
    public function
    show(Type $type)
    {
        //
    }

/**
 * Show the form for
    editing the specified
    resource.
 *
 *
 * @param \App\Mod
    els\Type $type
 * @return
    \Illuminate\Http\Res
    ponse
 */
    public function edit(Type
    $type)
    {
        //
    }

/**
 * Update the specified
    resource in storage.
 *
 *
 * @param \Illuminat
    e\Http\Request $re
    quest
 *
 * @param \App\Mod
    els\Type $type
 * @return
    \Illuminate\Http\Res
    ponse

```

```

    */
    public function
    update(Request
    $request, Type
    $type)
    {
        $rule = [
            'nama' => ['required'],
            'kode' => ['required'],
        ];

        if ($request->nama ==
            $type->nama) {
            $rule['nama'] =
                'required|unique:typ
                es';
        }

        if ($request->kode ==
            $type->kode) {
            $rule['kode'] =
                'required|unique:typ
                es';
        }

        $validatedData =
            $request-
            >validate($rule);

        Type::where('id', $type-
            >id)-
            >update($validated
            Data);

        return
            redirect(session('ind
            ex'))-
            >with('berhasil',
            'Successfully
            changed type data');
    }

/**
 * Remove the specified
    resource from
    storage.
 *
 *
 * @param \App\Mod
    els\Type $type
 * @return
    \Illuminate\Http\Res
    ponse
 */
    public function
    destroy(Type $type)

```

```

    {
        Type::destroy($type-
            >id);

        return
            redirect(session('ind
            ex'))-
            >with('berhasil',
            'Successfully
            deleted type data');
    }
}

```

RoomController.php

```

<?php

namespace
    App\Http\Controller
    s;

use App\Models\Room;
use Illuminate\Http\Request;
use
    Illuminate\Support\
    Facades\Session;

class RoomController
    extends Controller
    {
        /**
         * Display a listing of the
            resource.
         *
         * @return
            \Illuminate\Http\Res
            ponse
         */
        public function index()
        {
            Session::put('index',
                request()-
                >fullUrl());

            // $rooms = Room::all();
            $rooms =
                Room::paginate(10)
                -
                >withQueryString();

            return
                view('rooms.index',
                [
                    'title' => 'Rooms',
                    'rooms' => $rooms,
                ]
            );
        }
    }

```

```

}

/**
 * Show the form for
 * creating a new
 * resource.
 *
 * @return
 * \Illuminate\Http\Res
 * ponse
 */
public function create()
{
    //
}

/**
 * Store a newly created
 * resource in storage.
 *
 * @param \Illuminate\
 * Http\Request $re
 * quest
 * @return
 * \Illuminate\Http\Res
 * ponse
 */
public function
store(Request
$request)
{
    $rule = [
        'nama' => ['required',
        'unique:rooms'],
        'kode' => ['required',
        'unique:rooms'],
    ];

    $validatedData =
    $request->validate($rule);

    Room::create($validate
    dData);

    return
    redirect(session('ind
    ex'))->with('berhasil',
    'Successfully
    created room data');
}

/**
 * Display the specified
 * resource.
 *
 * @param \App\Mod
 * els\Room $room
 * @return
 * \Illuminate\Http\Res
 * ponse
 */
public function
show(Room $room)
{
    //
}

/**
 * Show the form for
 * editing the specified
 * resource.
 *
 * @param \App\Mod
 * els\Room $room
 * @return
 * \Illuminate\Http\Res
 * ponse
 */
public function edit(Room
$room)
{
    //
}

/**
 * Update the specified
 * resource in storage.
 *
 * @param \Illuminate\
 * Http\Request $re
 * quest
 * @param \App\Mod
 * els\Room $room
 * @return
 * \Illuminate\Http\Res
 * ponse
 */
public function
update(Request
$request, Room
$room)
{
    $rule = [
        'nama' => ['required'],
        'kode' => ['required'],
    ];

    if ($request->nama ==
    $room->nama) {
        $rule['nama'] =
        'required|unique:roo
        ms';
    }

    if ($request->kode ==
    $room->kode) {
        $rule['kode'] =
        'required|unique:roo
        ms';
    }

    $validatedData =
    $request->validate($rule);

    Room::where('id',
    $room->id)->update($validated
    Data);

    return
    redirect(session('ind
    ex'))->with('berhasil',
    'Successfully
    changed room
    data');
}

/**
 * Remove the specified
 * resource from
 * storage.
 *
 * @param \App\Mod
 * els\Room $room
 * @return
 * \Illuminate\Http\Res
 * ponse
 */
public function
destroy(Room
$room)
{
    Room::destroy($room->id);

    return
}

```

```

        redirect(session('index'))-
        >with('berhasil',
        'Successfully
        deleted room data');
    }
}

```

InventoryController.php

```
<?php
```

```

namespace
    App\Http\Controllers;

use App\Models\Inventory;
use App\Models\Room;
use App\Models\Type;
use App\Models\Unit;
use Barryvdh\DomPDF\Facade\Pdf;
use Carbon\Carbon;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
use Illuminate\Support\Facades\Session;
use Illuminate\Support\Facades\Storage;
use Illuminate\Support\Str;

class InventoryController
    extends Controller
    {
        /**
         * Display a listing of the
         * resource.
         *
         * @return \Illuminate\Http\Response
         */
        public function
        index(Request
        $request)
        {
            Session::put('index',
            request()-
            >fullUrl());

            // ADMIN

```

```

            if (Auth::user()-
            >is_admin) {
                $search =
                Str::lower($request-
                >search);
                if ($search ===
                'good') {
                    $search = 'bagus';
                } elseif ($search ===
                'damaged') {
                    $search = 'rusak';
                }
            }

            if ($request->filter ==
            'Inventory Code') {
                $inventories =
                Inventory::latest()-
                >where('kode',
                'LIKE', '%' . $search
                . '%')->paginate(20)-
                >withQueryString();
            } elseif ($request-
            >filter == 'Name') {
                $inventories =
                Inventory::latest()-
                >where('nama',
                'LIKE', '%' . $search
                . '%')->paginate(20)-
                >withQueryString();
            } elseif ($request-
            >filter == 'Unit') {
                $inventories =
                Inventory::latest()-
                >withWhereHas('unit', function ($query)
                use ($search) {
                    $query-
                    >where('nama',
                    'LIKE', '%' . $search
                    . '%');
                }->paginate(20)-
                >withQueryString();
            } elseif ($request-
            >filter ==
            'Condition') {
                $inventories =
                Inventory::latest()-
                >where('kondisi',
                'LIKE', '%' . $search
                . '%')->paginate(20)-
                >withQueryString();
            } elseif ($request-
            >filter ==
            'Verification') {
                if ($search == 'yes')

```

```

            {
                $inventories =
                Inventory::latest()-
                >where('verifikasi',
                true)->paginate(20)-
                >withQueryString();
            } elseif ($search ==
            'no') {
                $inventories =
                Inventory::latest()-
                >where('verifikasi',
                false)-
                >paginate(20)-
                >withQueryString();
            } else {
                $inventories =
                Inventory::latest()-
                >paginate(20)-
                >withQueryString();
            }
        } elseif (!$request-
        >filter) {
            $inventories =
            Inventory::latest()-
            >paginate(20)-
            >withQueryString();
        }

        return
        view('inventories.in
        dex', [
            'title' =>
            'Inventories',
            'inventories' =>
            $inventories,
        ]);
    } else {
        $search =
        Str::lower($request-
        >search);
        if ($search ===
        'good') {
            $search = 'bagus';
        } elseif ($search ===
        'damaged') {
            $search = 'rusak';
        }
    }

    if ($request->filter ==
    'Inventory Code') {
        $inventories =
        Inventory::latest()-
        >where('unit_id',
        Auth::user()-
        >unit_id)-

```

```

>where('kode',
'LIKE', '%' . $search
. '%')->paginate(20)-
>withQueryString();
} elseif ($request-
>filter == 'Name') {
    $inventories =
Inventory::latest()-
>where('unit_id',
Auth::user()-
>unit_id)-
>where('nama',
'LIKE', '%' . $search
. '%')->paginate(20)-
>withQueryString();
} elseif ($request-
>filter == 'Unit') {
    $inventories =
Inventory::latest()-
>withWhereHas('un
it', function ($query)
use ($search) {
        $query-
>where('nama',
'LIKE', '%' . $search
. '%');
    })->paginate(20)-
>withQueryString();
} elseif ($request-
>filter ==
'Condition') {
    $inventories =
Inventory::latest()-
>where('unit_id',
Auth::user()-
>unit_id)-
>where('kondisi',
'LIKE', '%' . $search
. '%')->paginate(20)-
>withQueryString();
} elseif ($request-
>filter ==
'Verification') {
    if ($search == 'yes')
    {
        $inventories =
Inventory::latest()-
>where('unit_id',
Auth::user()-
>unit_id)-
>where('verifikasi',
true)->paginate(20)-
>withQueryString();
    } elseif ($search ==
'no') {

```

```

        $inventories =
Inventory::latest()-
>where('unit_id',
Auth::user()-
>unit_id)-
>where('verifikasi',
false)-
>paginate(20)-
>withQueryString();
    } else {
        $inventories =
Inventory::latest()-
>where('unit_id',
Auth::user()-
>unit_id)-
>paginate(20)-
>withQueryString();
    }
} elseif (!$request-
>filter) {
    $inventories =
Inventory::latest()-
>where('unit_id',
Auth::user()-
>unit_id)-
>paginate(20)-
>withQueryString();
}

return
view('inventories.in
dex', [
    'title' =>
'Inventories',
    'inventories' =>
$inventories,
]);
}

/**
 * Show the form for
creating a new
resource.
 *
 * @return
\Illuminate\Http\Res
ponse
 */
public function create()
{
    return
view('inventories.cr
eate', [
    'title' => 'Create

```

```

Inventory',
'units' => Unit::all(),
    'rooms' =>
Room::all(),
    'types' => Type::all(),
]);
}

/**
 * Store a newly created
resource in storage.
 *
 * @param \Illuminat
e\Http\Request $re
quest
 * @return
\Illuminate\Http\Res
ponse
 */
public function
store(Request
$request)
{
    // ADMIN
    if (Auth::user()-
>is_admin) {
        $rule = [
            'unit_id' =>
['required'],
            'room_id' =>
['required'],
            'type_id' =>
['required'],
            'nama' =>
['required'],
            'tahun_perolehan'
=> ['required'],
            'harga_perolehan'
=> ['required'],
            'merk' =>
['required'],
            'kondisi' =>
['required'],
            // 'foto' =>
['nullable', 'image',
'max:1024'],
        ];

        $validatedData =
$request-
>validate($rule);
        $validatedData['verif
ikasi'] = 1;

```

```
// Kondisi Harga
    if ($request-
>harga_perolehan >
15000000) {
        $kode_harga = 1;
    } elseif ($request-
>harga_perolehan
<= 15000000) {
        $kode_harga = 2;
    }
}
```

```
// CEK KODE
INVENTARIS
    $inventory =
Inventory::latest()-
>where('unit_id',
$request->unit_id)
->where('room_id',
$request->room_id)
-
>where('tahun_pero
lehan', $request-
>tahun_perolehan)
->first();
if ($inventory) {
    $kode_terakhir =
explode('/',
$inventory->kode);
```

```
// if
($kode_terakhir[4]
== $request-
>tahun_perolehan) {
    $kode_baru =
unitById($request-
>unit_id)->kode . '/'
.
roomById($request-
>room_id)->kode .
'/' . $kode_harga . '/'
typeById($request-
>type_id)->kode . '/'
.
$request-
>tahun_perolehan .
'/'
str_pad($kode_terak
hir[5] + 1, 3, '0',
STR_PAD_LEFT);
    // } else {
    // $kode_baru =
unitById($request-
>unit_id)->kode . '/'
.
roomById($request-
>room_id)->kode .
```

```
'/' . $kode_harga . '/'
typeById($request-
>type_id)->kode . '/'
.
$request-
>tahun_perolehan .
'/' . str_pad(1, 3, '0',
STR_PAD_LEFT);
    // }
} else {
    $kode_baru =
unitById($request-
>unit_id)->kode . '/'
.
roomById($request-
>room_id)->kode .
'/' . $kode_harga . '/'
typeById($request-
>type_id)->kode . '/'
.
$request-
>tahun_perolehan .
'/' . str_pad(1, 3, '0',
STR_PAD_LEFT);
}
$validatedData['kode
'] = $kode_baru;
```

```
// // CEK JIKA ADA
FOTO
    // if ($request-
>hasFile('foto')) {
    // $validatedData['
foto'] = $request-
>file('foto')-
>store('foto/');
    // }
```

```
Inventory::create($v
alidatedData);

return
redirect(session('ind
ex'))-
>with('berhasil',
'Successfully added
Inventory Data');
```

```
} else {
    $rule = [
        'room_id' =>
['required'],
        'type_id' =>
['required'],
        'nama' =>
['required'],
        'tahun_perolehan'
=> ['required'],
```

```
'harga_perolehan'
=> ['required'],
        'merk' =>
['required'],
        'kondisi' =>
['required'],
        // 'foto' =>
['nullable', 'image',
'max:2048'],
    ];
```

```
$validatedData =
$request-
>validate($rule);
$validatedData['unit
_id'] = Auth::user()-
>unit_id;
```

```
// Kondisi Harga
    if ($request-
>harga_perolehan >
15000000) {
        $kode_harga = 1;
    } elseif ($request-
>harga_perolehan
<= 15000000) {
        $kode_harga = 2;
    }
}
```

```
// CEK KODE
INVENTARIS
    $inventory =
Inventory::latest()-
>where('unit_id',
Auth::user()-
>unit_id)
->where('room_id',
$request->room_id)
-
>where('tahun_pero
lehan', $request-
>tahun_perolehan)
->first();
if ($inventory) {
    $kode_terakhir =
explode('/',
$inventory->kode);
```

```
// if
($kode_terakhir[4]
== $request-
>tahun_perolehan) {
    $kode_baru =
unitById(Auth::user
()->unit_id)->kode .
```

```

        '/' .
        roomById($request-
        >room_id)->kode .
        '/' . $kode_harga . '/' .
        typeById($request-
        >type_id)->kode . '/'
        . $request-
        >tahun_perolehan .
        '/'
        .
        str_pad($kode_terak
        hir[5] + 1, 3, '0',
        STR_PAD_LEFT);
        // } else {
        //     $kode_baru =
        unitById(Auth::user
        ()->unit_id)->kode .
        '/'
        .
        roomById($request-
        >room_id)->kode .
        '/' . $kode_harga . '/' .
        typeById($request-
        >type_id)->kode . '/'
        . $request-
        >tahun_perolehan .
        '/' . str_pad(1, 3, '0',
        STR_PAD_LEFT);
        // }
    } else {
        $kode_baru =
        unitById(Auth::user
        ()->unit_id)->kode .
        '/'
        .
        roomById($request-
        >room_id)->kode .
        '/' . $kode_harga . '/' .
        typeById($request-
        >type_id)->kode . '/'
        . $request-
        >tahun_perolehan .
        '/' . str_pad(1, 3, '0',
        STR_PAD_LEFT);
    }
    $validatedData['kode
    '] = $kode_baru;

    // // CEK JIKA ADA
    FOTO
        // if ($request-
        >hasFile('foto')) {
        //     $validatedData['
        foto'] = $request-
        >file('foto')-
        >store('foto/');
        // }
    }

    Inventory::create($v
    alidatedData);

    return
    redirect(session('ind
    ex'))-
    >with('berhasil',
    'Successfully added
    Inventory Data');
    }
}

/**
 * Display the specified
    resource.
    *
    *
    * @param \App\Mod
    els\Inventory $inve
    ntory
    * @return
    \Illuminate\Http\Res
    ponse
    */
    public function
    show(Inventory
    $inventory)
    {
        Session::put('show',
        request()-
        >fullUrl());

        return
        view('inventories.sh
        ow', [
        'title' => 'Inventory ' .
        $inventory->kode,
        'inventory' =>
        $inventory,
        ]);
    }

    /**
    * Show the form for
    editing the specified
    resource.
    *
    *
    * @param \App\Mod
    els\Inventory $inve
    ntory
    * @return
    \Illuminate\Http\Res
    ponse
    */

    public function
    edit(Inventory
    $inventory)
    {
        return
        view('inventories.ed
        it', [
        'title' => 'Edit
        Inventory ' .
        $inventory->kode,
        'inventory' =>
        $inventory,
        'units' => Unit::all(),
        'rooms' =>
        Room::all(),
        'types' => Type::all(),
        ]);
    }

    /**
    * Update the specified
    resource in storage.
    *
    *
    * @param \Illuminat
    e\Http\Request $re
    quest
    *
    * @param \App\Mod
    els\Inventory $inve
    ntory
    * @return
    \Illuminate\Http\Res
    ponse
    */
    public function
    update(Request
    $request, Inventory
    $inventory)
    {
        // ADMIN
        if (Auth::user()-
        >is_admin) {
        $rule = [
        'unit_id' =>
        ['required'],
        'room_id' =>
        ['required'],
        'type_id' =>
        ['required'],
        'nama' =>
        ['required'],
        'tahun_perolehan'
        => ['required'],
        'harga_perolehan'
    
```

```

=> ['required'],
    'merk' =>
['required'],
    'kondisi' =>
['required'],
    // 'foto' =>
['nullable', 'image',
'max:1024'],
];

```

```

    $validatedData =
$request-
>validate($rule);

```

```

// Kondisi Harga
    if ($request-
>harga_perolehan >
15000000) {
        $kode_harga = 1;
    } elseif ($request-
>harga_perolehan
<= 15000000) {
        $kode_harga = 2;
    }

```

```

    // CEK KODE
    INVENTARIS
        $item =
Inventory::latest()-
>where('unit_id',
$request->unit_id)
->where('room_id',
$request->room_id)
->where('tahun_pero
lehan', $request-
>tahun_perolehan)
->first();
if ($item) {

```

```

    $kode_terakhir =
explode('/', $item-
>kode);

```

```

    // if
($kode_terakhir[4]
== $request-
>tahun_perolehan) {
        $kode_baru =
unitById($request-
>unit_id)->kode . '/'
.
roomById($request-
>room_id)->kode .
'/' . $kode_harga . '/' .

```

```

typeById($request-
>type_id)->kode . '/'
.
$request-
>tahun_perolehan .
'/' .
str_pad($kode_terak
hir[5] + 1, 3, '0',
STR_PAD_LEFT);
    // } else {
        // $kode_baru =
unitById($request-
>unit_id)->kode . '/'
.
roomById($request-
>room_id)->kode .
'/' . $kode_harga . '/' .
typeById($request-
>type_id)->kode . '/'
.
$request-
>tahun_perolehan .
'/' . str_pad(1, 3, '0',
STR_PAD_LEFT);
    // }
} else {
    $kode_baru =
unitById($request-
>unit_id)->kode . '/'
.
roomById($request-
>room_id)->kode .
'/' . $kode_harga . '/' .
typeById($request-
>type_id)->kode . '/'
.
$request-
>tahun_perolehan .
'/' . str_pad(1, 3, '0',
STR_PAD_LEFT);
}

```

```

//
$validatedData['kod
e'] = $kode_baru;

```

```

// // CEK JIKA ADA
FOTO

```

```

    // if ($request-
>hasFile('foto')) {
        // Storage::delete(
$inventory->foto);
        // $validatedData['
foto'] = $request-
>file('foto')-
>store('foto/');
    // }

```

```

Inventory::where('id',

```

```

$inventory->id)-
>update($validated
Data);

```

```

    return
redirect(session('sho
w'))-
>with('berhasil',
'Successfully
updated Inventory
Data');
} else {

```

```

    $rule = [
        'room_id' =>
['required'],
        'type_id' =>
['required'],
        'nama' =>
['required'],
        'tahun_perolehan'
=> ['required'],
        'harga_perolehan'
=> ['required'],
        'merk' =>
['required'],
        'kondisi' =>
['required'],
        // 'foto' =>
['nullable', 'image',
'max:1024'],
];

```

```

    $validatedData =
$request-
>validate($rule);

```

```

// Kondisi Harga
    if ($request-
>harga_perolehan >
15000000) {
        $kode_harga = 1;
    } elseif ($request-
>harga_perolehan
<= 15000000) {
        $kode_harga = 2;
    }

```

```

    // CEK KODE
    INVENTARIS

```

```

        $item =
Inventory::latest()-
>where('unit_id',
Auth::user()-
>unit_id)
->where('room_id',

```

```

$request->room_id)
-
>where('tahun_pero
lehan', $request-
>tahun_perolehan)
->first();
if ($item) {
    $kode_terakhir =
explode('/', $item-
>kode);

    // if
($kode_terakhir[4]
== $request-
>tahun_perolehan) {
        $kode_baru =
unitById(Auth::user
()->unit_id)->kode .
'/' .
roomById($request-
>room_id)->kode .
'/' . $kode_harga . '/' .
typeById($request-
>type_id)->kode . '/'
. $request-
>tahun_perolehan .
'/' .
str_pad($kode_terak
hir[5] + 1, 3, '0',
STR_PAD_LEFT);
        // } else {
        // $kode_baru =
unitById(Auth::user
()->unit_id)->kode .
'/' .
roomById($request-
>room_id)->kode .
'/' . $kode_harga . '/' .
typeById($request-
>type_id)->kode . '/'
. $request-
>tahun_perolehan .
'/' . str_pad(1, 3, '0',
STR_PAD_LEFT);
        // }
    } else {
        $kode_baru =
unitById(Auth::user
()->unit_id)->kode .
'/' .
roomById($request-
>room_id)->kode .
'/' . $kode_harga . '/' .
typeById($request-
>type_id)->kode . '/'

        $request-
>tahun_perolehan .
'/' . str_pad(1, 3, '0',
STR_PAD_LEFT);
    }
}

    $request-
>tahun_perolehan .
'/' . str_pad(1, 3, '0',
STR_PAD_LEFT);
}
$validatedData['kode
'] = $kode_baru;

// // CEK JIKA ADA
FOTO
    // if ($request-
>hasFile('foto')) {
        // Storage::delete(
$inventory->foto);
        // $validatedData['
foto'] = $request-
>file('foto')-
>store('foto/');
        // }

Inventory::where('id',
$inventory->id)-
>update($validated
Data);

        return
redirect(session('sho
w'))-
>with('berhasil',
'Successfully
updated Inventory
Data');
    }
}

/**
 * Remove the specified
resource from
storage.
 *
 *
 * @param \App\Mod
els\Inventory $inve
ntory
 * @return
\Illuminate\Http\Res
ponse
 */
    public function
destroy(Inventory
$inventory)
{
    // if ($inventory->foto)
    {
        // Storage::delete($in
ventory->foto);
    }
    Inventory::destroy($inve
ntory->id);

    return
redirect(session('ind
ex'))-
>with('berhasil',
'Successfully
deleted Inventory
Data');
}

    public function
verification(Invento
ry $verification)
{
    Inventory::where('id',
$verification->id)-
>update([
'verifikasi' => true
]);

    return
redirect(session('sho
w'))-
>with('berhasil',
'Successfully
verified Inventory
Data');
}

public function print()
{
    $inventories =
Inventory::all()-
>sortBy(['room_id',
'asc'],
['tahun_perolehan',
'asc']);
    $units = Unit::all();

    $pdf =
Pdf::loadView('inve
ntories.print',
compact([
'inventories',
'units',
]));

    return $pdf-
>setOption(['enable
_php' => true])-
>setPaper('a4')-
>stream('Report.pdf'

```

```
}  
}  
);
```