

SISTEM INFORMASI INVENTORY ASET BERBASIS QR CODE (SIAQ)



TUGAS AKHIR

Diajukan untuk memenuhi salah satu syarat kelulusan pada Diploma Tiga (D3)

Disusun Oleh:

Danish Yusrah	12210025
Winno Prasetya Ramadhan	12210037
Adriyansyah Putra	12211287

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BINA SARANA INFORMATIKA
JAKARTA
2024**

Proyek Sistem Informasi :

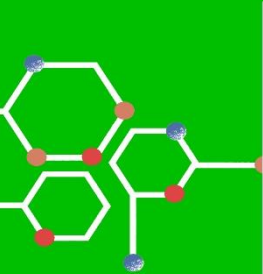
SIAQ



Sistem Inventori Aset Berbasis QR CODE

Kementrian Desa Pembangunan Daerah Tertinggal dan Transmigrasi

- Menggunakan Bahasa Pemrograman PHP
- Framework Codeigniter 3
- DBMS : MySQL



12210025

Danish Yusrah

12210037

Winno Prasetya Ramadhan

12211287

Adriyansyah Putra

Pembimbing

Nining Suharyanti, S.kom, M. Kom

Harun Al Rasyid, S.Sos, MM

KATA PENGANTAR

Alhamdulillah, puji serta syukur kami panjatkan kehadirat Allah SWT yang telah melimpahkan rahmat, taufik, hidayah serta nikmatnya sehingga penulis dapat menyelesaikan tugas ini dengan baik. Tugas Akhir ini penulis sajikan dalam bentuk buku sederhana. Dengan keterbatasan pengetahuan dan mengangkat dari beberapa sumber, penulis menyusun Buku Tutorial untuk Tugas Akhir dengan judul **“Aplikasi Sistem Inventori Aset Berbasis QRCode (SIAQ)”** sebagai salah satu persyaratan untuk kelulusan studi Diploma III pada Universitas Bina Sarana Informatika.

Sebagai bahan penulisan diambil berdasarkan hasil penelitian (eksperimen), observasi, dan beberapa sumber literatur yang mendukung penulisan ini. Penulis menyadari bahwa tanpa bimbingan dan dorongan dari semua pihak, maka penulisan buku ini tidak akan berjalan lancar. Oleh karena itu, pada kesempatan ini, ijinilah penulis menyampaikan ucapan terima kasih kepada:

1. Dr. Mochamad Wahyudi, MM, M.Kom, M.Pd, selaku Rektor Universitas Bina Sarana Informatika.
2. Dekan Fakultas Teknologi Informasi
3. Ketua Jurusan Sistem Informasi Universitas Bina Sarana Informatika.
4. Ibu Nining Surhayanti, S.Kom, M.Kom, selaku Dosen Pembimbing Tugas Akhir.
5. Bapak Harun Al Rasyid, S.Sos, MM, selaku Asisten Dosen Pembimbing Tugas Akhir.
6. Bapak Kurniawan Prambudi Utomo, SE, MM selaku Dosen Penasehat Akademik 12.6A.12 yang telah memberikan bimbingan dan pengarahan dalam penyelesaian laporan ini.
7. Seluruh staf / karyawan / dosen di lingkungan Universitas Bina Sarana Informatika.
8. Ibu Theresia Junidar, S.Pi. M.Ec selaku Kepala Pusat Data dan Informasi di Kementerian Desa, Pembangunan Daerah Tertinggal dan Transmigrasi..
9. Ibu Dian Ayu Permatasari selaku Kepala Bagian Tata Usaha Pusat Data dan Informasi di Kementerian Desa, Pembangunan Daerah Tertinggal dan Transmigrasi..
10. Seluruh staf/ pegawai di lingkungan Kementerian Desa, Pembangunan Daerah Tertinggal dan Transmigrasi.

11. Kedua orang tua yang telah memberikan dukungan baik secara moral maupun materil.
12. Teman-teman kelas 12.6A.12 di Universitas Bina Sarana Informatika yang telah membantu penulis dalam penyusunan laporan ini.
13. Serta semua pihak yang tidak dapat penulis sebutkan satu per satu yang turut membantu penulis dalam penyusunan laporan ini.

Penulis menyadari masih banyak kekurangan dalam menyusun buku ini. Untuk itu, penulis mengharapkan kritik dan saran yang membangun demi perbaikan dalam penulisan selanjutnya. Besar harapan penulis, semoga buku ini bermanfaat bagi penulis dan bagi pembaca yang berminat pada umumnya.

Jakarta, Juni 2024

Penulis

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	iii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang Masalah.....	1
1.2 Tools Pengembangan Aplikasi	3
1.3 Deskripsi Aplikasi	8
BAB II PERANCANGAN APLIKASI.....	11
2.1 Rancangan Sistem	11
2.2 Rancangan Database.....	19
2.3 Rancangan Form Master.....	28
2.4 Rancangan Form Transaksi	58
2.5 Rancangan Laporan	61
2.6 Pengujian Unit.....	65
BAB III PENUTUP	75
3.1 Kesimpulan.....	75
3.2 Saran	75
DAFTAR PUSTAKA.....	76
BIOGRAFI PENULIS	77

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

“Di era yang terus berkembang dengan cepat di bidang teknologi, penggunaan alat-alat teknologi telah menjadi keharusan dalam kegiatan sehari-hari untuk mempermudah segala aktivitas”(Kidi dalam Sama & Eric, 2023).

“Dengan adanya evolusi sistem informasi web, terjadi transformasi yang signifikan dalam pengolahan informasi dan data, sehingga memungkinkan pekerjaan dapat dilakukan dengan lebih efisien dan efektif” (Hirawan dalam Sama & Eric, 2023).

“Semakin majunya teknologi, semakin mudah pula bagi manusia untuk menyelesaikan tugas-tugas mereka dengan praktis. Teknologi informasi diartikan sebagai sebuah elemen yang memberikan kemudahan dalam berbagai aspek kehidupan manusia”(Sefullah dalam Sama & Eric, 2023).

“Salah satu contohnya adalah sistem informasi yang merupakan proses penting dalam pengolahan data, mulai dari pengumpulan, pengolahan, analisis, penyimpanan, hingga pengembangan untuk kepentingan individu maupun organisasi demi kemajuan yang lebih baik”(Irwanto dalam Sama & Eric, 2023).

Inventori adalah konsep yang mencerminkan sumber daya yang tersedia namun belum digunakan. Inventori bisa diartikan sebagai stock yang ada saat itu, daftar barang yang tersedia, atau jumlah stock barang yang dimiliki suatu organisasi. Fungsi utama inventori adalah memenuhi permintaan pelanggan dengan stok barang yang minimal. (Rahman dalam Limbong & Sianipar, 2022).

Aset adalah sesuatu yang memiliki nilai tukar atau kekayaan, jadi dari definisi di atas, aset adalah bentuk penanaman modal perusahaan. Aset dapat berupa harta kekayaan atau jasa yang dimiliki oleh perusahaan. Kekayaan harus dinyatakan dengan jelas, diukur dalam satuan uang, dan diurutkan berdasarkan seberapa lama atau cepat mereka berubah menjadi uang. (Roni dalam Limbong & Sianipar, 2022).

Inventori aset adalah salah satu dari beberapa faktor penting pada sebuah perusahaan atau suatu instansi untuk mendukung keseimbangan proses inventarisasi dalam sebuah perusahaan untuk mencapai tujuan perusahaan atau instansi.

Kementerian Desa, Pembangunan Daerah Tertinggal, dan Transmigrasi (Kemendesa PDTT) Republik Indonesia adalah instansi pemerintahan yang bertanggung jawab atas pembangunan desa dan kawasan perdesaan, pemberdayaan masyarakat desa, percepatan pembangunan daerah-daerah tertinggal, serta transmigrasi. Kementerian ini berada di bawah kendali dan tanggung jawab Presiden. Sejak 27 Oktober 2014, Kemendesa PDTT ini dipimpin oleh seorang Menteri bernama Bapak Marwan Ja'far. Kemudian pada tahun 2016 dialihkan kepada Bapak Eko Putro Sandjojo.

Melalui Keputusan Presiden Republik Indonesia Nomor 113/P Tahun 2019 Tentang Pembentukan Kementerian Negara dan Pengangkatan Menteri Negara Kabinet Indonesia Maju Periode Tahun 2019-2024, Kemendesa PDTT dipimpin oleh Bapak Abdul Halim Iskandar.

Di Kantor Kemendesa PDTT, sistem pencatatan inventori aset masih dilakukan secara manual. Proses pencarian data barang dan laporan sangat lambat karena harus dilakukan secara manual, menghabiskan waktu hingga satu hari untuk menyusun laporan inventori. Saat melakukan pengecekan, petugas harus datang langsung ke lokasi untuk mencocokkan dan mencatat data barang secara manual, yang membutuhkan waktu ekstra karena dilakukan dua kali.

Untuk meningkatkan efisiensi, Kantor Kemendesa PDTT sedang mengembangkan sistem inventori berbasis web dengan metode *Rapid Application Development* (RAD). Proses ini meliputi tahapan perencanaan, analisis, perencanaan kebutuhan, desain sistem, dan implementasi. Harapannya, sistem baru ini akan mempermudah divisi PUSDATIN dalam manajemen data inventori, khususnya dalam infrastruktur jaringan. Dengan sistem Informasi Inventori aset barang berbasis web dengan QRCode, Kantor Kemendesa PDTT dapat mengelola,

mencari, dan membuat laporan data inventori dengan cepat serta membantu petugas dalam penginputan data barang secara efisien.

Berdasarkan permasalahan yang telah disebutkan maka, dibuatkan suatu sistem informasi yang implementasinya bisa membantu penginputan, pengelolaan dan pelaporan data inventori aset yang terstruktur. Sehingga dapat mempermudah akses para staff dan admin pada divisi PUSDATIN pada bagian infrastruktur jaringan di Kemendesa PDTT dalam pendataan dan pengelolaan inventori aset serta pembuatan laporan dengan waktu yang lebih cepat mencakup wilayah Kemendesa PDTT yang cukup luas.

1.2 Tools Pengembangan Aplikasi

Dalam pengembangan website dengan framework CodeIgniter, kita akan memanfaatkan sejumlah tools kunci yang esensial untuk menciptakan aplikasi web yang kuat dan efisien. Tools tersebut termasuk CodeIgniter Framework sebagai fondasi utama, text editor atau IDE untuk penulisan kode yang efisien, web server untuk menjalankan aplikasi lokal, database management system (DBMS) untuk manajemen data, dan beragam alat pendukung lainnya seperti Composer, terminal atau CLI. Dengan bantuan kombinasi tools ini, kita dapat membuat suatu website yang handal dan sesuai dengan kebutuhan pengguna.

1.2.1 CodeIgniter Framework

“CodeIgniter merupakan aplikasi open source berupa framework PHP untuk membuat website lebih dinamis menggunakan PHP dengan pola MVC (Model, View, Controller)” (Praba dalam Ikhsan et al., 2024).

CodeIgniter memudahkan pengembang web untuk membuat dan membangun aplikasi web dengan cepat dari awal. Selain meningkatkan dinamika web, proses ini membantu pengembang dalam membuat aplikasi web yang ringan dan cepat. CodeIgniter memiliki dokumentasi yang lengkap dengan contoh implementasi kode yang detail (Sahi dalam Ikhsan et al., 2024).

Dokumentasi yang lengkap dan detail ini menjadi salah satu alasan kuat sehingga banyak orang yang lebih memilih CodeIgniter sebagai pilihannya. Terlepas dari kelebihan CodeIgniter, banyak orang yang belum menggunakannya karena mereka tidak cukup tahu tentang framework Codeigniter.

1.2.2 Text Editor (IDE)

Dalam dunia pengembangan perangkat lunak, text editor IDE menjadi sebuah perangkat yang tak tergantikan. Dengan menggabungkan fungsi dari text editor dan *Integrated Development Environment* (IDE), text editor IDE memberikan lingkungan yang lengkap bagi para pengembang untuk menulis, mengedit, dan mengelola kode secara efisien. Fitur-fitur seperti highlighting sintaksis, auto-completion, dan indentasi otomatis mempermudah dalam menulis kode yang bersih dan terstruktur. Selain itu, text editor IDE juga dilengkapi dengan kemampuan untuk mengintegrasikan alat-alat pengembangan lainnya seperti debugger, compiler, dan *version control system* (VCS) seperti Git. Hal ini memungkinkan pengembang untuk mengelola proyek mereka dengan lebih baik, melacak perubahan kode, dan berkolaborasi dengan tim dengan lebih efektif. Sebagai hasilnya, text editor IDE tidak hanya meningkatkan produktivitas, tetapi juga membantu meningkatkan kualitas dan kolaborasi dalam proses pengembangan perangkat lunak.

1.2.3 Web Server

Sebuah web server adalah perangkat lunak atau perangkat keras sebagai penyedia layanan untuk meng-host halaman web dan menyampaikan konten kepada pengguna melalui Internet.

Sistem komputer yang menyediakan layanan tertentu dalam jaringan disebut server. Dalam beberapa kasus, server juga disebut sebagai web server. Menjalankan perangkat lunak administratif, yang mengatur akses ke sumber daya jaringan seperti file dan printer, dan memberikan akses ke workstation

jaringan, adalah tugas utama server. Berbagai layanan biasanya disediakan oleh sistem operasi server yang menggunakan arsitektur klien/server. Contoh layanan server termasuk Mail Server, DHCP Server, HTTP Server, DNS Server, FTP Server, dan lainnya. Setiap sistem operasi server biasanya menyusun berbagai layanan ini, baik secara internal maupun dengan mendapatkan layanan dari pihak ketiga. Setiap layanan menangani permintaan klien (Prakoso, 2018).

1.2.4 Database Management System (DBMS)

Sebuah Database Management System (DBMS) adalah "sistem yang dirancang untuk mengelola basis data, menyediakan fungsionalitas untuk membuat, mengakses, mengelola, mencari, dan mengamankan data dalam sebuah database." DBMS memungkinkan pengguna untuk menyimpan, mengelola, dan mengakses data dengan efisien, serta menyediakan alat-alat untuk menjalankan query, mengatur keamanan data, dan melakukan tugas-tugas administratif terkait database. Dengan menggunakan DBMS, organisasi dapat mengoptimalkan pengelolaan data mereka, meningkatkan kinerja, dan mempertahankan konsistensi data dalam berbagai aplikasi dan sistem.

Menurut (Andaru dalam Farahat et al., 2022) basis data atau *database* adalah kumpulan informasi yang tersimpan secara terstruktur dalam komputer, memungkinkan penggunaan program komputer untuk memeriksa dan mengambil informasi dari basis data tersebut. Sistem perangkat lunak yang digunakan untuk mengatur dan mengeksekusi kueri dalam basis data disebut sebagai sistem manajemen basis data (Database Management System, DBMS). Ilmu informasi mempelajari sistem basis data ini secara mendalam.

1.2.5 Composer (Optional)

Composer adalah "manajer paket untuk PHP yang memungkinkan pengembang untuk mengelola dependensi kode secara efisien dalam proyek PHP mereka." Dengan Composer, pengembang dapat mendefinisikan paket-paket eksternal yang dibutuhkan oleh proyek PHP mereka, seperti framework atau library khusus, dan Composer akan mengelola instalasi dan pembaruan paket-paket tersebut secara otomatis.

Menurut (Febian dalam Tanari et al., 2019) composer adalah manajer dependensi yang dirancang khusus untuk PHP, mirip dengan Gem (Ruby) atau Maven (Java). Dengan Composer, Anda dapat menginstal dan memperbarui perpustakaan (library) PHP secara otomatis sesuai kebutuhan proyek Anda.

Menurut (Luthfi dalam Tanari et al., 2019), dependency manager dalam Composer mengacu pada fungsi Composer untuk mengatur hubungan "ketergantungan" antara berbagai perpustakaan (package) PHP yang dibuat oleh pengembang untuk digunakan oleh orang lain. Composer memastikan bahwa semua ketergantungan ini dikelola dengan baik dan dapat diintegrasikan ke dalam proyek PHP dengan mudah.

Penggunaan Composer mempermudah pengembangan PHP karena pengembang tidak perlu lagi mengunduh dan mengelola dependensi secara manual. Composer akan menangani semua proses instalasi, memastikan versi yang cocok, dan memperbarui paket-paket jika diperlukan. Ini membantu mengurangi kesalahan dan mengoptimalkan pengelolaan dependensi dalam proyek-proyek PHP yang kompleks. Selain itu, Composer juga memungkinkan pengguna untuk membuat dan mengelola proyek-proyek PHP dengan struktur yang terorganisir dengan baik, memisahkan kode menjadi berbagai modul atau paket-paket yang independen. Hal ini mempromosikan praktik-praktik pengembangan yang bersih dan modular, serta memudahkan penggunaan kode yang dapat digunakan ulang dalam proyek-proyek berbeda. Secara keseluruhan, Composer merupakan alat yang sangat berguna bagi pengembang PHP karena membantu dalam manajemen dependensi, struktur proyek yang baik, dan penggunaan kode yang bersih dan modular dalam proyek-proyek PHP modern.

1.2.6 Terminal atau Command Line Interface

Terminal atau *Command Line Interface* (CLI) adalah antarmuka pengguna yang memungkinkan pengguna untuk berinteraksi dengan sistem operasi atau program menggunakan baris perintah teks. Dalam terminal atau CLI, pengguna dapat memasukkan perintah-perintah teks langsung ke dalam sistem untuk menjalankan berbagai tugas, seperti mengelola file, menjalankan program, mengkonfigurasi sistem, dan

lain sebagainya. Penggunaan terminal atau CLI memerlukan pengetahuan tentang sintaks perintah dan cara kerja sistem operasi atau program yang sedang digunakan. Meskipun membutuhkan pembelajaran, penggunaan terminal atau CLI memberikan kontrol yang lebih langsung terhadap operasi-operasi yang dilakukan dalam sistem atau program, dan sering digunakan oleh para profesional seperti pengembang perangkat lunak dan sistem administrator dalam pekerjaan sehari-hari.

1.2.7 Browser

Browser, juga dikenal sebagai web browser atau peramban web, adalah perangkat lunak yang memungkinkan pengguna untuk menjelajahi dan mengakses berbagai halaman web serta konten-konten online lainnya. Browser bertindak sebagai antarmuka antara pengguna dan World Wide Web (WWW), menginterpretasikan kode-kode HTML, CSS, dan JavaScript dari situs web untuk menampilkan halaman web secara visual kepada pengguna.

Search engine (mesin pencari) adalah perangkat lunak yang dirancang untuk menemukan berbagai informasi yang tersimpan dalam layanan seperti World Wide Web (WWW), File Transfer Protocol (FTP), Mailing List, atau News Group yang terdapat di berbagai server dalam suatu jaringan tertentu. Hasil dari pencarian akan menampilkan berbagai informasi yang berasal dari website, blog, forum, dan sumber informasi lainnya. Hasil pencarian ini sering kali disebut sebagai Halaman Hasil Pencarian (Search Engine Result Pages, SERPs). (Syahputra & Kom, 2018).

1. Fungsi utama browser meliputi:

Menampilkan Halaman Web: Browser mengambil konten dari server web dan menampilkan halaman web dalam bentuk yang dapat dibaca oleh pengguna, termasuk teks, gambar, video, dan elemen-elemen interaktif lainnya.

2. Navigasi Web: Browser memungkinkan pengguna untuk menavigasi antara berbagai halaman web dengan menggunakan tombol kembali, maju, dan tombol navigasi lainnya. Pengguna juga

dapat memasukkan URL langsung ke dalam bilah alamat untuk mengakses situs web tertentu.

3. **Rendering dan Interaksi:** Browser bertanggung jawab untuk merender halaman web dengan benar, mengikuti standar web seperti HTML, CSS, dan JavaScript. Selain itu, browser juga memungkinkan interaksi pengguna seperti mengklik tautan, mengisi formulir, dan menjalankan skrip JavaScript.
4. **Penyimpanan dan Manajemen Data:** Browser menyimpan data pengguna seperti riwayat penelusuran, cookie, dan cache untuk memfasilitasi pengalaman pengguna yang lebih personal dan efisien.

1.3 Deskripsi Aplikasi

Dengan mengikuti perkembangan teknologi yang terus maju, kami hadirkan sebuah platform yang menggabungkan kecanggihan sistem QR code dengan kehandalan manajemen inventori barang. Website ini dirancang untuk memberikan solusi terbaik dalam pendataan dan pengelolaan aset, serta pelacakan lokasi dengan akurat, sehingga memungkinkan pengguna untuk mengoptimalkan pengelolaan inventori barang dengan lebih efisien dan efektif. Dengan fitur-fitur yang lengkap dan user-friendly, pengguna dapat dengan mudah melakukan registrasi barang, membuat QR code yang unik untuk setiap item, dan memantau lokasi barang secara real-time. Tak hanya itu, sistem juga menyediakan berbagai laporan dan analisis yang dapat membantu untuk pengambilan keputusan yang lebih tepat dan strategis terkait inventarisasi barang. Dibangun dengan keamanan data yang tinggi dan desain yang responsif, website ini memberikan pengalaman pengguna yang optimal dari berbagai perangkat, menjadikannya solusi terbaik untuk kebutuhan manajemen inventori barang pada berbagai skala dan jenis organisasi.

Deskripsi website inventori aset berbasis QR code akan meliputi beberapa elemen kunci:

Tujuan Utama:

Website ini didesain untuk melakukan pendataan dan pelacakan barang menggunakan QR code. QR code ditempel pada setiap barang untuk memudahkan pengguna dalam mengetahui lokasi keberadaan barang tersebut.

1. Fungsionalitas Utama:

- A. Pendataan Barang: Pengguna dapat mendaftarkan barang-barang dengan menginput informasi terkait, seperti nama barang, deskripsi, kondisi, dan lokasi awal.
- B. Pembuatan QR Code: Setelah data barang dimasukkan, sistem akan membuat QR code unik untuk setiap barang.
- C. Pelacakan Lokasi: Saat barang dipindahkan atau dipergunakan di lokasi lain, pengguna dapat memindai QR code menggunakan aplikasi khusus untuk memperbarui lokasi keberadaan barang tersebut.
- D. Pencarian dan Laporan: Pengguna dapat mencari barang berdasarkan nama, jenis barang, atau lokasi. Selain itu, sistem juga menyediakan fitur untuk menghasilkan laporan tentang inventori barang berdasarkan kriteria tertentu.

2. Keamanan dan Aksesibilitas:

- A. Otorisasi Pengguna: Website ini memiliki sistem otorisasi yang memungkinkan pengguna untuk memiliki hak akses yang sesuai dengan peran dan tanggung jawab mereka, seperti administrator dan staff.
- B. Keamanan Data: Data inventori dan lokasi barang disimpan secara aman dengan enkripsi dan sistem backup yang teratur untuk mencegah kehilangan data.

3. Antarmuka Pengguna yang Ramah:

- A. Desain Intuitif: Antarmuka website didesain agar mudah dipahami dan digunakan oleh pengguna dengan berbagai tingkat kemampuan teknologi.
- B. Responsif: Website ini dapat diakses dari berbagai perangkat, seperti komputer, tablet, atau smartphone, dengan tampilan yang responsif dan optimal.

- C. Dengan menggunakan website inventori aset berbasis QR code, proses pendataan, pelacakan, dan manajemen inventori barang akan menjadi lebih efisien dan terorganisir. Sistem QR code juga memudahkan pengguna dalam mengetahui lokasi tepat dari setiap barang secara real-time.

BAB II

PERANCANGAN APLIKASI

2.1 Rancangan Sistem

Perancangan sistem adalah proses untuk membangun sebuah sistem yang memenuhi spesifikasi kebutuhan fungsional dengan memperhatikan performa dan penggunaan sumber daya. Kepuasan dari segi biaya, waktu, dan perangkat merupakan bagian penting dari perancangan sistem ini (Rosa dan Salahuddin dalam Istiawan & Nuralia, 2021).

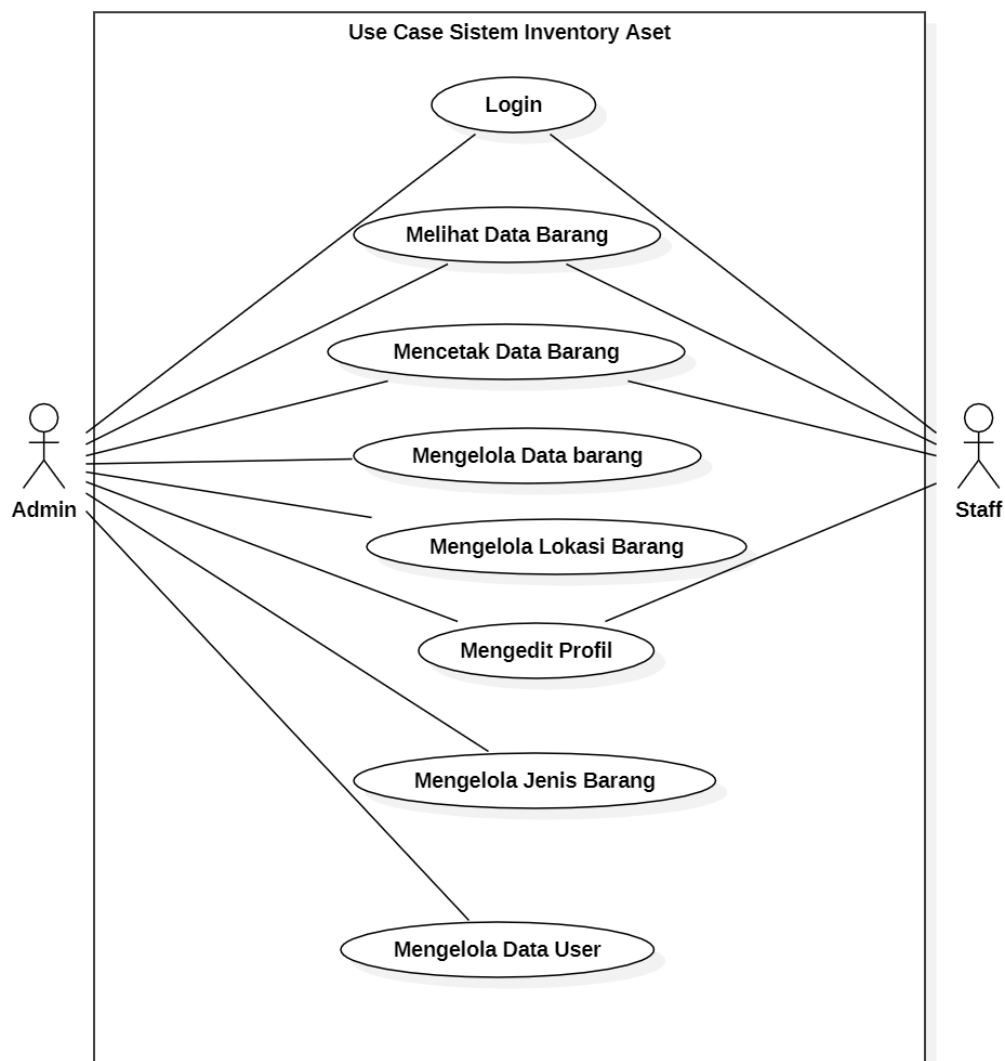
“Perancangan sistem adalah proses menggambar, merencanakan, dan menyusun elemen-elemen terpisah menjadi sebuah kesatuan utuh yang dapat dianggap sebagai perancangan sistem” (Jogiyanto dalam Istiawan & Nuralia, 2021).

UML (*Unified Modeling Language*) sangat dibutuhkan untuk perancangan suatu sistem aplikasi berbasis web. UML (Unified Modeling Language) adalah sebuah bahasa diagram yang digunakan untuk menggambarkan, menspesifikasi, membangun, dan mendokumentasikan sistem pengembangan perangkat lunak berbasis OOP (Object Oriented Programming). Selain sebagai alat visualisasi, UML juga dapat terhubung langsung dengan berbagai bahasa pemrograman. Unified Modeling Language (UML) digunakan sebagai standar untuk membuat *blueprint* perangkat lunak. (LAUNDRY, n.d.)

A. Use Case Diagram

Use case adalah langkah awal dalam membentuk suatu sistem. *Use case* adalah representasi dari kebutuhan fungsional sistem, di mana setiap *use case* menggambarkan skenario yang dilakukan oleh aktor dalam batasan sistem tertentu. Setiap *use case* dihubungkan dengan notasi garis untuk menggambarkan hubungan dan interaksi antara mereka. (Aliman, 2021).

Sebuah *use case* mempresentasikan sebuah interaksi antara sistem dengan actor-aktor yang terkait. Pada sistem inventori aset ini terdapat 2 aktor yakni admin dan staff. Dari dua aktor tersebut masing – masing memiliki deskripsinya sendiri.



Gambar II. 1 *Use Case Diagram* Inventori Aset

Table II. 1
Deskripsi *Use Case Login*

<i>Use Case Name</i>	<i>Login</i>
<i>Requirements</i>	A1
<i>Goal</i>	Admin dan staff dapat login dan masuk ke dashboard masing-masing.
<i>Pre-Conditions</i>	Admin dan staff memasukkan username dan password
<i>Post-Conditions</i>	Sistem menampilkan halaman masing-masing user
<i>Failed end Condition</i>	<i>Username</i> atau <i>password</i> salah
<i>Actor</i>	Admin dan Staff
<i>Main Flow / Basic Path</i>	1. <i>User</i> memasukkan <i>username</i> dan <i>password</i> 2. Sistem <i>username</i> dan <i>password</i> 3. Sistem menampilkan halaman dashboard
<i>Invariant A</i>	1. Sistem memvalidasi <i>username</i> dan <i>password</i> 2. Sistem menampilkan pesan <i>username</i> dan <i>password</i> salah

Table II. 2
Deskripsi *Use Case Melihat Data Barang*

<i>Use Case Name</i>	Melihat Data Barang
<i>Requirements</i>	A2
<i>Goal</i>	<i>User</i> dapat melihat data barang
<i>Pre-Conditions</i>	<i>User</i> telah <i>login</i>
<i>Post-Conditions</i>	Sistem menampilkan <i>page</i> data barang
<i>Failed end Condition</i>	Sistem tidak menampilkan data barang
<i>Actor</i>	Admin dan Staff
<i>Main Flow / Basic Path</i>	1. <i>User</i> melakukan login 2. <i>User</i> memilih menu Data Barang 3. Sistem menampilkan Data Barang
<i>Invariant A</i>	1. Sistem menampilkan Data Barang 2. Sistem menampilkan tombol download Excel

Table II. 3
Deskripsi *Use Case* Mencetak Data Barang

<i>Use Case Name</i>	Mencetak Data Barang
<i>Requirements</i>	A3
<i>Goal</i>	User dapat mencetak data barang menjadi file excel
<i>Pre-Conditions</i>	User telah login
<i>Post-Conditions</i>	Sistem mendownload data barang menjadi excel
<i>Failed end Condition</i>	Data Barang tidak dapat didownload
<i>Actor</i>	Admin dan Staff
<i>Main Flow / Basic Path</i>	1. User melakukan login 2. User memilih menu Data Barang 3. Sistem menampilkan Data Barang 4. User menekan tombol Download Excel 5. Sistem men-download Data Barang
<i>Invariant A</i>	1. User melakukan login 2. User memilih menu Data Barang 3. Sistem menampilkan Data Barang 4. User menekan tombol Download Excel 5. Sistem gagal men-download Data Barang

Table II. 4
Deskripsi *Use Case* Mengelola Data Barang

<i>Use Case Name</i>	Mengelola Data Barang
<i>Requirements</i>	A4
<i>Goal</i>	Admin dapat mengelola data barang yang telah di-input dan barang masuk
<i>Pre-Conditions</i>	Admin telah melakukan login
<i>Post-Conditions</i>	Sistem menyimpan perubahan data barang
<i>Failed end Condition</i>	Sistem gagal menyimpan data barang masuk dan menampilkan pesan error.
<i>Actor</i>	Admin
<i>Main Flow / Basic Path</i>	1. Admin telah melakukan login 2. Admin memilih menu Data Barang 3. Admin memilih tambah data 4. Sistem menampilkan form tambah barang 5. Admin mengisi form tambah barang 6. Sistem menyimpan perubahan data barang baru.
<i>Invariant A</i>	1. Admin telah melakukan login 2. Admin memilih menu Data Barang 3. Admin memilih tombol ubah 4. Sistem menampilkan form ubah barang 5. Admin mengisi form ubah barang 6. Sistem meng-update data barang
<i>Invariant B</i>	1. Admin telah melakukan login 2. Admin memilih menu Data Barang 3. Admin memilih tombol hapus 4. Sistem menghapus data barang

Table II. 5
Deskripsi *Use Case* Mengelola Lokasi Barang

<i>Use Case Name</i>	Mengelola Lokasi Barang
<i>Requirements</i>	A5
<i>Goal</i>	Admin dapat mengelola lokasi barang yang telah di- <i>input</i> dan lokasi barang baru
<i>Pre-Conditions</i>	Admin telah melakukan <i>login</i>
<i>Post-Conditions</i>	Sistem menyimpan perubahan lokasi barang
<i>Failed end Condition</i>	Sistem gagal menyimpan lokasi barang baru
<i>Actor</i>	Admin
<i>Main Flow / Basic Path</i>	1. Admin telah melakukan login 2. Admin memilih menu Lokasi Barang 3. Admin memilih tambah data 4. Sistem menampilkan form tambah lokasi 5. Admin mengisi form tambah lokasi 6. Sistem menyimpan perubahan lokasi barang baru.
<i>Invariant A</i>	1. Admin telah melakukan login 2. Admin memilih menu Lokasi Barang 3. Admin memilih tombol ubah 4. Sistem menampilkan form ubah lokasi 5. Admin mengisi form ubah lokasi 6. Sistem meng-<i>update</i> lokasi barang
<i>Invariant B</i>	1. Admin telah melakukan login 2. Admin memilih menu Lokasi Barang 3. Admin memilih tombol hapus 4. Sistem menghapus lokasi barang

Table II. 6
Deskripsi *Use Case* Mengelola Jenis Barang

<i>Use Case Name</i>	Mengelola Jenis Barang
<i>Requirements</i>	A6
<i>Goal</i>	Admin dapat mengelola jenis barang yang telah di- <i>input</i> dan jenis barang baru
<i>Pre-Conditions</i>	Admin telah melakukan <i>login</i>
<i>Post-Conditions</i>	Sistem menyimpan perubahan jenis barang
<i>Failed end Condition</i>	Sistem gagal menyimpan jenis barang baru
<i>Actor</i>	Admin
<i>Main Flow / Basic Path</i>	1. Admin telah melakukan login 2. Admin memilih menu Jenis Barang 3. Admin memilih tambah data 4. Sistem menampilkan form tambah jenis 5. Admin mengisi form tambah jenis 6. Sistem menyimpan perubahan jenis barang baru.
<i>Invariant A</i>	1. Admin telah melakukan login 2. Admin memilih menu Jenis Barang 3. Admin memilih tombol ubah 4. Sistem menampilkan form ubah jenis 5. Admin mengisi form ubah jenis 6. Sistem meng-<i>update</i> jenis barang
<i>Invariant B</i>	1. Admin telah melakukan login 2. Admin memilih menu Jenis Barang 3. Admin memilih tombol hapus 4. Sistem menghapus jenis barang

Table II. 7
Deskripsi Use Case Mengedit Profil

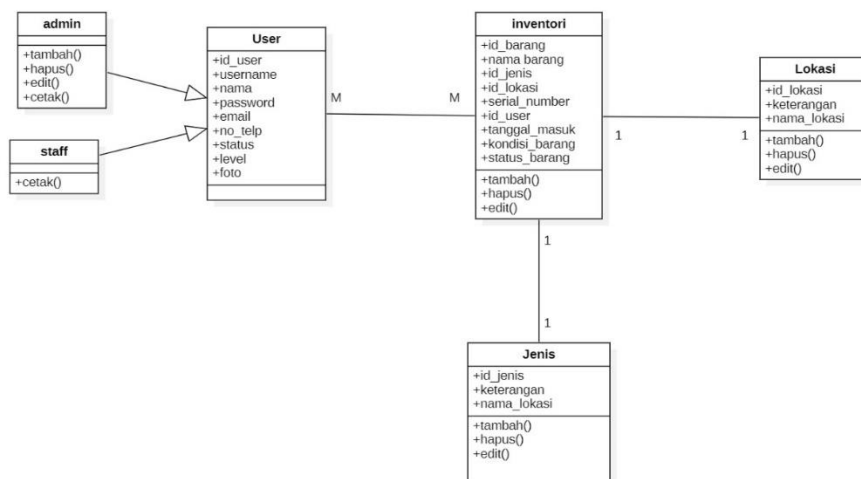
<i>Use Case Name</i>	Meng-edit Profil
<i>Requirements</i>	A7
<i>Goal</i>	User dapat mengubah profil
<i>Pre-Conditions</i>	User telah login
<i>Post-Conditions</i>	Sistem menyimpan perubahan profil user
<i>Failed end Condition</i>	Sistem gagal menyimpan perubahan profil, dan menampilkan pesan eror
<i>Actor</i>	Admin dan Staff
<i>Main Flow / Basic Path</i>	1. User melakukan login 2. User masuk ke bagian profil 3. User memilih ubah profil 4. Sistem menampilkan form ubah profil 5. User mengubah data form ubah profil 6. Sistem berhasil menyimpan perubahan profil
<i>Invariant A</i>	1. User melakukan login 2. User masuk ke bagian profil 3. User memilih ubah profil 4. Sistem menampilkan form ubah profil 5. User mengubah data form ubah profil 6. Sistem gagal menyimpan perubahan profil

Table II. 8
Deskripsi *Use Case* Mengelola Data User

<i>Use Case Name</i>	Mengubah Data User
<i>Requirements</i>	A8
<i>Goal</i>	Admin dapat mengubah profil
<i>Pre-Conditions</i>	Admin telah login
<i>Post-Conditions</i>	Sistem menyimpan perubahan data <i>user</i>
<i>Failed end Condition</i>	Sistem gagal menyimpan perubahan data <i>user</i> , dan menampilkan pesan eror
<i>Actor</i>	Admin
<i>Main Flow / Basic Path</i>	1. Admin telah melakukan login 2. Admin memilih menu Data User 3. Admin memilih tambah data 4. Sistem menampilkan form tambah pengguna 5. Admin mengisi form tambah pengguna 6. Sistem menyimpan perubahan data pengguna
<i>Invariant A</i>	1. Admin telah melakukan login 2. Admin memilih menu Data User 3. Admin memilih tombol ubah 4. Sistem menampilkan form ubah jenis 5. Admin mengisi form ubah data pengguna 6. Sistem meng-<i>update</i> data pengguna
<i>Invariant B</i>	1. Admin telah melakukan login 2. Admin memilih menu Data <i>User</i> 3. Admin memilih tombol hapus 4. Sistem menghapus data <i>user</i>

B. Class Diagram

“*Class diagram* (diagram kelas) adalah gambaran grafis yang mengilustrasikan hubungan antara kelas-kelas, yang mencakup atribut dan fungsi dari objek tersebut.” (The Elements of UML 2.0 : Scott W.Ambler, 2005:47 Arianti et al., 2022).



Gambar II. 2 *Class Diagram* Inventori Aset

2.2 Rancangan Database

Perancangan *database* adalah proses untuk menentukan konten dan struktur data yang diperlukan untuk mendukung berbagai perancangan sistem. Perancangan database ini difokuskan pada pembuatan ERD dan spesifikasi file yang diperlukan dalam sistem inventori aset ini.

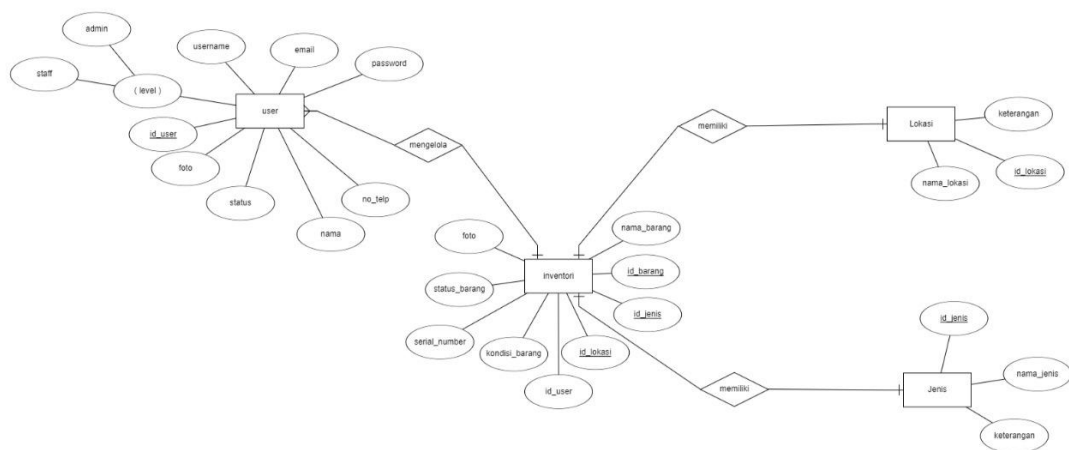
A. *Entity Relationship Diagram* (ERD)

“*Entity Relationship Diagram* (ERD) adalah sebuah model yang digunakan untuk menggambarkan hubungan antar data dalam sebuah basis data, berdasarkan objek-objek data dasar yang memiliki hubungan melalui relasi.”(Wei et al. dalam Indarta et al., 2021).

“ERD digunakan untuk memodelkan struktur data dan hubungan antar data dengan menggunakan berbagai notasi dan simbol yang tepat.”(Liu et al. dalam Indarta et al., 2021)

“Perancangan database bertujuan untuk menghindari redudansi dan duplikasi data agar sistem dapat menghasilkan informasi yang efektif. Proses normalisasi dan hubungan antar tabel dalam ERD diperlukan untuk mencapai tujuan tersebut.” (Effendy dalam Indarta et al., 2021)

Proses tersebut yang terdapat pada sistem informasi manajemen aset ini adalah sebagai berikut :



Gambar II. 3 Entity Relationship Diagram Inventori Aset

B. Spesifikasi File

Entitas - entitas yang berada didalam rancangan ERD diuraikan lebih rinci dengan menggunakan spesifikasi file. Adapun spesifikasi file yang diuraikan terdiri dari:

1. Spesifikasi File User

Nama File	: User
Akronim	: User
Fungsi	: untuk menyimpan data user
Tipe File	: File Master
Organisasi File	: <i>Indexed Sequential</i>
Akses File	: Random
Media	: Hard Disk
Panjang Record	: 570

Kunci Field : id_user
Software : Mysql

Tabel II. 9 Spesifikasi File User

No	Nama Field	Tipe	Panjang	Ket
1	id_user	varchar	50	Primary Key
2	nama	varchar	50	
3	username	varchar	50	
4	email	varchar	100	
5	notelp	varchar	15	
6	level	enum	-	
7	password	varchar	255	
8	foto	varchar	50	
9	status	enum	-	

2. Spesifikasi File Inventori

Nama File : Inventori
Akronim : Inventori
Fungsi : untuk menyimpan data barang
Tipe File : File Transaksi
Organisasi File : *Indexed Sequential*
Akses File : Random
Media : Hard Disk
Panjang Record : 1110
Kunci Field : id_barang
Software : Mysql

Tabel II. 10 Spesifikasi File Inventori

No	Nama Field	Tipe	Panjang	Ket
1	id_barang	varchar	20	Primary Key
2	nama_barang	varchar	60	
3	id_lokasi	int	20	Foreign Key
4	id_jenis	int	20	Foreign Key
5	foto	varchar	225	
6	serial_number	varchar	255	
7	tanggal_masuk	varchar	255	
8	kondisi_barang	varchar	255	
9	status_barang	enum	-	'Terpakai', 'Tidak Terpakai'.

3. Spesifikasi File Jenis

Nama File : Jenis
 Akronim : Jenis
 Fungsi : untuk menyimpan data jenis barang
 Tipe File : File Master
 Organisasi File : *Indexed Sequential*
 Akses File : Random
 Media : Hard Disk
 Panjang Record : 40
 Kunci Field : id_jenis
 Software : Mysql

Tabel II. 11 Spesifikasi File Jenis

No	Nama Field	Tipe	Panjang	Ket
1	id_jenis	int	20	Primary Key
2	nama_jenis	varchar	20	
3	ket	text	-	

4. Spesifikasi File Lokasi

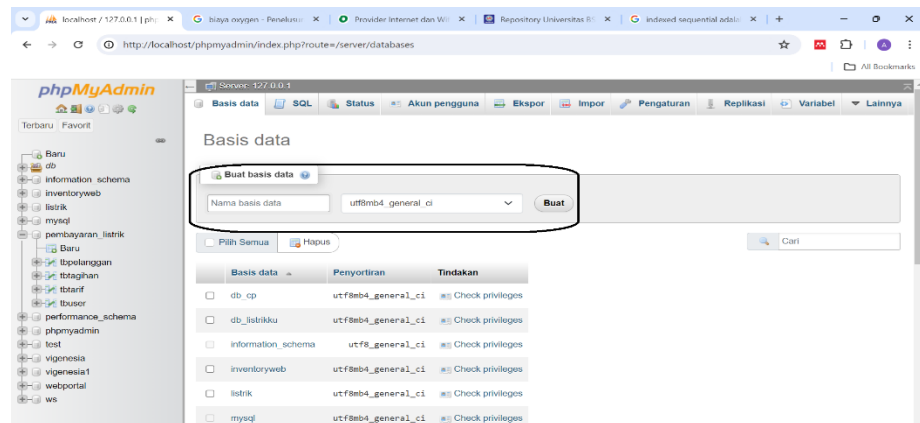
Nama File : lokasi
 Akronim : lokasi
 Fungsi : untuk menyimpan data lokasi barang
 Tipe File : File Master
 Organisasi File : *Indexed Sequential*
 Akses File : Random
 Media : Hard Disk
 Panjang Record : 40
 Kunci Field : id_lokasi
 Software : Mysql

Tabel II. 12 Spesifikasi File Satuan

No	Nama Field	Tipe	Panjang	Ket
1	id_lokasi	int	20	Primary Key
2	nama_lokasi	varchar	20	
3	ket	text	-	

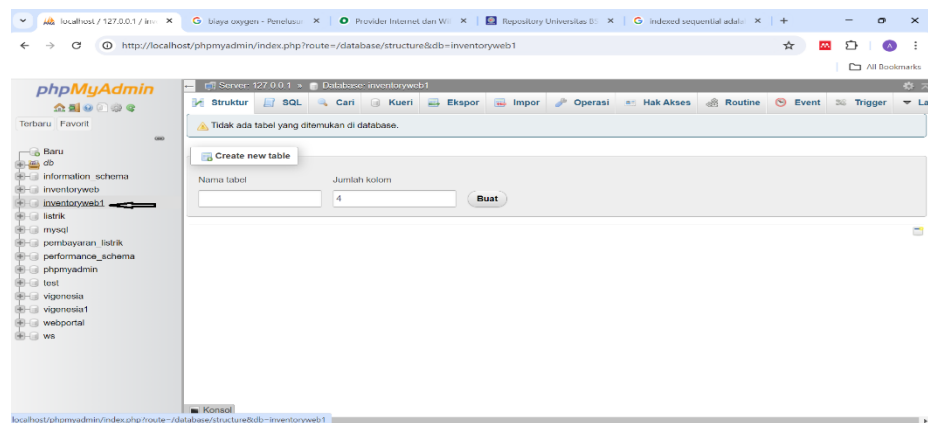
C. Database Inventori Aset

Untuk pembuatan database dapat menggunakan phpMyAdmin. Langkah pertama klik “Baru”, kemudian buat basis data dengan dengan nama “inventoriweb1”, lalu klik “Buat”.



Gambar II. 4 Tampilan Pembuatan Database Baru

Setelah berhasil dibuat akan terlihat database dengan tabel yang masih kosong.

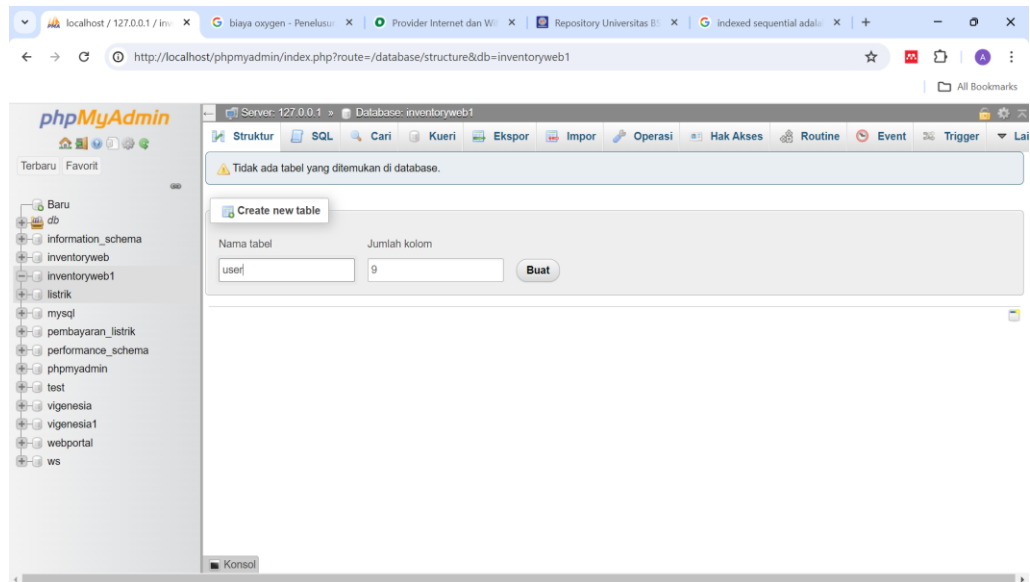


Gambar II. 5 Tampilan Setelah *Database* Berhasil Dibuat

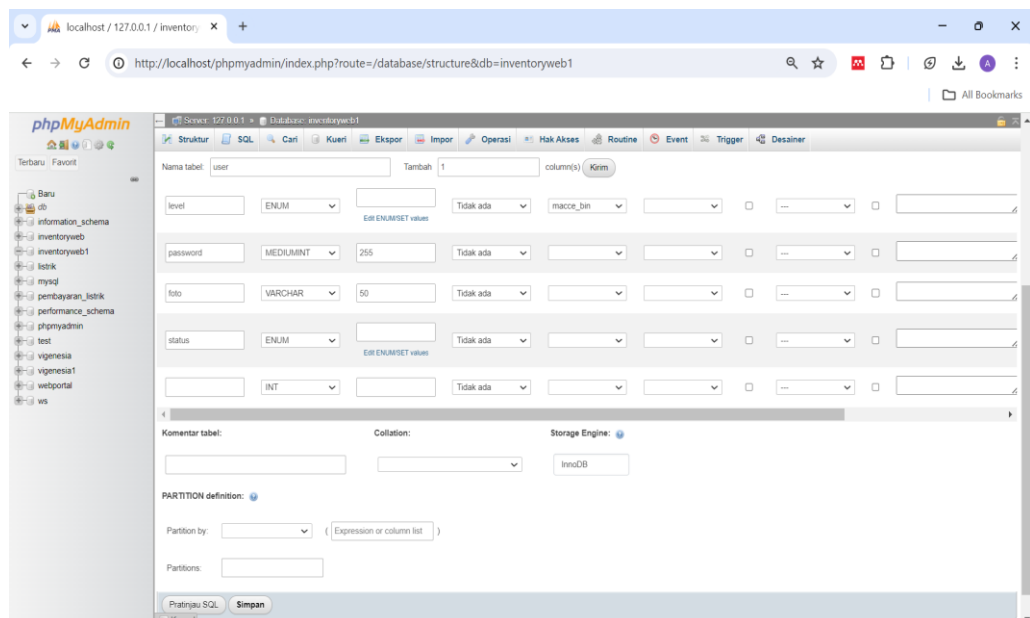
Kemudian, setelah masuk ke *database* inventoriweb1, ketikkan beberapa nama tabel beserta jumlah *field* sesuai dengan tabel yang ada pada spesifikasi file sebelumnya.

A. Tabel User

Isi nama tabel pertama dengan nama “user”. Kemudian tentukan jumlah kolom lalu klik “Buat”.

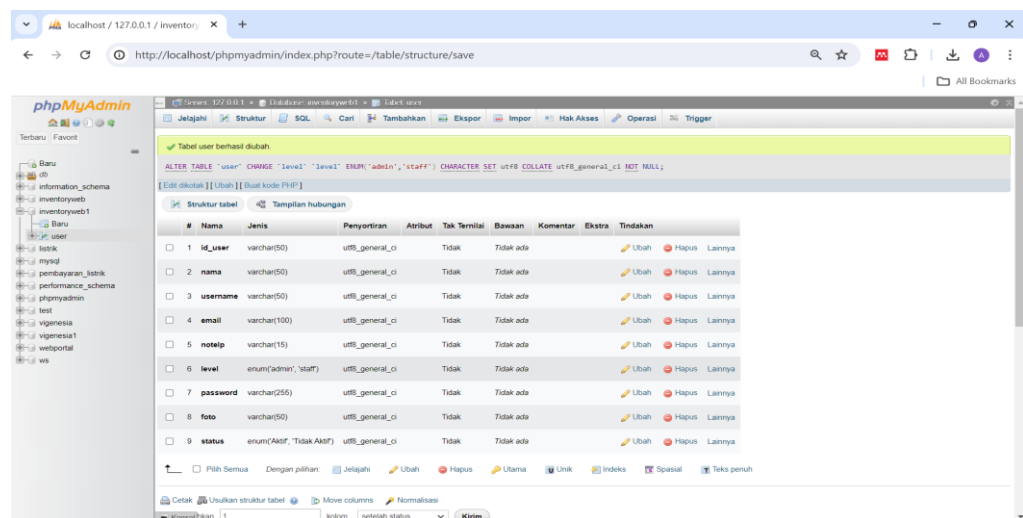


Gambar II. 6 Pengisian Nama Tabel dan Jumlah Kolom



Gambar II. 7 Pengisian Spesifikasi Tabel User

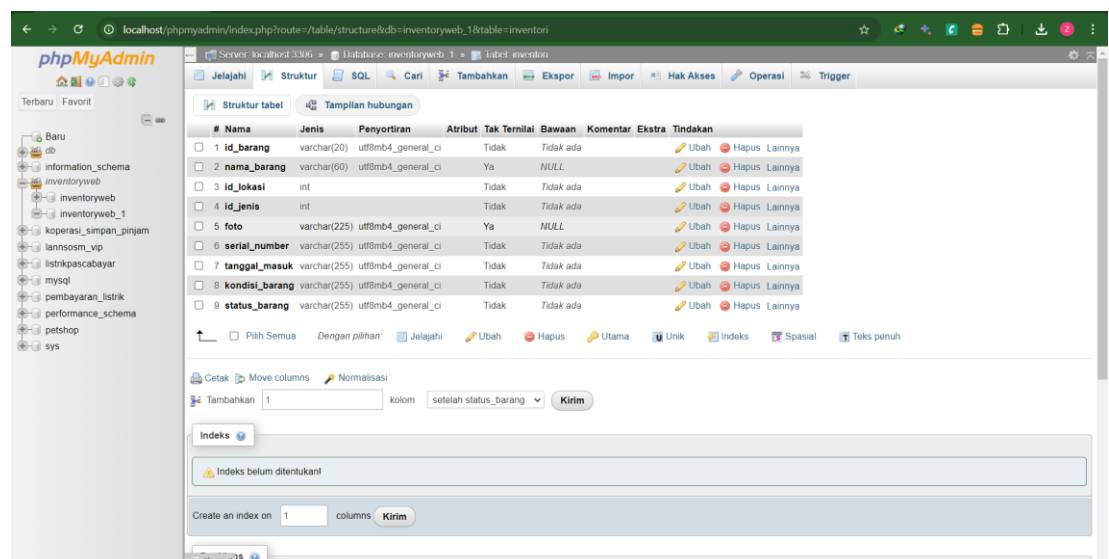
Setelah selesai mengisi nama tabel beserta *field* sesuai dengan *spesifikasi file* klik “Simpan”. Tampilan tabel user ketika berhasil membuat tabel:



Gambar II. 8 Tampilan Setelah Tabel User Berhasil Dibuat

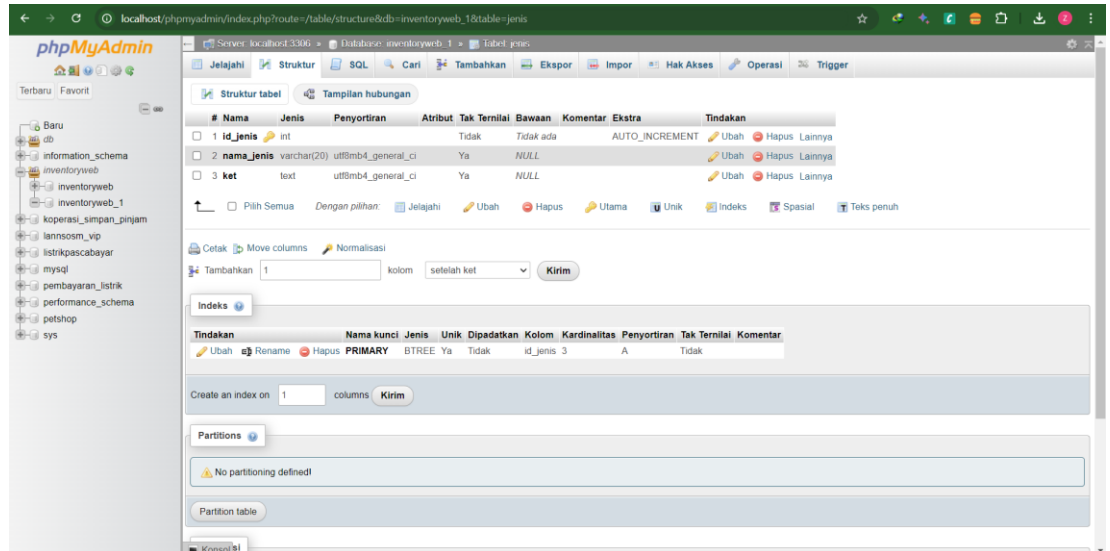
Setelah selesai pengerjaan tabel user, lanjutkan pengerjaan database “inventoriweb1” dengan membuat kembali beberapa tabel berikutnya seperti contoh pembuatan tabel user.

B. Tabel Inventori



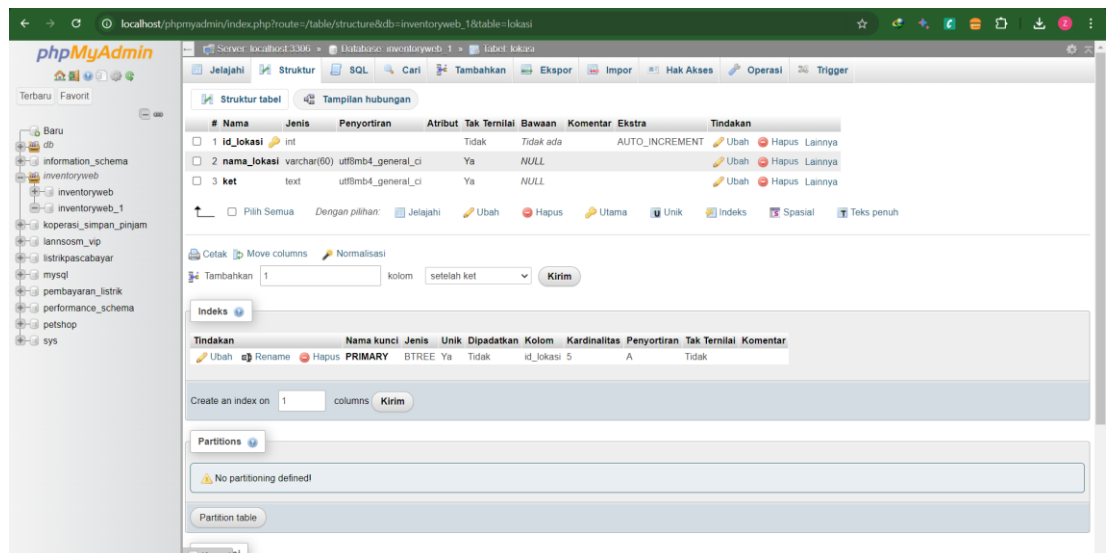
Gambar II. 9 Tampilan Setelah Tabel Inventori Berhasil Dibuat

C. Tabel Jenis

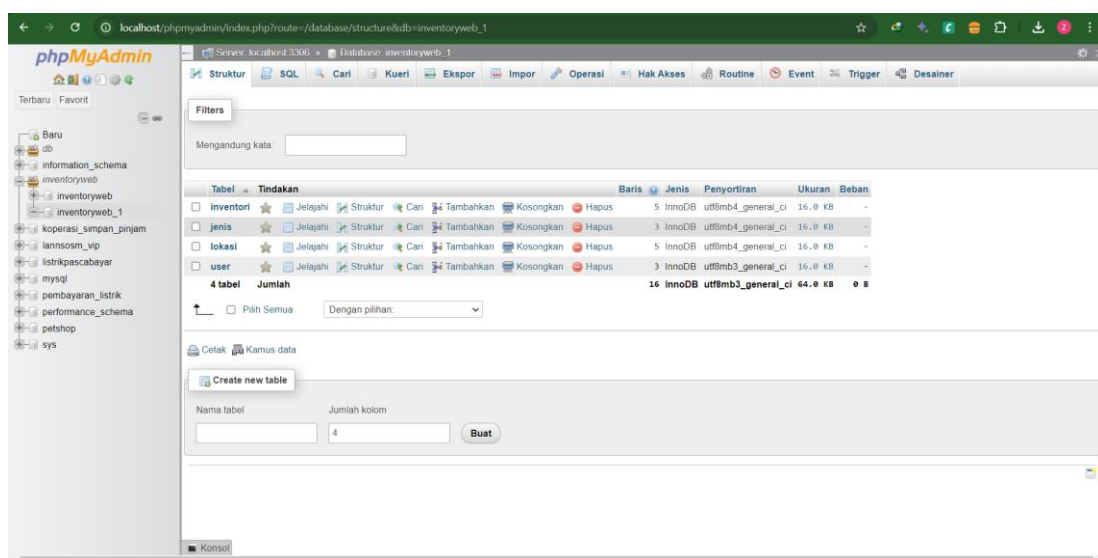


Gambar II. 10 Tampilan Setelah Tabel Jenis Berhasil Dibuat

D. Tabel Lokasi



Gambar II. 11 Tampilan Setelah Tabel Lokasi Berhasil Dibuat



Gambar II. 12 Tampilan Setelah Database “inventoriweb_1” berhasil dibuat

2.3 Rancangan Form Master

Form master pada aplikasi inventori aset berbasis website menggunakan QR code adalah bagian yang berfungsi untuk mengelola informasi-informasi dasar terkait barang dan lokasi pada sistem tersebut. Berikut adalah penjelasan untuk masing-masing form master yang disebutkan:

A. Membuat Halaman Form Master Login



Gambar II. 13 Halaman Login

1. Buat File Controller

Buat file baru dengan nama `Login.php` dalam direktori **application/controllers/** pada proyek CodeIgniter Anda.

```

1 <?php
2 defined('BASEPATH') or exit('No direct script access allowed');
3
4 /**
5  * Author: Danish Yusrah, Adriansyah Putra, Winno Prasetya
6  */
7 class Login extends CI_Controller
8 {
9     public function __construct()
10     {
11         parent::__construct();
12         $this->load->helper('url');
13         $this->load->helper('download');
14         $this->load->library('pagination');
15         $this->load->helper('cookie');
16         $this->load->model('login_model');
17     }
18
19     public function index()
20     {
21         // Memuat view untuk header halaman login
22         $this->load->view('templates/header_login');
23         // Memuat view untuk halaman login
24         $this->load->view('login/index');
25         // Memuat view untuk footer halaman login
26         $this->load->view('templates/footer_login');
27     }
28
29     public function proses_login()
30     {
31         // Mengambil nilai username dan password dari form
32         $username = $this->input->post('user');
33         $password = $this->input->post('pwd');
34
35         // Membuat kondisi untuk pencarian data user berdasarkan username dan password
36         $where = array(
37             'username' => $username,
38             'password' => md5($password) // Password dienkripsi dengan MD5 sebelum dicocokkan
39         );
40
41         // Memanggil model login_model untuk melakukan pengecekan login
42         $cek = $this->login_model->cek_login($where, 'user')->num_rows();
43         $data = $this->login_model->cek_login($where, 'user')->row_array();
44
45         // Jika data login ditemukan, membuat session login
46         if ($cek > 0) {
47             $userdata = [
48                 'id_user' => $data['id_user'],
49                 'username' => $data['nama'],
50                 'status' => $data['status'],
51                 'level' => $data['level'],
52                 'foto' => $data['foto']
53             ];
54
55             $this->session->set_userdata('login_session',$userdata);
56
57             $respon = array('respon' => 'success');
58             echo json_encode($respon);
59         }
60         else {
61             $respon = array('respon' => 'failed');
62             echo json_encode($respon);
63         }
64     }
65
66     public function logout()
67     {
68         // Menghapus session login saat logout
69         $this->session->unset_userdata('login_session');
70         // Mengarahkan pengguna kembali ke halaman login
71         redirect('login');
72     }
73 }
74

```

Gambar II. 14 Controller Login

2. Buat File Model Login

Buat file model untuk login_model sesuai dengan kebutuhan aplikasi Anda. Pastikan model ini berada di direktori **application/models/**. Contoh file **Login_model.php**:

```

1  <?php
2  defined('BASEPATH') or exit('No direct script access allowed');
3
4  class Login_model extends CI_Model
5  {
6      public function __construct()
7      {
8          parent::__construct();
9          // Load database untuk mengakses tabel yang berisi data login
10         $this->load->database();
11     }
12
13     public function cek_login($where, $table)
14     {
15         return $this->db->get_where($table, $where);
16     }
17 }
18

```

Gambar II. 15 Model Login

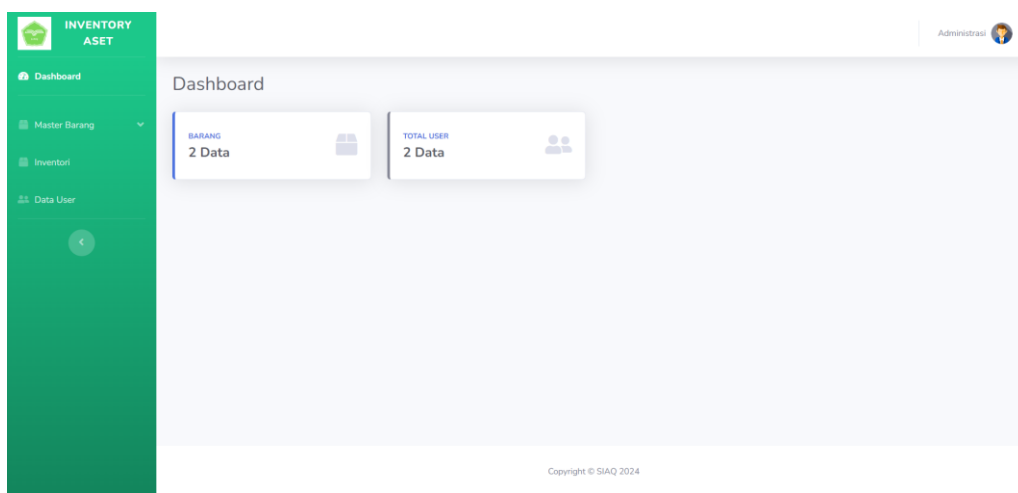
Pastikan juga untuk mengonfigurasi database Anda di **application/config/database.php** sesuai dengan pengaturan yang benar.

3. Buat File View Login

Buat file view untuk header, halaman login, dan footer sesuai dengan struktur yang Anda gunakan. Misalnya, dalam direktori **application/views/**, Anda dapat memiliki file seperti **header_login.php**, **index.php** (untuk halaman login), dan **footer_login.php**. Isi file **header_login.php** dan **footer_login.php** dengan kode yang sesuai untuk tampilan header dan footer halaman login.

Setelah langkah-langkah ini, Anda dapat mengakses halaman login dengan mengunjungi URL aplikasi Anda, misalnya **`http://localhost/nama_aplikasi/`**. Pastikan juga Anda telah mengaktifkan modul url di file **`application/config/autoload.php`** agar fungsi-fungsi URL CodeIgniter seperti `base_url()` dapat digunakan tanpa harus memuat helper secara manual.

B. Membuat Halaman Dashboard Home



Gambar II. 16 Halaman Dashboard

1. Buat File Controller Home

Buat file baru dengan nama Home.php dalam direktori **application/controllers/** pada proyek CodeIgniter Anda.

```

1 <?php
2 defined('BASEPATH') or exit('No direct script access allowed');
3
4 /**
5  * Author: Danish Yusrah, Adriansyah Putra, Winno Prasetya
6  */
7 class Home extends CI_Controller
8 {
9     public function __construct()
10     {
11         parent::__construct();
12         $this->load->helper('url');
13         $this->load->helper('download');
14         $this->load->helper('cookie');
15         $this->load->model('barang_model'); // Memuat model barang_model untuk operasi terkait barang
16         $this->load->model('user_model'); // Memuat model user_model untuk operasi terkait user
17     }
18
19     public function index()
20     {
21         $data['title'] = 'Dashboard'; // Menyimpan judul halaman dalam variabel data
22         // Menghitung jumlah data barang dan jumlah data user menggunakan model masing-masing
23         $data['jmlbarang'] = $this->barang_model->dataJoin()->num_rows();
24         $data['jmlUser'] = $this->user_model->data()->num_rows();
25
26         // Menghitung tahun saat ini, tahun sebelumnya, dan dua tahun yang lalu
27         $data['yearnow'] = date('Y', strtotime('+0 year'));
28         $data['previousyear'] = date('Y', strtotime('-1 year'));
29         $data['twoyearago'] = date('Y', strtotime('-2 year'));
30
31         // Memuat tampilan header dengan data yang disimpan dalam variabel $data
32         $this->load->view('templates/header', $data);
33         // Memuat tampilan halaman utama (dashboard)
34         $this->load->view('home/index');
35         // Memuat tampilan footer
36         $this->load->view('templates/footer');
37     }
38 }
39

```

Gambar II. 17 *Controller Home*

Contoh File User_model.php

```

1  <?php
2  defined('BASEPATH') or exit('No direct script access allowed');
3
4  class User_model extends CI_Model
5  {
6      public function __construct()
7      {
8          parent::__construct();
9          // Load database untuk mengakses tabel yang berisi data user
10         $this->load->database();
11     }
12
13     public function data()
14     {
15         // Misalnya, fungsi ini mengambil data dari tabel user
16         // Anda dapat menggantinya sesuai dengan kebutuhan aplikasi Anda
17         $this->db->select('*');
18         $this->db->from('user');
19         return $this->db->get();
20     }
21 }
22

```

Gambar II. 18 User Model

Pastikan untuk mengonfigurasi database Anda di application/config/database.php sesuai dengan pengaturan yang benar.

2. Buat File View

Buat file view untuk header, halaman utama, dan footer sesuai dengan struktur yang Anda gunakan. Misalnya, dalam direktori **application/views/**, Anda dapat memiliki file seperti **header.php**, **index.php** (untuk halaman utama), dan **footer.php**. Isi file **header.php** dengan kode yang sesuai untuk tampilan header Anda. Demikian pula dengan **footer.php** untuk tampilan footer.

3. Berikut Kodingan secara Lengkapnya

```

1 <?php
2 defined('BASEPATH') or exit('No direct script access allowed');
3
4 /**
5  * Author: Danish Yusrah, Adriansyah Putra, Winno Prasetya
6  */
7 class Home extends CI_Controller
8 {
9     public function __construct()
10     {
11         parent::__construct();
12         $this->load->helper('url');
13         $this->load->helper('download');
14         $this->load->helper('cookie');
15         $this->load->model('barang_model'); // Memuat model barang_model untuk operasi terkait barang
16         $this->load->model('user_model'); // Memuat model user_model untuk operasi terkait user
17     }
18
19     public function index()
20     {
21         $data['title'] = 'Dashboard'; // Menyimpan judul halaman dalam variabel data
22         // Menghitung jumlah data barang dan jumlah data user menggunakan model masing-masing
23         $data['jmlbarang'] = $this->barang_model->dataJoin()->num_rows();
24         $data['jmlUser'] = $this->user_model->data()->num_rows();
25
26         // Menghitung tahun saat ini, tahun sebelumnya, dan dua tahun yang lalu
27         $data['yearnow'] = date('Y', strtotime('+0 year'));
28         $data['previousyear'] = date('Y', strtotime('-1 year'));
29         $data['twoyearago'] = date('Y', strtotime('-2 year'));
30
31         // Memuat tampilan header dengan data yang disimpan dalam variabel $data
32         $this->load->view('templates/header', $data);
33         // Memuat tampilan halaman utama (dashboard)
34         $this->load->view('home/index');
35         // Memuat tampilan footer
36         $this->load->view('templates/footer');
37     }
38 }

```

Gambar II. 19 *Controller Home* Lengkap

C. Membuat Halaman Form Data Lokasi Master Barang

The screenshot displays the 'Data Lokasi' page within the 'INVENTORY ASET' application. The interface includes a sidebar with navigation options: Dashboard, Master Barang, Inventori, and Data User. The main content area shows a table titled 'Data Lokasi' with columns for 'No', 'Lokasi Barang', 'Keterangan', and 'Aksi'. There are 4 entries listed, all with 'PUSDATIN RUANG' as the location. The 'Aksi' column contains green checkmarks and red minus signs. At the bottom of the table, it says 'Showing 1 to 4 of 4 entries' and includes 'Previous', '1', and 'Next' navigation links. The top right corner shows the user 'Administrasi' and a 'Tambah Data' button.

No	Lokasi Barang	Keterangan	Aksi
1.	PUSDATIN RUANG 5	PUSDATIN RUANG 5	 
2.	PUSDATIN RUANG 1	PUSDATIN RUANG 1	 
3.	PUSDATIN RUANG 2	PUSDATIN RUANG 2	 
4.	PUSDATIN RUANG 3	PUSDATIN RUANG 3	 

Showing 1 to 4 of 4 entries

Previous 1 Next

Copyright © SIAQ 2024

Gambar II. 20 Halaman Lokasi Barang

1. Buat File Controller Lokasi

Buat file baru dengan nama **Lokasi.php** dalam direktori `application/controllers/` pada proyek CodeIgniter Anda.

```

1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  /**
5   * Author: Danish Yusrah, Adriansyah Putra, Winno Prasetya
6   */
7
8  class Lokasi extends CI_Controller {
9
10     public function __construct() {
11         parent::__construct();
12         $this->load->helper('url');
13         $this->load->helper('download');
14         $this->load->library('pagination');
15         $this->load->helper('cookie');
16         $this->load->model('lokasi_model');
17     }
18
19     public function index()
20     {
21         $data['title'] = 'Lokasi Barang';
22         $data['lokasi'] = $this->lokasi_model->data()->result();
23
24         $this->load->view('templates/header', $data);
25         $this->load->view('lokasi/index');
26         $this->load->view('templates/footer');
27     }
28
29     public function proses_tambah()
30     {
31         $lokasi = $this->input->post('lokasi');
32         $ket = $this->input->post('ket');
33
34         $data=array(
35             'nama_lokasi' => $lokasi,
36             'ket' => $ket,
37         );
38
39         $this->lokasi_model->tambah_data($data, 'lokasi');
40         $this->session->set_flashdata('Pesan', '
41         <script>
42         $(document).ready(function() {
43             swal.fire({
44                 title: "Berhasil ditambahkan!",
45                 icon: "success",
46                 confirmButtonColor: "#4e73df",
47             });
48         });
49         </script>
50         ');
51         redirect('lokasi');
52     }
53
54     public function proses_ubah()
55     {
56         $kode = $this->input->post('id_lokasi');
57         $lokasi = $this->input->post('lokasi');
58         $ket = $this->input->post('ket');
59
60         $data=array(
61             'nama_lokasi' => $lokasi,
62             'ket' => $ket
63         );
64
65         $where = array(
66             'id_lokasi' => $kode
67         );
68
69         $this->lokasi_model->ubah_data($where, $data, 'lokasi');
70         $this->session->set_flashdata('Pesan', '
71         <script>
72         $(document).ready(function() {
73             swal.fire({
74                 title: "Berhasil diubah!",
75                 icon: "success",
76                 confirmButtonColor: "#4e73df",
77             });
78         });
79         </script>
80         ');
81         redirect('lokasi');
82     }
83
84     public function proses_hapus($id)
85     {
86         $where = array('id_lokasi' => $id );
87         $this->lokasi_model->hapus_data($where, 'lokasi');
88         $this->session->set_flashdata('Pesan', '
89         <script>
90         $(document).ready(function() {
91             swal.fire({
92                 title: "Berhasil dihapus!",
93                 icon: "success",
94                 confirmButtonColor: "#4e73df",
95             });
96         });
97         </script>
98         ');
99         redirect('lokasi');
100    }
101
102    public function getData()
103    {
104        $id = $this->input->post('id');
105        $where = array('id_lokasi' => $id );
106        $data = $this->lokasi_model->detail_data($where, 'lokasi')->result();
107        echo json_encode($data);
108    }
109 }
110

```

Gambar II. 21 *Controller* Lokasi Barang

2. Buat File Model Lokasi

Buat file model untuk satuan_model sesuai dengan kebutuhan aplikasi Anda. Pastikan model ini berada di direktori **application/models/**. Contoh file **Lokasi_model.php**:

Pastikan juga untuk mengonfigurasi database Anda di **application/config/database.php** sesuai dengan pengaturan yang benar.

```

1  <?php
2
3  /**
4   * Author: Danish Yusrat, Adriansyah Putra, Winno Prasetya
5   */
6  class satuan_model extends ci_model{
7
8
9      function data()
10     {
11         $this->db->order_by('id_satuan','DESC');
12         return $query = $this->db->get('satuan');
13     }
14
15     public function ambilId($table, $where)
16     {
17         return $this->db->get_where($table, $where);
18     }
19
20     public function hapus_data($where, $table)
21     {
22         $this->db->where($where);
23         $this->db->delete($table);
24         if ($this->db->affected_rows() == 1) {
25             return TRUE;
26         }
27         return false;
28     }
29
30
31
32
33     public function detail_data($where, $table)
34     {
35         return $this->db->get_where($table,$where);
36     }
37
38     public function tambah_data($data, $table)
39     {
40         $this->db->insert($table, $data);
41     }
42
43     public function ubah_data($where, $data, $table)
44     {
45         $this->db->where($where);
46         $this->db->update($table, $data);
47     }
48
49
50
51     public function buat_kode() {
52         $this->db->select('RIGHT(satuan.id_satuan,4) as kode', FALSE);
53         $this->db->order_by('id_satuan','DESC');
54         $this->db->limit(1);
55         $query = $this->db->get('satuan'); //cek dulu apakah ada sudah ada kode di tabel.
56         if($query->num_rows() > 0){
57             //jika kode ternyata sudah ada.
58             $data = $query->row();
59             $kode = intval($data->kode) + 1;
60         }
61         else {
62             //jika kode belum ada
63             $kode = 1;
64         }
65         $kodemax = str_pad($kode, 4, "0", STR_PAD_LEFT); // angka 4 menunjukkan jumlah digit angka 0
66         $kodejadi = "STN-".$kodemax;
67         return $kodejadi;
68     }
69
70
71
72
73
74
75

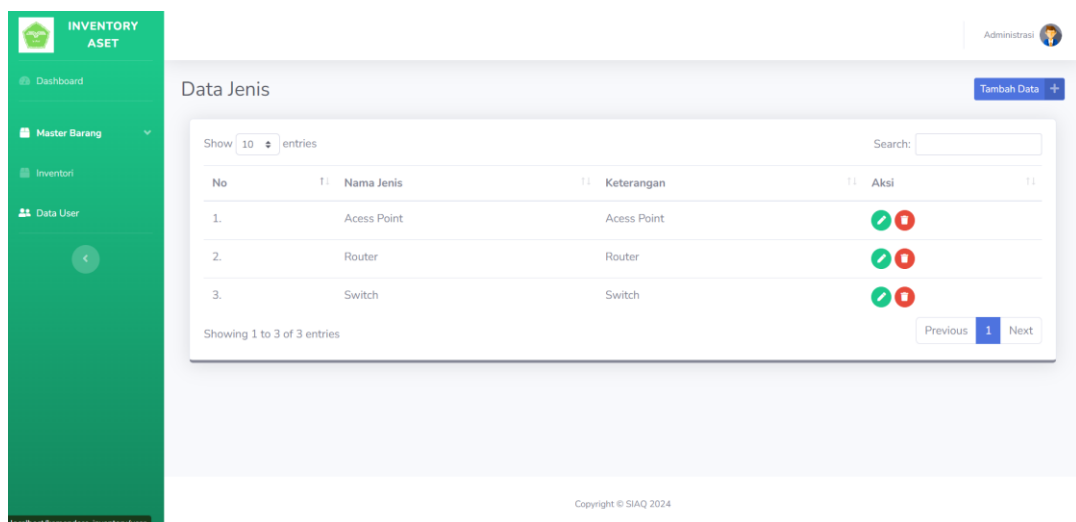
```

Gambar II. 22 Lokasi Model

3. Buat File View

Buat file view untuk header, halaman index satuan, dan footer sesuai dengan struktur yang Anda gunakan. Misalnya, dalam direktori **application/views/**, Anda dapat memiliki file seperti **header.php**, **index.php** (untuk halaman index satuan), dan **footer.php**. Isi file **header.php**

D. Membuat Halaman Data Form Jenis Master Barang



Gambar II. 23 Halaman Jenis Barang

1. Buat File Controller Jenis

Buat file baru dengan nama **Jenis.php** dalam direktori **application/controllers/** pada proyek CodeIgniter Anda.

```

1  <?php
2  defined('BASEPATH') or exit('No direct script access allowed');
3
4  /**
5   * Author: Danish Yusrah, Adriansyah Putra, Winno Prasetya
6   */
7  class Jenis extends CI_Controller
8  {
9      public function __construct()
10     {
11         parent::__construct();
12         $this->load->helper('url');
13         $this->load->helper('download');
14         $this->load->library('pagination');
15         $this->load->helper('cookie');
16         $this->load->model('jenis_model');
17     }
18
19     public function index()
20     {
21         $data['title'] = 'Jenis Barang';
22         $data['jenis'] = $this->jenis_model->data()->result();
23
24         $this->load->view('templates/header', $data);
25         $this->load->view('jenis/index');
26         $this->load->view('templates/footer');
27     }
28
29     public function proses_tambah()
30     {
31         $jenis = $this->input->post('jenis');
32         $ket = $this->input->post('ket');
33
34         $data = array(
35             'nama_jenis' => $jenis,
36             'ket' => $ket,
37         );
38
39         $this->jenis_model->tambah_data($data, 'jenis');
40
41         $this->session->set_flashdata('Pesan', '
42         <script>
43         $(document).ready(function() {
44             swal.fire({
45                 title: "Berhasil ditambahkan!",
46                 icon: "success",
47                 confirmButtonColor: "#4e73df",
48             });
49         });
50         </script>
51         ');
52
53         redirect('jenis');
54     }
55
56     public function proses_ubah()
57     {
58         $kode = $this->input->post('idjenis');
59         $jenis = $this->input->post('jenis');
60         $ket = $this->input->post('ket');
61
62         $data = array(
63             'nama_jenis' => $jenis,
64             'ket' => $ket,
65         );
66
67         $where = array(
68             'id_jenis' => $kode
69         );
70
71         $this->jenis_model->ubah_data($where, $data, 'jenis');
72
73         $this->session->set_flashdata('Pesan', '
74         <script>
75         $(document).ready(function() {
76             swal.fire({
77                 title: "Berhasil diubah!",
78                 icon: "success",
79                 confirmButtonColor: "#4e73df",
80             });
81         });
82         </script>
83         ');
84
85         redirect('jenis');
86     }
87
88     public function proses_hapus($id)
89     {
90         $where = array('id_jenis' => $id);
91         $this->jenis_model->hapus_data($where, 'jenis');
92
93         $this->session->set_flashdata('Pesan', '
94         <script>
95         $(document).ready(function() {
96             swal.fire({
97                 title: "Berhasil dihapus!",
98                 icon: "success",
99                 confirmButtonColor: "#4e73df",
100             });
101         });
102         </script>
103         ');
104
105         redirect('jenis');
106     }
107
108     public function getData()
109     {
110         $id = $this->input->post('id');
111         $where = array('id_jenis' => $id );
112         $data = $this->jenis_model->detail_data($where, 'jenis')->result();
113         echo json_encode($data);
114     }
115 }
116

```

Gambar II. 24 Controller Jenis Barang

2. Buat File Model Jenis

Buat file model untuk jenis_model sesuai dengan kebutuhan aplikasi Anda. Pastikan model ini berada di direktori **application/models/**. Contoh file **Jenis_model.php**:

```

1  <?php
2  defined('BASEPATH') or exit('No direct script access allowed');
3
4  class Jenis_model extends CI_Model
5  {
6      public function __construct()
7      {
8          parent::__construct();
9          $this->load->database();
10     }
11
12     public function data()
13     {
14         return $this->db->get('jenis');
15     }
16
17     public function tambah_data($data, $table)
18     {
19         $this->db->insert($table, $data);
20     }
21
22     public function ubah_data($where, $data, $table)
23     {
24         $this->db->where($where);
25         $this->db->update($table, $data);
26     }
27
28     public function hapus_data($where, $table)
29     {
30         $this->db->where($where);
31         $this->db->delete($table);
32     }
33
34     public function detail_data($where, $table)
35     {
36         return $this->db->get_where($table, $where);
37     }
38 }
39

```

Gambar II. 25 *Jenis Model*

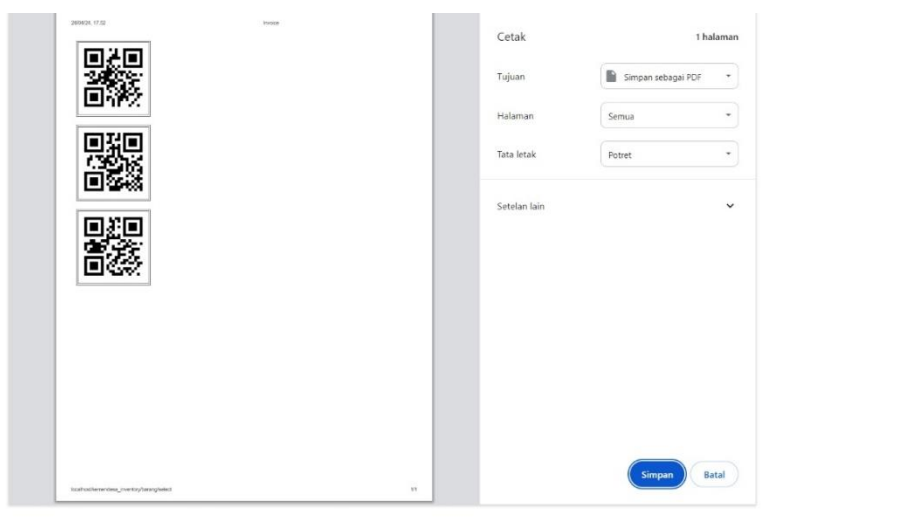
Pastikan juga untuk mengonfigurasi database Anda di **application/config/database.php** sesuai dengan pengaturan yang benar.

3. Buat File View

Buat file view untuk header, halaman index jenis, dan footer sesuai dengan struktur yang Anda gunakan. Misalnya, dalam direktori **application/views/**, Anda dapat memiliki file seperti **header.php**, **index.php** (untuk halaman index jenis), dan **footer.php**. Isi file **header.php** dan **footer.php** dengan kode yang sesuai untuk tampilan header dan footer halaman.

Setelah langkah-langkah ini, Anda dapat mengakses halaman index jenis dengan mengunjungi URL aplikasi Anda, misalnya **http://localhost/nama_aplikasi/jenis**. Pastikan juga Anda telah mengaktifkan modul url di file **application/config/autoload.php** agar fungsi-fungsi URL CodeIgniter seperti **base_url()** dapat digunakan tanpa harus memuat helper secara manual.

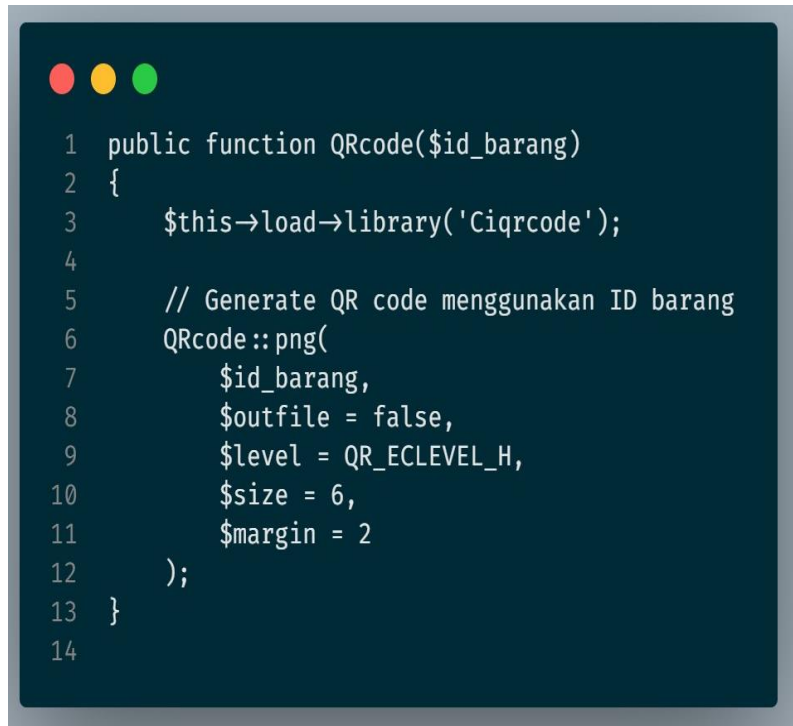
E. Halaman Form Cetak QRCode Master Barang



Gambar II. 26 Halaman Form Cetak QRCode

1. Buat Controller untuk QRcode

Tambahkan fungsi QRcode pada controller Barang yang dapat menerima parameter ID barang untuk menghasilkan QR code.

A screenshot of a code editor with a dark blue background and light green text. The code is a PHP function named `QRcode` that takes `$id_barang` as an argument. It loads the `Ciqrbarcode` library and uses the `QRcode::png` method to generate a QR code. The method is called with `$id_barang`, `$outfile = false`, `$level = QR_ECLEVEL_H`, `$size = 6`, and `$margin = 2`. The code is numbered from 1 to 14.

```
1 public function QRcode($id_barang)
2 {
3     $this->load->library('Ciqrbarcode');
4
5     // Generate QR code menggunakan ID barang
6     QRcode::png(
7         $id_barang,
8         $outfile = false,
9         $level = QR_ECLEVEL_H,
10        $size = 6,
11        $margin = 2
12    );
13 }
14
```

Gambar II. 27 *Controller QRCode*

2. Pastikan Data Barang Terpilih Tersedia

Pastikan bahwa variabel `$selected_barcode` pada halaman invoice Anda berisi data yang dipilih dari database atau sumber data lainnya.

3. Perbarui HTML Invoice

Perbarui kode HTML Anda untuk mencetak QR code berdasarkan data barang yang dipilih.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Invoice</title>
6   <link href="https://fonts.googleapis.com/css?family=Nunito" rel="stylesheet">
7   <style media="print">
8     @page {
9       size: A4;
10      margin: 0;
11    }
12
13    body {
14      font-family: 'Nunito', sans-serif;
15      margin: 50px; /* Margin agar tidak tercetak di tepi kertas */
16    }
17
18    .container {
19      display: flex;
20      flex-wrap: wrap;
21      gap: 10px; /* Jarak antar QR code */
22    }
23
24    .qr-code {
25      width: calc(25% - 10px); /* Ukuran kolom QR code */
26      padding: 5px;
27      text-align: center;
28      border: 1px solid black;
29    }
30
31    img {
32      max-width: 100%;
33      height: auto;
34    }
35
36    .page-break {
37      page-break-after: always;
38    }
39
40    /* Style khusus untuk halaman terakhir jika kurang dari 12 barcode */
41    @media print {
42      @page :last {
43        margin-bottom: 50px; /* Berikan margin bawah agar barcode tidak terpotong */
44      }
45    }
46  </style>
47 </head>
48
```

```

1 <body>
2 <?php
3     $count = 0; // inisialisasi variabel hitung
4     $barcodes_per_page = 12; // 3 mendatar x 4 menurun = 12 QR code per halaman
5
6     // Balik urutan array barang agar yang pertama berada di atas
7     $reversed_barang = array_reverse($barang);
8
9     $total_barcodes = count($reversed_barang);
10    $pages = ceil($total_barcodes / $barcodes_per_page);
11    ?>
12

```

```

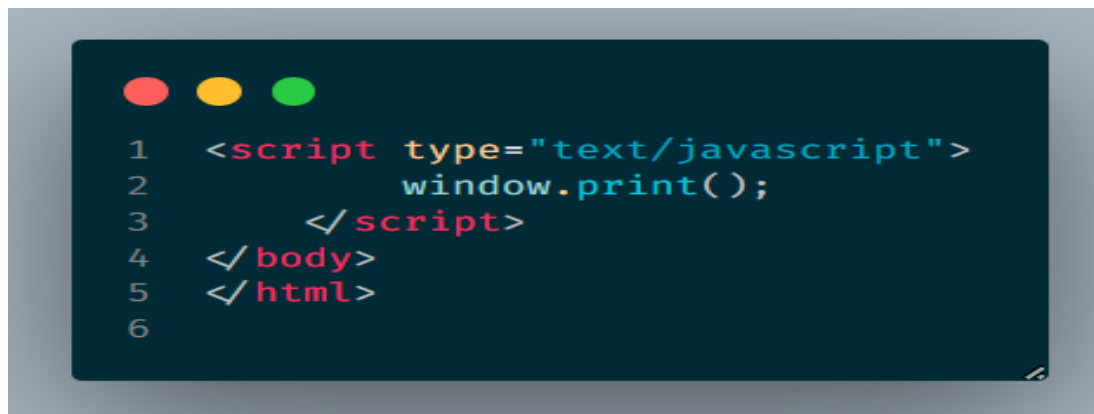
1 <?php for ($page = 1; $page ≤ $pages; $page++) : ?>
2     <?php if ($page > 1) : ?>
3         <div class="page-break"></div>
4     <?php endif; ?>
5     <div class="container">
6

```

```

1 <?php
2     $start = ($page - 1) * $barcodes_per_page;
3     $end = min($start + $barcodes_per_page, $total_barcodes);
4
5     for ($i = $start; $i < $end; $i++) :
6         ?>
7         <div class="qr-code">
8             
9         </div>
10    <?php endfor; ?>
11    </div> <!-- Tutup container untuk halaman -->
12    <?php endfor; ?>
13

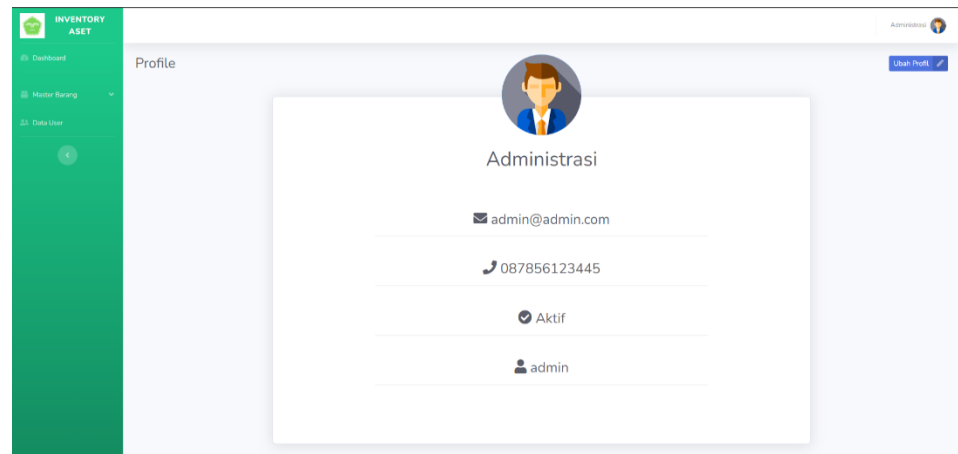
```



Gambar II. 28 *HTML untuk QRCode*

Pastikan bahwa URL QR code di tag `<img>` sesuai dengan struktur URL pada aplikasi CodeIgniter Anda. Jika ada perbedaan pada struktur URL, sesuaikan dengan base URL dan rute yang digunakan dalam aplikasi Anda.

F. Halaman Profile



Gambar II. 29 *Halaman Profile*

1. Penggunaan Helper dan Library



```
1 $this->load->helper('url');
2 $this->load->helper('download');
3 $this->load->library('pagination');
4
```

Gambar II. 30 *Helper dan Library*

Helper url digunakan untuk pengaturan URL pada aplikasi. Helper download digunakan untuk download file. Library pagination digunakan untuk mengatur halaman.

2. Load Model()



```
1 $this->load->model('user_model');
2
```

Gambar II. 31 *Load Model*

Ini memuat model user_model yang digunakan untuk interaksi dengan database terkait pengguna.

3. Metode index()

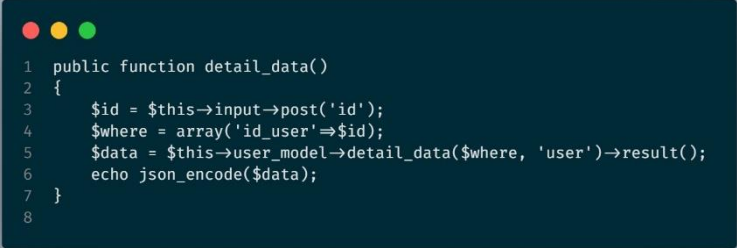


```
1 public function index()
2 {
3     $data['title'] = 'Profile';
4     $this->load->view('templates/header', $data);
5     $this->load->view('profile/index');
6     $this->load->view('templates/footer');
7 }
8
```

Gambar II. 32 *Metode Index*

Fungsi ini memuat halaman utama profil dengan judul 'Profile'.

4. Metode detail_data()

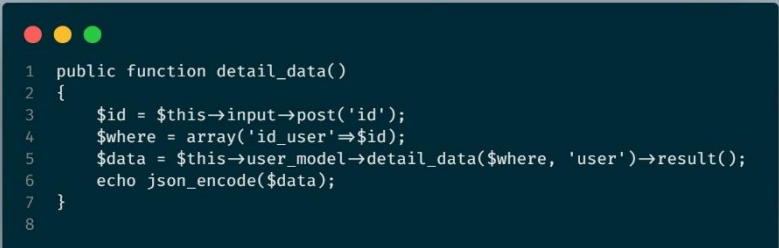


```
1 public function detail_data()
2 {
3     $id = $this->input->post('id');
4     $where = array('id_user'=>$id);
5     $data = $this->user_model->detail_data($where, 'user')->result();
6     echo json_encode($data);
7 }
8
```

Gambar II. 33 Metode Detail Data

Metode ini mengambil data detail pengguna berdasarkan ID pengguna yang dipost melalui AJAX dan mengembalikan data dalam format JSON.

5. Metode ubah(\$id)

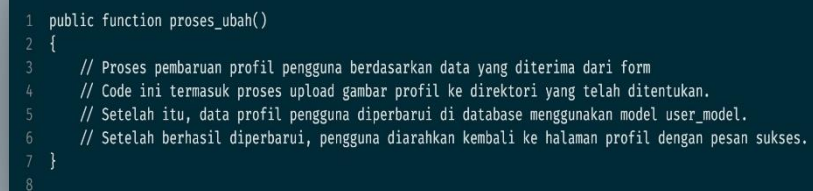


```
1 public function detail_data()
2 {
3     $id = $this->input->post('id');
4     $where = array('id_user'=>$id);
5     $data = $this->user_model->detail_data($where, 'user')->result();
6     echo json_encode($data);
7 }
8
```

Gambar II. 34 Metode ubah(id\$)

Fungsi ini memuat halaman form untuk mengubah profil pengguna berdasarkan ID pengguna yang diberikan.

6. Metode proses_ubah()



```
1 public function proses_ubah()
2 {
3     // Proses pembaruan profil pengguna berdasarkan data yang diterima dari form
4     // Code ini termasuk proses upload gambar profil ke direktori yang telah ditentukan.
5     // Setelah itu, data profil pengguna diperbarui di database menggunakan model user_model.
6     // Setelah berhasil diperbarui, pengguna diarahkan kembali ke halaman profil dengan pesan sukses.
7 }
8
```

Gambar II. 35 Metode Proses Ubah Profile

```
// Setelah berhasil diperbarui, pengguna diarahkan kembali
ke halaman profil dengan pesan sukses.
}
```

Metode ini memproses pembaruan profil pengguna berdasarkan data yang diterima dari form, termasuk proses upload gambar profil dan perbaruan data di database

7. Berikut Kodingan Controller Profile Secara Lengkap


```

1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  /**
5   * Author: Danish Yusrah, Adriansyah Putra, Winno Prasetya
6   */
7
8  class Profile extends CI_Controller {
9      public function __construct() {
10         parent::__construct();
11         $this->load->helper('url');
12         $this->load->helper('download');
13         $this->load->library('pagination');
14         $this->load->helper('cookie');
15         $this->load->model('user_model');
16     }
17

```

```

1  public function index() {
2      $data['title'] = 'Profile';
3      $this->load->view('templates/header', $data);
4      $this->load->view('profile/index');
5      $this->load->view('templates/footer');
6  }
7
8  public function detail_data() {
9      $id = $this->input->post('id');
10     $where = array('id_user' => $id);
11     $data = $this->user_model->detail_data($where, 'user')->result();
12     echo json_encode($data);
13 }
14
15 public function ubah($id) {
16     $data['title'] = 'Profile';
17     $where = array('id_user' => $id);
18     $data['user'] = $this->user_model->detail_data($where, 'user')->result();
19     $this->load->view('templates/header', $data);
20     $this->load->view('profile/ubah_profil');
21     $this->load->view('templates/footer');
22 }
23

```

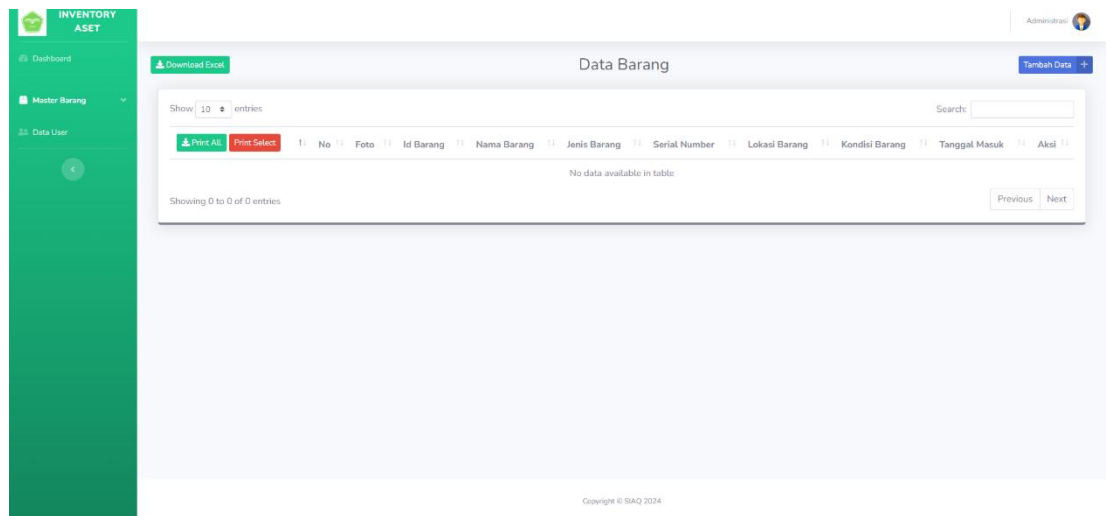
```

1 public function proses_ubah() {
2     $config['upload_path'] = './assets/upload/pengguna/';
3     $config['allowed_types'] = 'png|jpg|JPG|jpeg|JPEG|gif|GIF|tif|TIF|tiff|TIFF';
4
5     $namaFile = $_FILES['photo']['name'];
6     $error = $_FILES['photo']['error'];
7     $this->load->library('upload', $config);
8
9     $kode = $this->input->post('iduser');
10    $namaL = $this->input->post('namaL');
11    $user = $this->input->post('user');
12    $notelp = $this->input->post('notelp');
13    $email = $this->input->post('email');
14    $level = $this->input->post('level');
15    $status = $this->input->post('status');
16    $pass = $this->input->post('pwd');
17    $passLama = $this->input->post('pwdLama');
18    $flama = $this->input->post('fotoLama');
19
20    if($pass == ''){
21        $passUpdate = $passLama;
22    } else {
23        $passUpdate = md5($pass);
24    }
25
26    if ($namaFile == '') {
27        $ganti = $flama;
28    } else {
29        if (!$this->upload->do_upload('photo')) {
30            $error = $this->upload->display_errors();
31            redirect('user/ubah/'.$kode);
32        } else {
33            $data = array('photo' => $this->upload->data());
34            $nama_file = $data['photo']['file_name'];
35            $ganti = str_replace(" ", "_", $nama_file);
36            if($flama == ''){
37                unlink('./assets/upload/pengguna/'.$flama.'');
38            }
39        }
40    }
41
42    $data = array(
43        'nama' => $namaL,
44        'username' => $user,
45        'notelp' => $notelp,
46        'email' => $email,
47        'level' => $level,
48        'password' => $passUpdate,
49        'status' => $status,
50        'foto' => $ganti
51    );
52
53    $where = array('id_user' => $kode);
54
55    $this->user_model->ubah_data($where, $data, 'user');
56    $this->session->set_flashdata('Pesan', '
57    <script>
58    $(document).ready(function() {
59        swal.fire({
60            title: "Berhasil diubah!",
61            icon: "success",
62            confirmButtonColor: "#4e73df",
63        });
64    });
65    </script>
66    ');
67    redirect('profile');
68 }
69 }
70 }
71

```

Gambar II. 36 Controller Profile

G. Halaman Form Data Pengguna



Gambar II. 37 Halaman Data Pengguna

1. Penggunaan Helper dan Library



Gambar II. 38 *Helper dan Library User*

Helper url digunakan untuk pengaturan URL pada aplikasi. Helper download digunakan untuk download file. Library pagination digunakan untuk mengatur halaman.

2. Load Model

Ini memuat model `user_model` yang digunakan untuk interaksi dengan data pengguna di database



Gambar II. 39 *Load Model User*

3. Metode index()

```

1 public function index()
2 {
3     $data['title'] = 'Profile';
4     $this->load->view('templates/header', $data);
5     $this->load->view('profile/index');
6     $this->load->view('templates/footer');
7 }
8

```

Gambar II. 40 Metode *Index Profile*

Fungsi ini memuat halaman utama pengguna dengan judul 'User'.

4. Metode tambah()

```

1 public function tambah()
2 {
3     $data['title'] = 'User';
4
5     $this->load->view('templates/header', $data);
6     $this->load->view('user/form_tambah');
7     $this->load->view('templates/footer');
8 }
9

```

Gambar II. 41 Metode Tambah Data *User*

Fungsi ini memuat halaman form untuk menambahkan pengguna baru.

5. Metode ubah(\$id)

```

1 public function ubah($id)
2 {
3     $data['title'] = 'User';
4     $where = array('id_user'=>$id);
5     $data['user'] = $this->user_model->detail_data($where, 'user')->result();
6
7     $this->load->view('templates/header', $data);
8     $this->load->view('user/form_ubah');
9     $this->load->view('templates/footer');
10 }
11

```

Gambar II. 42 Metode Ubah Data *User*

Fungsi ini memuat halaman form untuk mengubah data pengguna berdasarkan ID pengguna.

6. Metode detail_data(\$id)

```

1 public function detail_data()
2 {
3     $id = $this->input->post('id');
4     $where = array('id_user'=>$id);
5     $data = $this->user_model->detail_data($where, 'user')->result();
6     echo json_encode($data);
7 }
8

```

Gambar II. 43 Metode Detail Data *User*

Fungsi ini memuat halaman detail pengguna berdasarkan ID pengguna.

7. Metode proses_hapus(\$id)

```

1 public function proses_hapus($id)
2 {
3     // Proses menghapus data pengguna dari database berdasarkan ID
4     // Juga menghapus foto profil pengguna jika bukan foto default
5     // Menampilkan pesan sukses dan redirect kembali ke halaman pengguna
6 }
7

```

Gambar II. 44 Metode Hapus Data *User*

Fungsi ini memproses penghapusan data pengguna dari database. Fungsi ini memproses penghapusan data pengguna dari database.

8. Metode proses_tambah()

```

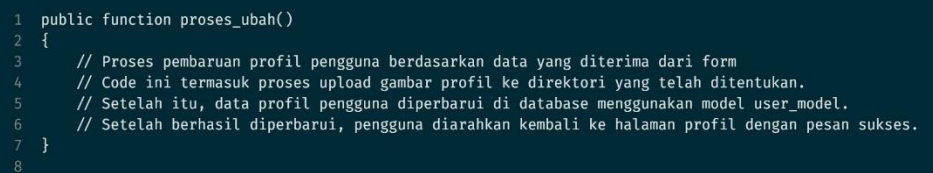
1 public function proses_tambah()
2 {
3     // Proses penambahan data pengguna baru ke database
4     // Mengelola upload foto profil dan data pengguna
5     // Menampilkan pesan sukses dan redirect kembali ke halaman pengguna
6 }
7

```

Gambar II. 45 Metode Tambah Data *User*

Fungsi ini memproses penambahan data pengguna baru ke database.

9. Metode proses_ubah()



```
1 public function proses_ubah()
2 {
3     // Proses pembaruan profil pengguna berdasarkan data yang diterima dari form
4     // Code ini termasuk proses upload gambar profil ke direktori yang telah ditentukan.
5     // Setelah itu, data profil pengguna diperbarui di database menggunakan model user_model.
6     // Setelah berhasil diperbarui, pengguna diarahkan kembali ke halaman profil dengan pesan sukses.
7 }
8
```

Gambar II. 46 Metode Ubah Data *User*

Fungsi ini memproses pembaruan data pengguna di database.

10. Berikut ini Kodingan Lengkapnya

```

1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  /**
5   * Author: Danish Yusrah, Adriansyah Putra, Winno Prasetya
6   */
7
8  class User extends CI_Controller {
9
10     public function __construct() {
11         parent::__construct();
12         $this->load->helper('url');
13         $this->load->helper('download');
14         $this->load->library('pagination');
15         $this->load->helper('cookie');
16         $this->load->model('user_model');
17     }
18
19     public function index() {
20         $data['title'] = 'User';
21         $data['user'] = $this->user_model->data()->result();
22         $this->load->view('templates/header', $data);
23         $this->load->view('user/index');
24         $this->load->view('templates/footer');
25     }
26

```

```

1  public function tambah() {
2      $data['title'] = 'User';
3      $this->load->view('templates/header', $data);
4      $this->load->view('user/form_tambah');
5      $this->load->view('templates/footer');
6  }
7
8  public function ubah($id) {
9      $data['title'] = 'User';
10     $where = array('id_user'=>$id);
11     $data['user'] = $this->user_model->detail_data($where, 'user')->result();
12     $this->load->view('templates/header', $data);
13     $this->load->view('user/form_ubah');
14     $this->load->view('templates/footer');
15 }
16
17 public function detail_data($id) {
18     $data['title'] = 'User';
19     $where = array('id_user'=>$id);
20     $data['data'] = $this->user_model->detail_data($where, 'user')->result();
21     $this->load->view('templates/header', $data);
22     $this->load->view('user/detail');
23     $this->load->view('templates/footer');
24 }
25

```

```
1 public function proses_hapus($id) {
2     $where = array('id_user' => $id);
3     $foto = $this->user_model->ambilFoto($where);
4     if($foto) {
5         if($foto != 'user.png') {
6             unlink('./assets/upload/pengguna/'.$foto.'');
7         }
8         $this->user_model->hapus_data($where, 'user');
9     }
10    $this->session->set_flashdata('Pesan','
11    <script>
12    $(document).ready(function() {
13        swal.fire({
14            title: "Berhasil dihapus!",
15            icon: "success",
16            confirmButtonColor: "#4e73df",
17        });
18    });
19    </script>
20    ');
21    redirect('user');
22 }
23
```



```

1 public function proses_tambah() {
2     $config['upload_path'] = './assets/upload/pengguna/';
3     $config['allowed_types'] = 'png|jpg|jpeg|gif|tif|tiff';
4     $this->load->library('upload', $config);
5
6     $kode = $this->user_model->buat_kode();
7     $namaL = $this->input->post('namaL');
8     $user = $this->input->post('user');
9     $notelp = $this->input->post('notelp');
10    $email = $this->input->post('email');
11    $level = $this->input->post('level');
12    $pass = $this->input->post('pwd');
13    $status = "Aktif";
14
15    $namaFile = $_FILES['photo']['name'];
16    if ($namaFile == '') {
17        $ganti = 'user.png';
18    } else {
19        if (!$this->upload->do_upload('photo')) {
20            redirect('user/tambah');
21        } else {
22            $data = array('photo' => $this->upload->data());
23            $nama_file = $data['photo']['file_name'];
24            $ganti = str_replace(" ", "_", $nama_file);
25        }
26    }
27
28    $data = array(
29        'id_user' => $kode,
30        'nama' => $namaL,
31        'username' => $user,
32        'notelp' => $notelp,
33        'email' => $email,
34        'level' => $level,
35        'password' => md5($pass),
36        'status' => $status,
37        'foto' => $ganti
38    );
39
40    $this->user_model->tambah_data($data, 'user');
41    $this->session->set_flashdata('Pesan', '
42    <script>
43    $(document).ready(function() {
44        swal.fire({
45            title: "Berhasil ditambah!",
46            icon: "success",
47            confirmButtonColor: "#4e73df",
48        });
49    });
50    </script>
51    ');
52    redirect('user');
53 }
54

```

```

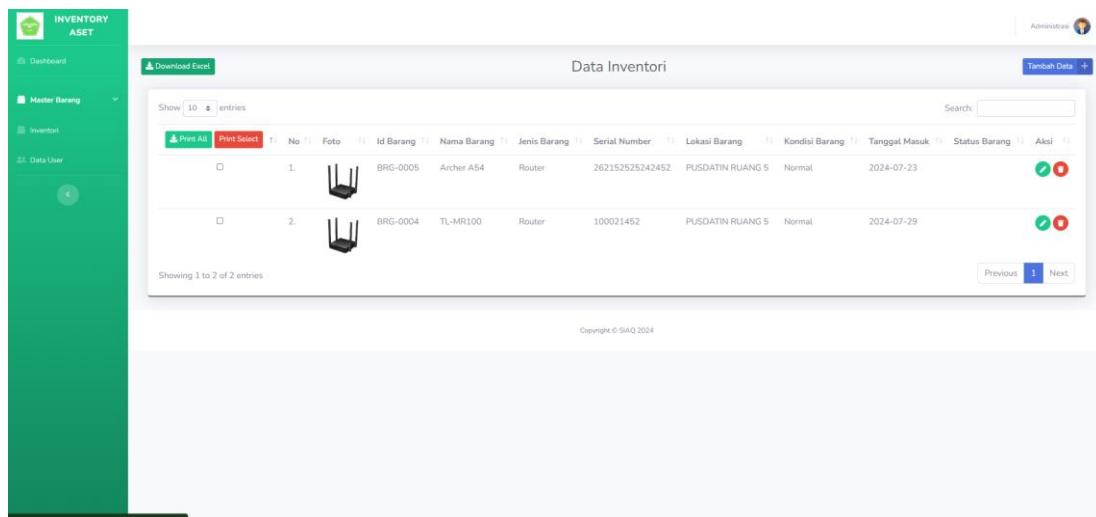
1 public function proses_ubah() {
2     $config['upload_path'] = './assets/upload/pengguna/';
3     $config['allowed_types'] = 'png|jpg|jpeg|gif|tif|tiff';
4     $this->load->library('upload', $config);
5
6     $kode = $this->input->post('iduser');
7     $namaL = $this->input->post('namaL');
8     $user = $this->input->post('user');
9     $notelp = $this->input->post('notelp');
10    $email = $this->input->post('email');
11    $level = $this->input->post('level');
12    $status = $this->input->post('status');
13    $pass = $this->input->post('pwd');
14    $passLama = $this->input->post('pwdLama');
15    $flama = $this->input->post('fotoLama');
16
17    $passUpdate = ($pass == '') ? $passLama : md5($pass);
18
19    $namaFile = $_FILES['photo']['name'];
20    if ($namaFile == '') {
21        $ganti = $flama;
22    } else {
23        if (!$this->upload->do_upload('photo')) {
24            redirect('user/ubah/'.$kode);
25        } else {
26            $data = array('photo' => $this->upload->data());
27            $nama_file = $data['photo']['file_name'];
28            $ganti = str_replace(" ", "_", $nama_file);
29            if ($flama != 'user.png') {
30                unlink('./assets/upload/pengguna/'.$flama.'');
31            }
32        }
33    }
34
35    $data = array(
36        'nama' => $namaL,
37        'username' => $user,
38        'notelp' => $notelp,
39        'email' => $email,
40        'level' => $level,
41        'password' => $passUpdate,
42        'status' => $status,
43        'foto' => $ganti
44    );
45
46    $where = array('id_user' => $kode);
47    $this->user_model->ubah_data($where, $data, 'user');
48    $this->session->set_flashdata('Pesan', '
49    <script>
50    $(document).ready(function() {
51        swal.fire({
52            title: "Berhasil diubah!",
53            icon: "success",
54            confirmButtonColor: "#4e73df",
55        });
56    });
57    </script>
58    ');
59    redirect('user');
60 }
61 }
62

```

Gambar II. 47 Kode Lengkap *Profile*

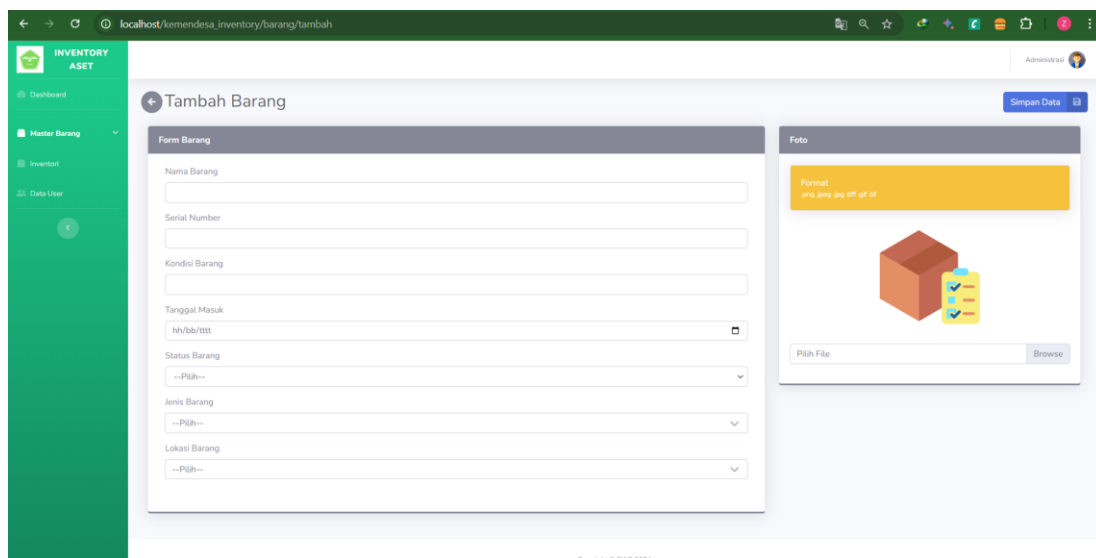
2.4 Rancangan Form Transaksi

A. Membuat Halaman Data Inventori Barang



Gambar II. 47 Halaman Data Inventori Barang

B. Membuat Halaman Data Form Tambah Barang



Gambar II. 48 Halaman Data Form Tambah Barang

1. Buat File Controller Barang

Buat file baru dengan nama `barang.php` dalam direktori `application/controllers/` pada proyek CodeIgniter Anda.

```

1  <?php
2  defined('BASEPATH') or exit('No direct script access allowed');
3
4  /**
5   * Author: Danish Yusrah, Adriansyah Putra, Winno Prasetya
6   */
7  class Barang extends CI_Controller
8  {
9
10     public function __construct()
11     {
12         parent::__construct();
13         // Memuat helper dan library yang diperlukan
14         $this->load->helper('url');
15         $this->load->helper('download');
16         $this->load->library('pagination');
17         $this->load->helper('cookie');
18         // Memuat model yang digunakan
19         $this->load->model('barang_model');
20         $this->load->model('jenis_model');
21         $this->load->model('satuan_model');
22         $this->load->model('laporan_model');
23     }
24
25     public function index()
26     {
27         // Mengatur judul halaman
28         $data['title'] = 'Barang';
29         // Mengambil data barang dari model untuk ditampilkan
30         $data['barang'] = $this->barang_model->dataJoin()->result();
31
32         $this->load->view('templates/header', $data);
33         $this->load->view('barang/index');
34         $this->load->view('templates/footer');
35     }
36
37     // Fungsi untuk men-generate barcode
38     public function create_barcode($kode)
39     {
40         // Memuat library Zend
41         $this->load->library('zend');
42         $this->zend->load('Zend/Barcode');
43
44         // Generate barcode menggunakan kode yang diberikan
45         Zend_Barcode::render('code128', 'image', array('text' => $kode), array());
46     }
47
48     // Fungsi untuk men-generate QR code
49     public function QRcode($kodenya = '123')
50     {
51         $this->load->library('Ciqrcode');
52         // Generate QR code menggunakan kode yang diberikan
53         Qrcode::png(
54             $kodenya,
55             $outfile = false,
56             $level = QR_ECLEVEL_H,
57             $size = 6,
58             $margin = 2
59         );
60     }
61
62     // Fungsi select digunakan untuk proses cetak barang yang dipilih
63     public function select()
64     {
65         // Memproses data yang di-submit untuk mencetak barang yang dipilih
66         if (isset($_POST['printBrgBtn'])) {
67             if (!empty($this->input->post('checkbox_value'))) {
68                 $barang = $this->input->post('checkbox_value');
69                 $checked_id = [];
70                 foreach ($barang as $b) {
71                     array_push($checked_id, $b);
72                 }
73                 $data['barang'] = $this->barang_model->printSelectedBrg($checked_id, 'barang')->result();
74                 $this->load->view('cetak/print3', $data);
75             } else {
76                 // Jika tidak ada barang yang dipilih, tampilkan pesan dan redirect
77                 $this->session->set_flashdata('Pesan', '');
78                 <script>
79                 $(document).ready(function() {
80                     swal.fire({
81                         title: "Pilih Barang Dahulu!",
82                         confirmButtonColor: "#4e73df",
83                     });
84                 });
85             </script>
86             ');
87             redirect('barang');
88         }
89     }
90 }
91
92 }
93

```

Gambar II. 49 Controller Barang

2. Buat Model Barang

Buat file model-model yang diperlukan untuk controller Barang. Anda perlu membuat model `Barang_model`, `Jenis_model`, `Satuan_model`, dan `Laporan_model` sesuai dengan kebutuhan aplikasi Anda. Pastikan model-model ini berada di direktori `application/models/`. Contoh file **Barang_model.php**:

```

1 <?php
2 defined('BASEPATH') or exit('No direct script access allowed');
3
4 class Barang_model extends CI_Model
5 {
6     public function __construct()
7     {
8         parent::__construct();
9         $this->load->database();
10    }
11
12    // Implementasikan fungsi-fungsi yang diperlukan untuk model Barang_model sesuai kebutuhan aplikasi Anda
13    // Misalnya, fungsi dataJoin() untuk mengambil data barang dengan join tabel lain
14 }
15

```

Gambar II. 50 *Barang Model*

3. Buat File View

Buat file view untuk header, halaman index barang, dan footer sesuai dengan struktur yang Anda gunakan. Misalnya, dalam direktori `application/views/`, Anda dapat memiliki file seperti **header.php**, **index.php** (untuk halaman index barang), dan **footer.php**. Isi file `header.php` dan `footer.php` dengan kode yang sesuai untuk tampilan header dan footer halaman.

4. Konfigurasi Routes

Terakhir, pastikan untuk menyesuaikan konfigurasi routes di **application/config/routes.php** agar aplikasi dapat mengetahui rute mana yang harus diarahkan ke controller Barang dan Setelah langkah-langkah ini, Anda dapat mengakses halaman index barang dengan mengunjungi URL aplikasi Anda, misalnya **`http://localhost/nama_aplikasi/barang`**. Pastikan juga Anda

telah mengaktifkan modul url di file **application/config/autoload.php** agar fungsi-fungsi URL CodeIgniter seperti `base_url()` dapat digunakan tanpa harus memuat helper secara manual.



```

1  <?php
2  defined('BASEPATH') or exit('No direct script access allowed');
3
4  $route['default_controller'] = 'home/index';
5
6  $route['home'] = 'Home/index';
7  $route['profile'] = 'Profile/index';
8
9  //user & login
10 $route['user'] = 'User/index';
11 $route['login'] = 'Login/index';
12
13 //master
14 $route['jenis'] = 'Jenis/index';
15 $route['satuan'] = 'Lokasi/index';
16 $route['barang'] = 'Barang/index';
17
18
19
20
21
22
23 $route['(:any)'] = 'gagal/index/$1';
24 $route['404_override'] = 'Gagal/index';
25 $route['translate_uri_dashes'] = FALSE;
26

```

Gambar II. 51 Route

2.5 Rancangan Laporan

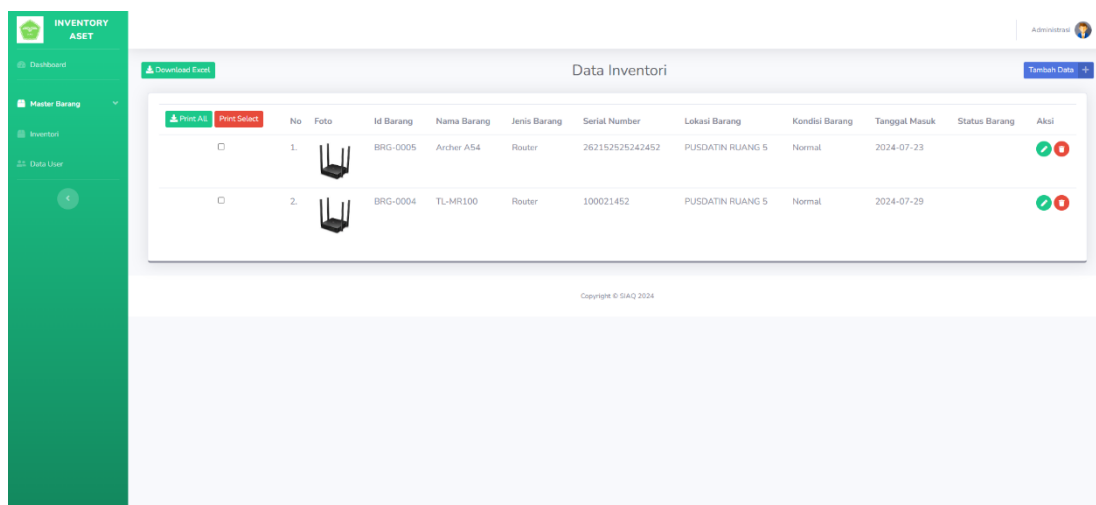
Rancangan laporan pada website inventori aset berbasis QR code melibatkan beberapa aspek yang penting. Pertama-tama, ada platform website yang digunakan untuk mengelola inventori aset secara keseluruhan. Di dalam platform ini, QR code menjadi kunci identifikasi yang unik untuk setiap aset.

Saat QR code dipindai, sistem dapat mengakses informasi lengkap mengenai aset tersebut dari database inventori.

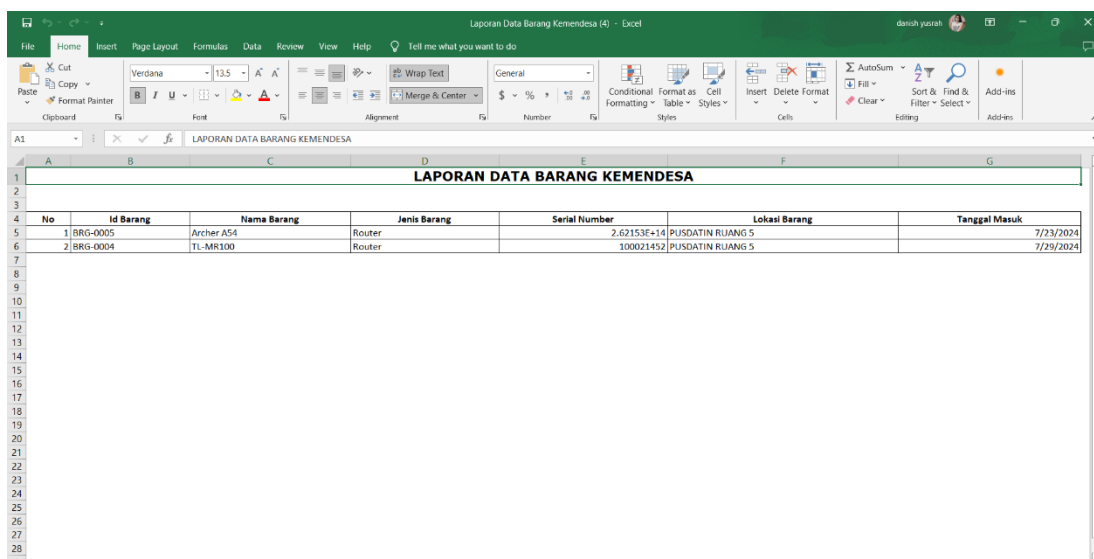
Fitur laporan ekspor memungkinkan pengguna untuk menghasilkan laporan berdasarkan kriteria tertentu, seperti rentang tanggal atau jenis aset. Setelah kriteria dipilih, sistem akan mengumpulkan data yang sesuai dari database. Informasi yang umumnya disertakan dalam laporan Excel termasuk detail aset seperti nama, deskripsi, tanggal perolehan, status, dan lokasi.

Proses eksportasi laporan memungkinkan pengguna untuk mengunduh laporan dalam format Excel untuk keperluan pengelolaan atau pelaporan lebih lanjut. Ini memberikan kemudahan dalam mengakses informasi inventori dengan format yang mudah dibaca dan dikelola. Integrasi dengan database inventori memastikan bahwa data yang ditampilkan dalam laporan selalu terkini dan akurat sesuai dengan informasi yang tersimpan dalam sistem inventori.

A. Membuat Form Laporan



Gambar II. 51 Form Laporan Barang

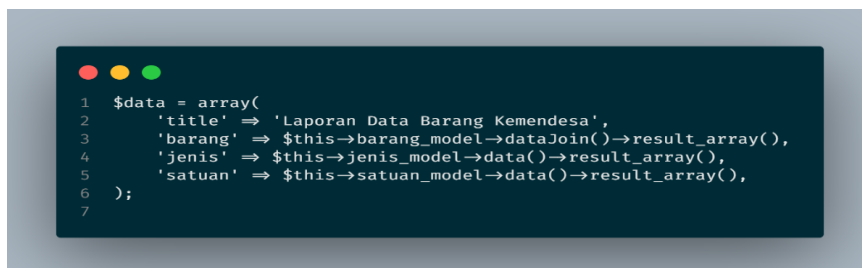


Gambar II. 52 Hasil Cetak Laporan Barang

1. Fungsi excel()

Ini adalah sebuah method yang berada dalam sebuah controller pada aplikasi web yang menggunakan framework PHP (seperti CodeIgniter). Nama method ini menunjukkan bahwa fungsinya adalah untuk menghasilkan file Excel.

2. Pembuatan Array Data



```

1 $data = array(
2     'title' => 'Laporan Data Barang Kemendesa',
3     'barang' => $this->barang_model->dataJoin()->result_array(),
4     'jenis' => $this->jenis_model->data()->result_array(),
5     'satuan' => $this->satuan_model->data()->result_array(),
6 );
7

```

Gambar II. 53 *Array Laporan Barang*

Di sini, sebuah array `\$data` dibuat untuk mengemas data yang akan dipassing ke view (tampilan).

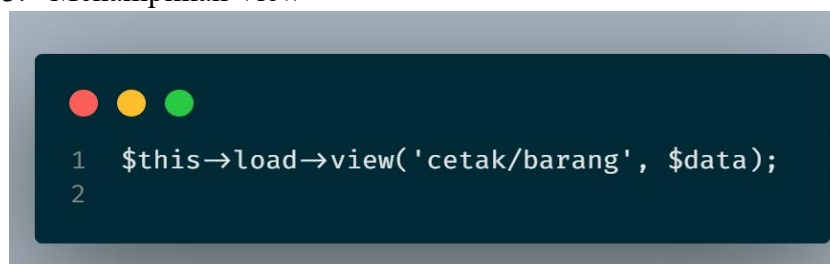
'title' => 'Laporan Data Barang Kemendesa': Ini adalah judul yang akan digunakan pada laporan.

'barang' => `\$this->barang\_model->dataJoin()->result\_array()` : Mengambil data barang dari model `barang\_model` dengan menggunakan method `dataJoin()` yang kemudian diubah menjadi array hasil (`result\_array()`).

`jenis' => `\$this->jenis_model->data()->result_array()` : Mengambil data jenis barang dari model `jenis_model` dan diubah menjadi array hasil.

`satuan' => `\$this->satuan_model->data()->result_array()` : Mengambil data satuan barang dari model `satuan_model` dan diubah menjadi array hasil.

3. Menampilkan View



```

1 $this->load->view('cetak/barang', $data);
2

```

Gambar II. 54 *Menampilkan View Cetak Barang*

Baris ini memanggil view dengan nama `cetak/barang` dan melewati data yang sudah disiapkan sebelumnya dalam array `\$data` ke dalam view tersebut.

2.6 Pengujian Unit

Table II. 9 Hasil Pengujian Black Box testing Halaman Login

No	Skenario Pengujian	Test case	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Username dan password harapan tidak diisi kemudian klik tombol <i>login</i>	Username: (kosong) Password : (kosong)	Sistem akan menolak akses user dan Menampilkan pesan “Username masih kosong!”	Sesuai harapan	valid
2	Mengetikan username diisi password tidak diisi atau kosong kemudia klik tombol <i>login</i>	Username : (admin) Password: (kosong)	Sistem akan menolak akses user dan menampilkan pesan “Password masih kosong!”	Sesuai harapan	Valid
3	Username tidak diisi atau kosong dan password diisi kemudian klik tombol <i>login</i>	Username : (kosong) Password : (admin)	Sistem akan menolak akses user dan menampilkan pesan “Username masih kosong!”	Sesuai harapan	Valid
4	Mengetikan salah satu kondisi salah pada	Username : admin (Benar) Password :	Sistem akan menolak akses user dan menampilkan	Sesuai harapan	Valid

	username atau password kemudian klik tombol <i>login</i>	Yamin (salah)	“user dan password salah”		
5	Mengetikan username dan password dengan data yang benar kemudian klik tombol <i>login</i>	Username : admin (benar) Password : 123123 (benar)	Sistem menerima akses login dan kemudian langsung menampilkan halaman Dashboard admin	Sesuai harapan	Valid

Table II. 10 Hasil Pengujian *Black Box Testing* Halaman Form Tambah Data Inventori

No	Skenario Pengujian	Test case	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Id barang otomatis terisi Nama barang, serial number, kondisi barang, tanggal masuk, Jenis barang, lokasi barang Tidak diisi kemudian klik tombol simpan data	Id barang : BRG-005 Nama barang (kosong) Serial number (kosong) Kondisi barang (kosong) Tanggal masuk (kosong) Jenis barang (kosong) Lokasi barang (kosong) Pilih file untuk menambahkan foto barang	Sistem akan menolak menyimpan data dan menampilkan “nama barang, serial number ,kondisi barang ,tanggal masuk ,jenis barang, lokasi barang tidak boleh kosong!”	Sesuai harapan	Valid
2	Id Barang Otomatis terisi sendiri Nama barang diisi kemudian serial number, kondisi barang, tanggal masuk, tanggal masuk, Jenis barang, lokasi barang Tidak diisi lalu klik tombol simpan data	Id Barang : BRG-005 Nama barang (diisi) Serial number (kosong) Kondisi barang (kosong) Tanggal masuk (kosong) Jenis barang (kosong) Lokasi barang (kosong)	Sistem akan menolak menyimpan data dan menampilkan “serial number, kondisi barang, tanggal masuk, tanggal masuk, Jenis barang, lokasi barang Tidak boleh kosong!”	Sesuai harapan	Valid

		Pilih file untuk menambahkan foto barang			
3	Id Barang Otomatis terisi sendiri Nama barang & serial number diisi kondisi barang, tanggal masuk, tanggal masuk, Jenis barang, lokasi barang tidak diisi kemudian klik tombol simpan	Id Barang : BRG-005 Nama barang (diisi) Serial number (diisi) Kondisi barang (kosong) Tanggal masuk (kosong) Jenis barang (kosong) Lokasi barang (kosong) Pilih file untuk menambahkan foto barang	Sistem akan menolak menyimpan data dan menampilkan “kondisi barang, tanggal masuk, tanggal masuk, Jenis barang, lokasi barang tidak boleh kosong!”	Sesuai harapan	Valid
4	Id Barang Otomatis terisi sendiri Nama barang, serial number, kondisi barang, tanggal masuk, tanggal masuk, Jenis barang terisi , lokasi barang tidak diisi kemudian tekan tombol simpan	Id Barang : BRG-005 Nama barang (diisi) Serial number (diisi) Kondisi barang (diisi) Tanggal masuk (diisi) Jenis barang (diisi) Lokasi barang (kosong)	Sistem akan menolak menyimpan data dan menampilkan “lokasi barang tidak boleh kosong!”	Sesuai harapan	Valid

		Pilih file untuk menambahkan foto barang			
5	Id Barang Otomatis terisi Sendiri Nama barang, serial number, kondisi barang, tanggal masuk, tanggal masuk, Jenis barang terisi , lokasi barang diisi kemudian klik tombol simpan	Id Barang : BRG-005 Nama barang (diisi) Serial number (diisi) Kondisi barang (diisi) Tanggal masuk (diisi) Jenis barang (diisi) Lokasi barang (diisi) Pilih file untuk menambahkan foto barang	Sistem akan menyimpan data dan menampilkan “data produk tersimpan”	Sesuai harapan	Valid

Table II. 11 Hasil Pengujian *Black Box Testing* Halaman *Form* Tambah Lokasi Barang

No	Skenario Pengujian	Test case	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Lokasi barang, keterangan tidak diisi (kosong) kemudian klik tombol simpan	Lokasi barang (kosong) Keterangan (kosong)	Sistem akan menolak menyimpan data dan menampilkan pesan “Nama Lokasi wajib diisi”	sesuai harapan	valid
2	Lokasi barang diisi dan keterangan kosong kemudian klik tombol simpan	Lokasi barang (diisi) Keterangan (kosong)	Sistem akan menyimpan data Lokasi dan menampilkan pesan “Berhasil ditambahkan”	Sesuai harapan	Valid
3	Lokasi barang & keterangan diisi kemudian klik tombol simpan	Lokasi barang (Diisi) Keterangan (diisi)	Sistem akan menyimpan data Lokasi dan menampilkan pesan “Berhasil ditambahkan”	Sesuai harapan	Valid

Table II. 12 Hasil Pengujian *Black Box Testing* Halaman *Form* Jenis Barang

No	Skenario Pengujian	Test case	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Jika ada <i>inputan</i> yang diisi dan ada <i>inputan</i> yang tidak diisi kemudian tekan tombol simpan	Jenis barang (diisi) Keterangan (kosong)	Sistem akan menyimpan data Jenis dan menampilkan pesan “Berhasil ditambahkan”	Sesuai harapan	valid
2	Jika data yang ada <i>inputan</i> tidak diisi dan kemudian tekan tombol simpan	Jenis barang (kosong) Keterangan (kosong)	Sistem akan menolak menyimpan data dan menampilkan pesan “Nama Jenis wajib diisi”	Sesuai harapan	valid
3	Jika semua inputan diisi dan sesuai dengan validasi , kemudian klik tombol simpan	Jenis barang (diisi) Keterangan (diisi)	Sistem akan menyimpan data Jenis dan menampilkan pesan “Berhasil ditambahkan”	Sesuai harapan	Valid

Table II. 13 Hasil Pengujian *Black Box Testing* Halaman *Form* Tambah Data User

No	Skenario Pengujian	Test case	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Nama lengkap terisi mengosongkan salah satu field contoh email kosong sedangkan username ,password dan level diisi	Nama user : (Danish) Username: (kosong) Nomor Telepon: (kosong) Email: (kosong) Level: (kosong) Password: (kosong) Konfirmasi Password: (kosong)	Sistem akan menolak Akses user dan menampilkan Pesan dari field yang kosong, contohnya : “Username wajib diisi”	Sesuai Harapan	Valid
2	Semua field terisi	Nama user : (Danish) Username: (Danish) Nomor Telepon: (087873220852) Email: (danish@gmail.com) Level: (admin) Password: (123123) Konfirmasi Password: (123123)	Sistem akan menerima Akses user Dan Menampilkan “Berhasil disimpan”	Sesuai harapan	Valid

Table II. 14 Hasil Pengujian *Black Box Testing* Mencetak Data Barang

No	Skenario Pengujian	Test case	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Mencetak data barang	Menekan tombol donwload Excel	Menampilkan Hasil cetak data barang	Sesuai harapan	Valid

Table II. 15 Hasil Pengujian *Black Box Testing* Mengedit Profil

No	Skenario Pengujian	Test case	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Mengedit dan update profil	klik menu pengguna, klik icon edit pada pengguna yang akan diedit, isi form edit, klik update	berhasil mengedit dan update profil	Sesuai harapan	Valid
2	Mengosongkan Data Profile Sebelum Menekan Simpan	Nama lengkap : Kosong Email : Kosong Nomor telepon : Kosong Alamat : Kosong Foto profil : Kosong	Menolak Edit, muncul pesan bahwa input tidak boleh kosong	Sesuai harapan	Valid

Table II. 16 Hasil Pengujian *Black Box Testing* Halaman *logout*

No	Skenario Pengujian	Test case	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Mengklik button logout	Klik button logout	Akan kembali menampilkan login	Sesuai harapan	valid

Table II. 17 Hasil Pengujian *Black Box Testing Upload Foto*

No	Skenario Pengujian	Test case	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Mengupload foto dengan jenis file selain gambar (doc, xls, pdf, dsb)	Upload file dokumen word di kotak “Browse” pada bagian upload foto.	Gagal upload dan memunculkan pesan “Format bukan gambar”	Sesuai harapan	valid
2	Mengupload foto dengan jenis gambar (png, jpg, jpeg, dsb)	Upload file dokumen word di kotak “Browse” pada bagian upload foto.	Upload gambar berhasil dan tersimpan	Sesuai harapan	valid

BAB III

PENUTUP

3.1 Kesimpulan

Kesimpulan dari penulisan buku tutorial ini sebagai laporan Tugas Akhir adalah sebagai berikut:

1. Sistem informasi yang dibuat membantu kegiatan karyawan untuk mengkomputerisasi pencatatan aset yang masuk dan lokasi penempatannya.
2. Dengan adanya sistem inventori aset ini, memudahkan pegawai untuk mencari keberadaan lokasi aset yang digunakan.
3. Adanya fitur QRcode dapat memudahkan pegawai untuk mencari detail barang sesuai dengan kode barangnya dengan melakukan scan pada QRcode tersebut.
4. Pada fitur pencarian data pada sistem informasi ini akan membantu pegawai untuk menghemat waktu dalam proses pencarian barang.

3.2 Saran

Sistem informasi ini belum bisa dikatakan sempurna, terdapat beberapa saran yang diharapkan dapat menjadi perhatian untuk pengembangan kedepannya.

1. Pada aplikasi sistem inventori aset berbasis QRcode ini hanya dikembangkan dengan framework codeigniter 3, sehingga dapat dikembangkan lagi dengan bahasa pemrograman lain sesuai kemajuan teknologi.
2. Dalam mengembangkan sistem alangkah baiknya untuk dapat menganalisa secara mendetail sesuai dengan kebutuhan terlebih dahulu sehingga sistem yang dibangun sesuai dengan spesifikasi dan mudah melakukan pemeliharaan (*maintenance*) kedepan.
3. Aplikasi sistem inventori aset ini masih dapat dikembangkan lagi menjadi sistem yang lebih kompleks sesuai kebutuhan pengguna.

DAFTAR PUSTAKA

- Aliman, W. (2021). Perancangan perangkat lunak untuk menggambar diagram berbasis android. *Syntax Literate; Jurnal Ilmiah Indonesia*, 6(6), 3091–3098.
- Arianti, T., Fa'izi, A., Adam, S., & Wulandari, M. (2022). Perancangan Sistem Informasi Perpustakaan Menggunakan Diagram UML (Unified Modelling Language). *Jurnal Ilmiah Komputer Terapan Dan Informasi*, 1(1), 19–25.
- Farahat, A. F., Dewi Ambarwati, A., & Lestari, A. (2022). *Pengaruh E-Commerce, E-Crm, Dan Database Terhadap Sistem Informasi Penjualan (Literature Sistem Informasi Manajemen)*. 1(1). <https://doi.org/10.38035/jim.v1i1>
- Ikhsan, M., Akbar, Z., Dani, R., & Ediansa, O. (2024). Sosialisasi dan Pelatihan Framework Codeigniter Untuk Mahasiswa Universitas Muhammadiyah Jambi. *ASPIRASI: Publikasi Hasil Pengabdian Dan Kegiatan Masyarakat*, 2(1), 70–76.
- Indarta, Y., Irfan, D., Muksir, M., Simatupang, W., & Ranuharja, F. (2021). Analisis dan Perancangan Database Menggunakan Model Konseptual Data Warehouse Sistem Manajemen Transaksi Toko Online Haransaf. *Edukatif: Jurnal Ilmu Pendidikan*, 3(6), 4448–4455.
- Istiawan, N., & Nuralia, N. (2021). Perancangan Sistem Informasi Manajemen Koleksi Museum Berbasis Web (Studi Kasus Museum Negeri Provinsi Lampung). *Jurnal Informatika Dan Rekayasa Perangkat Lunak*, 2(1), 102–109.
- LAUNDRY, P. A. (n.d.). *PEMODELAN UML PENDAPATAN*.
- Limbong, T., & Sianipar, J. F. (2022). *Sistem Informasi Inventori Aset dan Logistik Rumah Sakit Bina Kasih Berbasis Website dengan Menggunakan Metode Rapid Application Development (RAD)* (Vol. 04, Issue 02).
- Prakoso, R. D. (2018). Implementasi Dan Perbandingan Performa Proxmox Dalam Virtualisasi Dengan Tiga Virtual Server. *J. Manaj. Inform*, 8(1), 79–86.
- Sama, H., & Eric, E. (2023). Perancangan dan Implementasi Website Sistem Inventori di SMK Globe National Plus 2 Kota Batam. *National Conference for Community Service Project (NaCosPro)*, 5(1), 942–951.
- Syahputra, A. K., & Kom, M. (2018). Search Engine & Web Browser. *Internet Dan Website*, 1–23.
- Tanari, A., Handojo, A., & Andjarwirawan, J. (2019). Aplikasi Pencarian Jurnal Ilmiah Dengan Term Frequency-Inverse Document Frequency. *Jurnal Infra*, 7(1), 46–50.

BIOGRAFI PENULIS

DANISH YUSRAH

Saya, Danish Yusrah, lahir pada tanggal 6 Oktober 2002 di kota Manggar, sebuah kota kecil yang terletak di Pulau Belitung, Indonesia. Sejak kecil, saya menunjukkan ketertarikan yang besar terhadap dunia teknologi dan ilmu pengetahuan. Kecintaan saya pada bidang ini terus berkembang seiring bertambahnya usia dan pengalaman di dunia pendidikan.

Saya memulai pendidikan formal di SDN 8 Manggar. Selama enam tahun belajar di sana, saya dikenal sebagai siswa yang cerdas dan tekun. Saya sering kali menunjukkan minat yang mendalam terhadap pelajaran-pelajaran yang berkaitan dengan ilmu pengetahuan dan teknologi, seperti matematika dan ilmu komputer dasar. Bakat dan minat saya dalam bidang IT mulai tampak sejak dini, saat saya sering membantu teman-teman mengatasi masalah teknis dengan komputer sekolah.

Setelah menyelesaikan pendidikan di tingkat dasar, saya melanjutkan pendidikan di SMPN 2 Manggar. Di sini, minat saya terhadap teknologi semakin berkembang. Saya bergabung dengan komunitas komputer sekolah dan aktif mengikuti berbagai kegiatan yang berhubungan dengan IT.

Selepas lulus dari SMP, saya kemudian melanjutkan pendidikan ke SMAN 1 Manggar. Masa-masa di sekolah menengah atas ini menjadi periode yang sangat penting dalam hidup saya. Saya memilih jurusan IPS agar bisa lebih fokus mendalami mata pelajaran yang mendukung minat saya, seperti ekonomi, geografi, dan sosiologi, namun tetap melanjutkan eksplorasi pribadi dalam bidang teknologi informasi. Di SMAN 1 Manggar, saya tidak hanya aktif dalam kegiatan akademik, tetapi juga ikut serta dalam berbagai organisasi yang berkaitan dengan IT, seperti komunitas komputer.

Kegigihan saya dalam mengejar ilmu pengetahuan di bidang IT tidak hanya terbatas di sekolah. Di waktu luang, saya sering mengikuti kursus online, membaca buku-buku terkait teknologi, dan mengerjakan berbagai aktivitas yang memperkaya pengetahuan saya.

Saat ini, saya sedang menempuh pendidikan D3 di Universitas Bina Sarana Informatika (UBSI). Pilihan ini saya ambil untuk lebih mengembangkan keterampilan teknis dan pengetahuan saya dalam bidang teknologi informasi. Di UBSI, saya

mendapatkan banyak kesempatan untuk belajar dan berinteraksi dengan dosen dan mahasiswa lain yang memiliki minat yang sama, sehingga semakin memperkaya pengalaman akademis dan praktis saya.

Ketertarikan saya pada bidang IT bukan hanya tentang memahami cara kerja teknologi, tetapi juga tentang bagaimana teknologi dapat digunakan untuk memecahkan masalah sehari-hari dan meningkatkan kualitas hidup. Saya bercita-cita untuk melanjutkan studi ke jenjang yang lebih tinggi, khususnya di bidang teknologi informasi atau ilmu komputer, dengan harapan dapat berkontribusi dalam pengembangan teknologi di masa depan.

Dengan semangat dan dedikasi yang saya miliki, saya terus berupaya untuk mencapai impian dan aspirasi, serta bertekad untuk membawa perubahan positif melalui teknologi. Saya percaya bahwa dengan kerja keras dan komitmen, saya dapat mencapai tujuan tersebut dan memberikan dampak yang signifikan dalam dunia teknologi.

WINNO PRASETYA RAMADHAN

Saya, Winno Prasetya Ramadhan, lahir pada tanggal 2 Desember 2002 di Jakarta, Indonesia. Sejak kecil, saya dikenal sebagai anak yang penuh semangat dan memiliki rasa ingin tahu yang tinggi, terutama dalam bidang teknologi dan ilmu pengetahuan. Perjalanan pendidikan saya dimulai di SDN Grogol Utara 05 Pagi, di mana saya menunjukkan minat yang besar terhadap pelajaran sains dan komputer.

Selama enam tahun di SDN Grogol Utara 05 Pagi, saya dikenal sebagai siswa yang aktif dan rajin. Saya seringkali membantu guru dalam mengoperasikan perangkat teknologi yang ada di sekolah, seperti komputer dan proyektor. Ketertarikan saya pada bidang teknologi sudah terlihat sejak dini, dan hal ini terus berkembang seiring berjalannya waktu.

Setelah menyelesaikan pendidikan dasar, saya melanjutkan ke SMPN 271 Jakarta. Di sekolah menengah pertama ini, minat saya terhadap teknologi semakin mendalam. Saya bergabung dengan komunitas komputer sekolah dan aktif mengikuti berbagai kegiatan yang berhubungan dengan teknologi informasi. Saya juga sering memanfaatkan perpustakaan sekolah untuk membaca buku-buku tentang teknologi dan komputer, yang semakin memperkaya pengetahuan saya.

Lulus dari SMP, saya kemudian melanjutkan pendidikan ke SMK Widyapatria 2. Memilih jurusan yang berfokus pada teknologi, saya merasa bahwa SMK adalah pilihan yang tepat untuk mengembangkan keterampilan saya secara lebih praktis dan mendalam. Di SMK Widyapatria 2, saya tidak hanya belajar tentang teori, tetapi juga banyak terlibat dalam praktik langsung yang sangat berguna untuk masa depan saya. Saya sering mengikuti workshop dan seminar tentang teknologi informasi yang diadakan di sekolah maupun di luar sekolah.

Setelah lulus dari SMK, saya memutuskan untuk melanjutkan pendidikan ke jenjang yang lebih tinggi di Universitas Bina Sarana Informatika (UBSI). Saat ini, saya sedang menempuh pendidikan di UBSI dengan jurusan sistem informasi, di mana saya terus mengasah keterampilan teknis dan pengetahuan saya dalam bidang teknologi informasi. Di sini, saya aktif dalam berbagai kegiatan akademik dan organisasi, yang semuanya membantu saya untuk lebih mempersiapkan diri dalam mencapai tujuan karier saya di masa depan.

Ketertarikan saya pada teknologi bukan hanya sekadar hobi, tetapi juga sebagai sarana untuk memecahkan masalah dan menciptakan solusi yang berguna bagi masyarakat. Saya bercita-cita untuk berkontribusi dalam pengembangan teknologi informasi yang inovatif dan berkelanjutan. Dengan semangat dan dedikasi yang saya miliki, saya terus berupaya untuk mencapai impian dan aspirasi saya. Saya percaya bahwa dengan kerja keras dan komitmen, saya dapat memberikan dampak yang positif dalam dunia teknologi dan ilmu pengetahuan.

ADRIYANSYAH PUTRA

Saya, Adriyansyah Putra, lahir pada tanggal 26 Mei 1997 di Jakarta, Indonesia. Sejak kecil, saya telah menunjukkan minat yang kuat dalam bidang ilmu pengetahuan dan teknologi. Pendidikan saya dimulai di SDN 01 Ulujami, di mana saya aktif belajar dan menunjukkan prestasi yang baik dalam berbagai mata pelajaran.

Setelah menyelesaikan pendidikan dasar, saya melanjutkan ke SMPN 177 Jakarta. Di sekolah menengah pertama ini, saya semakin tertarik pada ilmu sosial dan humaniora, sambil tetap mempertahankan ketertarikan saya dalam teknologi dan komputer. Saya aktif dalam berbagai kegiatan sekolah dan mengembangkan keterampilan sosial serta kepemimpinan.

Setelah lulus dari SMP, saya melanjutkan pendidikan ke SMAN 63 Jakarta. Di sekolah menengah atas ini, saya memilih jurusan IPA untuk mendalami ilmu pengetahuan yang lebih mendalam. Saya aktif dalam klub sains sekolah dan berpartisipasi dalam berbagai kompetisi akademik yang memperluas wawasan dan pengetahuan saya.

Setelah lulus dari SMA, saya memutuskan untuk melanjutkan pendidikan ke jenjang yang lebih tinggi di Universitas Bina Sarana Informatika (UBSI). Saat ini, saya sedang menempuh pendidikan di UBSI dengan jurusan Sistem Informasi. Di universitas ini, saya belajar tentang teknologi informasi, pengembangan perangkat lunak, sistem informasi, dan berbagai mata kuliah lainnya yang relevan. Saya aktif dalam kegiatan akademik dan organisasi kampus, yang membantu saya mengembangkan keterampilan yang dibutuhkan dalam industri teknologi informasi.

Selama di UBSI, saya juga aktif dalam proyek-proyek pengembangan perangkat lunak dan sistem informasi yang bertujuan untuk memberikan solusi terbaik dalam dunia bisnis dan industri. Saya memiliki aspirasi untuk berkarier di bidang teknologi informasi, dengan tujuan untuk menggabungkan pengetahuan dan keterampilan saya dalam mengembangkan solusi IT yang inovatif dan berkelanjutan.

Proyek Sistem Informasi :

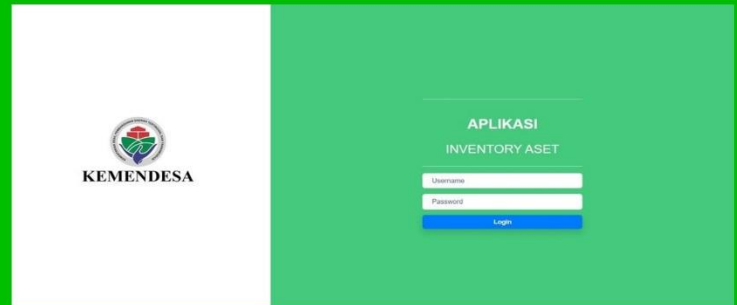
SIAQ



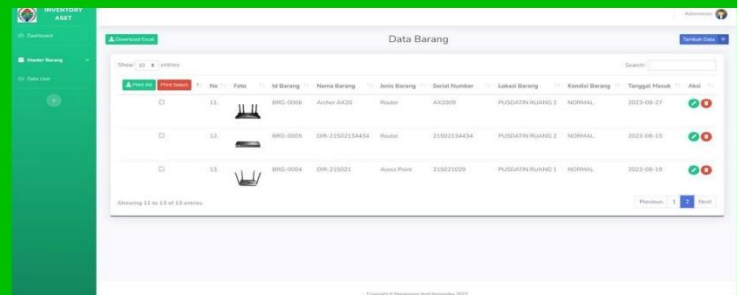
Sistem Inventori Aset Berbasis QRCODE

Kementrian Desa Pembangunan Daerah Tertinggal dan Transmigrasi

Inventori aset memegang peranan penting dalam menunjang pengelolaan aset yang dimiliki. Perusahaan atau suatu instansi harus bisa mengelola hal ini dengan baik, sehingga pada praktiknya proses pencatatan yang dilakukan di perusahaan atau instansi tersebut yang masih menggunakan sistem manual, mulai dari memproses permintaan material, memproses barang masuk, kemudian melakukan tracking barang keluar dan pembuatan laporan oleh administrator. Perusahaan yang tidak mengelola persediaannya dengan baik, pencatatan informasi persediaan yang tidak lengkap, laporan penerimaan atau pengeluaran barang, barangsering tertahan di gudang pusat, dan pembuatan laporan persediaan membutuhkan waktu yang relatif lama antara data produk di pusat dan data produk di tengah. Elemen data cabang. Untuk itu aplikasi SIAQ ini mengadopsi metodologi air terjun dalam membangun aplikasi ini dengan mengusulkan pendekatan pengembangan perangkat lunak yang sistematis dan berurutan mulai dari level dan perkembangan sistem pada saat analisis , desain, kode, pengujian dan pemeliharaan. Aplikasi SIAQ ini merupakan solusi untuk mempercepat sistem inventarisasi aset Anda yang sudah ada. Untuk kemudahan akses kapan saja. Hal ini tidak hanya membuat proses inventarisasi aset menjadi lebih efektif dan efisien, namun juga memudahkan petugas logistik dalam melakukan proses inventarisasi aset di kantor pusat dan kantor cabang.



Halaman Login



Halaman Dashboard



Halaman Print QRCODE



**PRODI SISTEM INFORMASI
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BINA SARANA INFORMATIKA
JAKARTA
2024**