

## OPTIMALISASI INVENTORI DAN PENJUALAN: PENDEKATAN ANALISIS DAN PERANCANGAN BERBASIS WEB

William Surya Admadja<sup>1</sup>, Riki<sup>2</sup>  
Universitas Buddhi Dharma Tangerang  
[Williamsuryaadmadja@gmail.com](mailto:Williamsuryaadmadja@gmail.com)

**Abstract**

*This research aims to design and develop an inventory management application with real-time stock monitoring through a prototyping approach. Efficient and effective inventory management is an important asset for an organization or company, which can improve performance and productivity. The designed web-based application allows real-time inventory access and management, making it easier to manage and monitor stock. The design process involves analyzing user needs, evaluating existing inventory management systems, designing user interfaces, developing databases, creating prototypes, and testing applications. The use of web technology allows access from a variety of devices, including desktop computers, tablets, and browser. The result of this research inventory management application that includes features such as recording incoming and outgoing goods, stock updates, transaction tracking, and creating inventory reports. The implementation of this system is expected to increase efficiency in inventory management, reduce the risk of losing goods, and provide accurate and fast information for stakeholders.*

**Article History**

Submitted: 3 Januari 2025

Accepted: 8 Januari 2025

Published: 9 Januari 2025

**Key Words**

Prototyping Design, Inventory Management Applications, Real-Time Stock Monitoring, Databases, Web Based

**Abstrak**

Penelitian ini bertujuan untuk merancang dan mengembangkan aplikasi pengelolaan inventori dengan pemantauan stok *real-time* melalui pendekatan prototyping. Pengelolaan inventori yang efisien dan efektif merupakan aset penting bagi organisasi atau perusahaan, yang dapat meningkatkan kinerja dan produktivitas. Aplikasi berbasis *web* yang dirancang memungkinkan akses dan manajemen inventori secara *real-time*, memudahkan pengelolaan dan pengawasan stok. Proses desain melibatkan analisis kebutuhan pengguna, evaluasi sistem pengelolaan inventori yang ada, perancangan antarmuka pengguna, pengembangan basis data, pembuatan prototipe, serta pengujian aplikasi. Penggunaan teknologi *web* memungkinkan akses dari berbagai perangkat, termasuk komputer *desktop*, tablet, dan *browser*. Hasil dari penelitian ini adalah sebuah aplikasi pengelolaan inventori yang mencakup fitur-fitur seperti pencatatan barang masuk dan keluar, pembaruan stok, pelacakan transaksi, dan pembuatan laporan inventori. Implementasi sistem ini diharapkan dapat meningkatkan efisiensi dalam pengelolaan inventori, mengurangi risiko kehilangan barang, dan menyediakan informasi yang akurat dan cepat bagi para pemangku kepentingan.

**Sejarah Artikel**

Submitted: 3 Januari 2025

Accepted: 8 Januari 2025

Published: 9 Januari 2025

**Kata Kunci**

Desain Prototyping, Aplikasi Pengelolaan Inventori, Pemantauan Stok Real-Time, Basis Data, Berbasis Web.

**PENDAHULUAN**

Manajemen inventori adalah bagian integral dari operasi sukses suatu organisasi, termasuk perusahaan, institusi pemerintah, dan organisasi nirlaba. Inventori meliputi semua barang, perlengkapan, atau aset yang dimiliki oleh organisasi dan digunakan dalam proses bisnis atau layanan yang mereka tawarkan. Manajemen inventori mencakup pengadaan, pengelolaan, pengendalian, dan pemantauan inventori ini selama siklus hidupnya. Dengan perkembangan teknologi informasi, sistem informasi manajemen inventori telah menjadi sangat penting untuk memastikan efisiensi dan akurasi dalam pengelolaan aset organisasi.

Proses manajemen inventori yang baik harus didasarkan pada hasil dari analisis yang telah dilakukan. Analisis manajemen sistem adalah proses memilah-milah suatu permasalahan menjadi bagian-bagian yang lebih kecil untuk dipelajari guna mempermudah pemecahan permasalahan dari suatu sistem informasi. Pada tahap analisis, diperlukan suatu pendekatan yang tepat guna menghindari kesalahan-kesalahan yang mungkin muncul pada tahap perancangan dan implementasi, karena tahap ini merupakan tahapan yang sangat penting.

Manajemen inventori yang efektif sangat penting untuk menjaga operasional dan kesehatan finansial suatu organisasi. Ketidakmampuan dalam mengelola manajemen dapat menyebabkan masalah seperti kelangkaan persediaan yang menghambat kelancaran operasional atau kelebihan stok yang memberatkan keuangan. Melalui pemantauan yang cermat terhadap manajemen, organisasi dapat menghindari potensi kerugian finansial, meningkatkan efisiensi pengelolaan persediaan, mengurangi biaya penyimpanan, dan memastikan pemenuhan permintaan konsumen tepat waktu (Dwi Agustina Cahyaningrum et al., 2023).

Di era *digital* saat ini, perkembangan teknologi informasi telah memberikan peluang besar untuk meningkatkan efisiensi dan efektivitas manajemen inventori. Sistem manajemen inventori yang didukung oleh teknologi dapat memberikan banyak keuntungan, mulai dari pemantauan *real-time*, analisis data yang akurat dll (Eid Bait Bin Saleem & Ullah, 2023). Namun, implementasi sistem ini tidak selalu mudah dan sering kali menimbulkan beberapa tantangan, seperti biaya implementasi yang tinggi, kesulitan dalam integrasi dengan sistem yang sudah ada, dan resistensi dari pihak-pihak yang terlibat.

Oleh karena itu, penelitian yang bertujuan untuk merancang dan mengoptimalkan sistem manajemen inventori menjadi sangat relevan dan penting (Komala Sari & Isnaini, 2021). Dengan mengidentifikasi kebutuhan dan tantangan yang dihadapi oleh organisasi dalam mengelola manajemen mereka, serta memanfaatkan potensi teknologi informasi yang ada, diharapkan dapat diciptakan sebuah sistem manajemen inventori yang efisien, terintegrasi, dan dapat memberikan nilai tambah yang signifikan bagi organisasi.

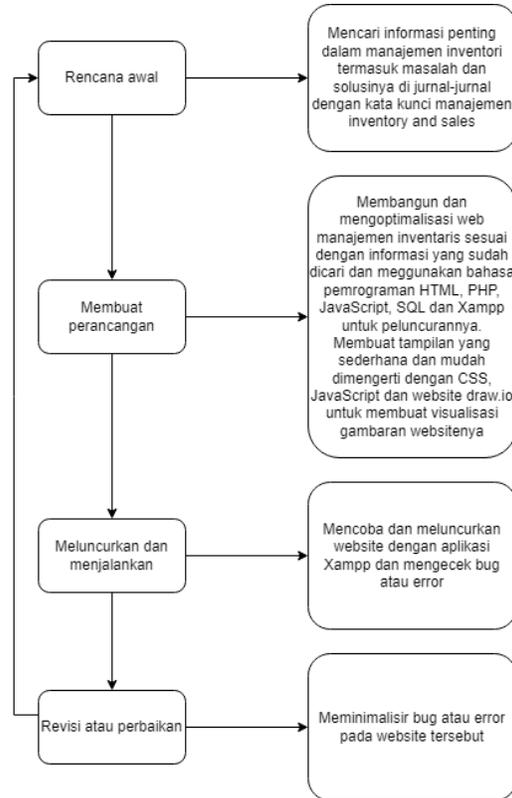
Penelitian ini akan fokus pada perancangan *prototyping* untuk aplikasi pengelolaan inventori dengan pemantauan stok *real-time*. Tujuan utamanya adalah untuk merancang dan mengembangkan aplikasi yang memungkinkan pengelolaan inventori secara *real-time*, memudahkan pengawasan stok, dan meningkatkan efisiensi operasional. Melalui penelitian ini, diharapkan dapat dibentuk solusi-solusi yang dapat membantu organisasi dalam mengatasi tantangan dalam mengelola inventori mereka, serta meningkatkan kinerja dan daya saing organisasi di pasar yang semakin kompetitif.

Kesulitan dalam mengambil informasi yang telah disimpan dan akhirnya dapat menghambat langkah berikutnya dalam alur kerja merupakan masalah umum yang dihadapi oleh banyak organisasi. Oleh karena itu, inventori harus dikelola menggunakan sistem manajemen yang baik dan benar. Dengan berpedoman pada latar belakang tersebut, penelitian ini berjudul "Optimalisasi Inventori dan Penjualan: Pendekatan Analisis dan Perancangan Berbasis Web" diharapkan dapat memberikan kontribusi signifikan dalam bidang manajemen inventori berbasis teknologi.

## METODE PENELITIAN

### Kerangka Penelitian

Kerangka penelitian ini menggunakan landasan yang mirip dengan pendekatan *prototipe* yang mengatur struktur dan arah penelitian ini. Ini mencakup tahapan-tahapn penting yang membantu mengoptimalisasi dan merancang aplikasi manajemen inventori berbasis *web* ini



**Gambar 3.1 Kerangka Penelitian**

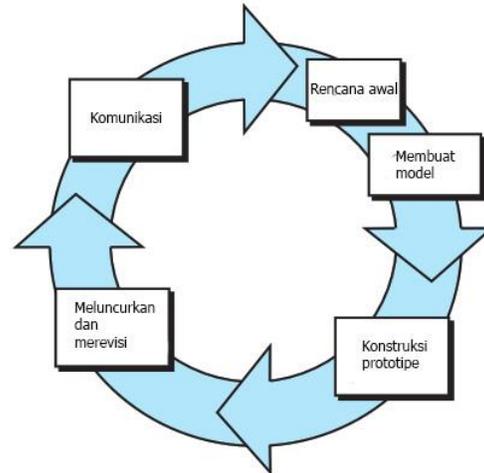
### .Metode Pendekatan

**Metode prototipe** adalah pendekatan dalam pengembangan sistem di mana model awal (prototipe) dari sistem atau produk dibuat, diuji, dan diperbaiki melalui iterasi berulang sebelum menghasilkan produk akhir. Tujuan utama dari metode ini adalah untuk memahami kebutuhan pengguna dan mengeksplorasi ide-ide desain dengan cepat, sehingga meminimalkan risiko pengembangan dan memastikan bahwa produk akhir memenuhi kebutuhan dan harapan pengguna.

Menggunakan metode prototipe dalam pengembangan *website* manajemen inventori memiliki beberapa keuntungan yang signifikan. Pertama, metode ini memungkinkan tim pengembang untuk secara cepat merancang dan mengimplementasikan versi awal dari website manajemen inventori. Dengan adanya prototipe yang dapat diuji secara langsung oleh pengguna dan pemangku kepentingan, tim dapat mengidentifikasi kebutuhan dan persyaratan yang sebenarnya lebih tepat waktu.

Kedua, prototipe memungkinkan pengguna untuk melihat dan merasakan bagaimana website akan berfungsi dalam kehidupan nyata sebelum pengembangan penuh dimulai. Ini memungkinkan untuk pengumpulan umpan balik yang lebih baik, yang mana bisa mengarah pada peningkatan desain dan fungsi. Berikut adalah penjelasan singkat untuk setiap tahapan dalam metode prototipe:





**Gambar 3.2 Pendekatan Prototype**

1. **Rencana awal** adalah tahap awal di mana tujuan dan lingkup proyek ditentukan secara kasar. Tujuan dari tahap ini adalah untuk mendefinisikan ruang lingkup umum proyek, menentukan kebutuhan utama, serta merencanakan pendekatan dan sumber daya yang dibutuhkan untuk mengembangkan prototipe. Aktivitas yang dilakukan dalam tahap ini antara lain:
  - a. Identifikasi masalah atau kebutuhan yang harus diselesaikan.
  - b. Pengidentifikasian stakeholder dan pemangku kepentingan yang terlibat.
  - c. Penentuan fokus utama prototipe dan fitur-fitur kunci yang akan dikembangkan.

*Website* manajemen stok harus memiliki fitur-fitur penting seperti *dashboard* yang intuitif untuk memberikan gambaran umum tentang status stok secara *real-time*. Fitur pelacakan kapan barang masuk dan keluar yang memungkinkan pengguna mencatat dan memantau aliran barang secara akurat juga sangat penting. Selain itu, sistem pemberitahuan otomatis untuk stok yang baru dimasukkan. Fungsi manajemen data produk, termasuk kategori, dan harga, mempermudah pengelolaan inventori. Fitur laporan penjualan tertinggi dan penjualan terbaru juga diperlukan untuk memberikan wawasan mendalam melalui data tentang tren penggunaan dan perputaran stok.
2. **Membuat Model** melibatkan pengembangan desain prototipe yang cepat berdasarkan pada kebutuhan dan lingkup yang telah ditetapkan. Aktivitas dalam tahap ini mencakup:
  - a. Pembuatan sketsa atau mockup awal antarmuka pengguna (*UI/UX design*).
  - b. Spesifikasi sederhana mengenai fungsionalitas dasar yang akan dimasukkan dalam prototipe.
  - c. Pemodelan struktur data atau aliran kerja (*workflow*) dasar yang akan diimplementasikan.

**Login Panel**

Username

Password

**Gambar 3.3.1 Sketsa Login**

Sketsa ini menunjukkan tampilan login dalam sistem pengelolaan inventori Jika Id Pengguna / Primary Key ( id) (int,11) dan name (name) (Varchar,60) dan Password (password) (Varchar,225) dan Tingkatan Pengguna Sistem / Foreign Key (User.level) (Int,11) dan Tingkatan Pengguna Sistem / Foreign Key (Image) (Varchar,255) dan Status Pengguna (Status) (Int,1) dan Waktu Terakhir Pengguna Login (Last\_login) (Datetime).

**Sistem** Tanggal Pengguna

**Dashboard**  
 Pengaturan pengguna  
 Kategori  
 Produk  
 Media gambar  
 Penjualan  
 Laporan penjualan

Jumlah Pengguna

Jumlah Kategori

Jumlah Produk

Jumlah Penjualan

Produk penjualan tertinggi

Judul	Total terjual	Total jumlah

Penjualan terbaru

No.	Nama produk	Tanggal	Total

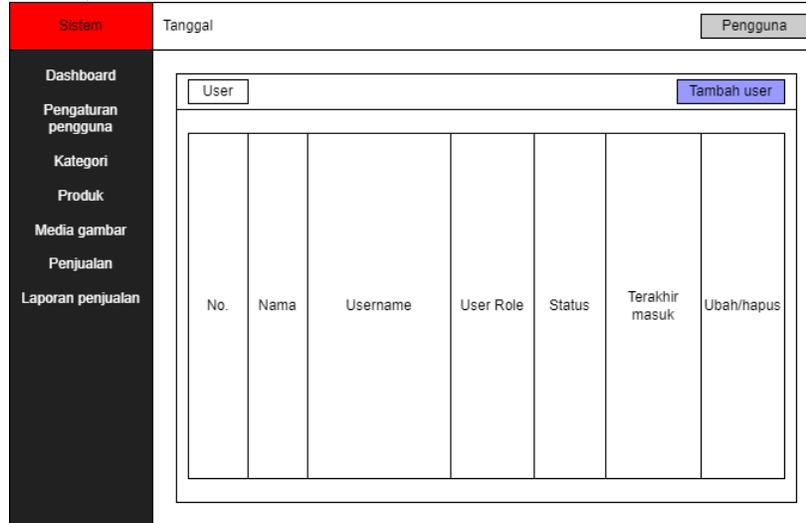
Produk terbaru

Produk

**Gambar 3.3.2 Sketsa Dashboard Admin**

Sketsa ini menunjukkan tampilan dashboard dapat melihat produk penjualan tertinggi, penjualan terbaru, produk terbaru, jumlah pengguna, jumlah kategori, jumlah produk, jumlah penjualan dalam Categories Id Kategori / Primary Key (ID) (Int,11) dan Nama Kategori / Foreign Key (Name) (Varchar,60) lalu products Id Barang / Primary Key (ID) (Int,11) dan Nama Barang / Foreign Key (Name) (Varchar,255) dan Jumlah Barang (Quantity) (Varchar,50) dan Harga Beli Barang (buy\_price) (Decimal,25,2) dan Harga jual barang (Sales\_price) (Int,11) dan Id Kategori (Categorie\_ID) (Int,11) dan Id file yang diunggah (Media\_id) (Int,11) dan Waktu saat data di input (Date,Datetime) lalu Sales Id Penjualan / Primary Key (Int,11) dan Id Barang / Foreign Key (Product\_id) (Int,11) dan Jumlah Barang (Quantity) (Int,11) dan Harga barang terjual (Price) (Decimal,25,2) dan Waktu saat Penjualan Dilakukan (date) (date) Users Pengguna / Primary Key

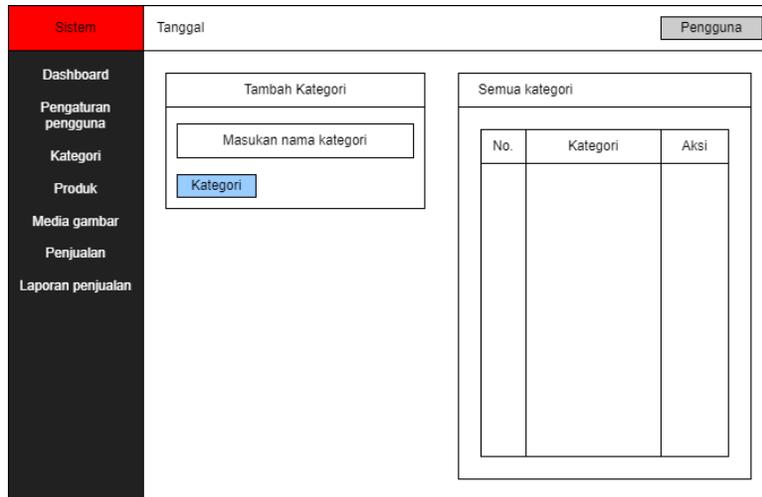
(Id) (Int,11) dan Nama pengguna (Name) (Varchar,60) dan Nama pengguna di sistem (Username) (Varchar,255) dan Kata sandi pengguna untuk login (Password) (Varchar,255) dan Tingkatan pengguna sistem/Foreign Key (User\_level) (Int,11) dan Gambar/foto pengguna (Image) (Varchar,255) dan Status pengguna (Status) (Int,1) dan Waktu terakhir pengguna login (Last\_login) (Datetime)



Gambar 3.3.3 Sketsa Atur Pengguna

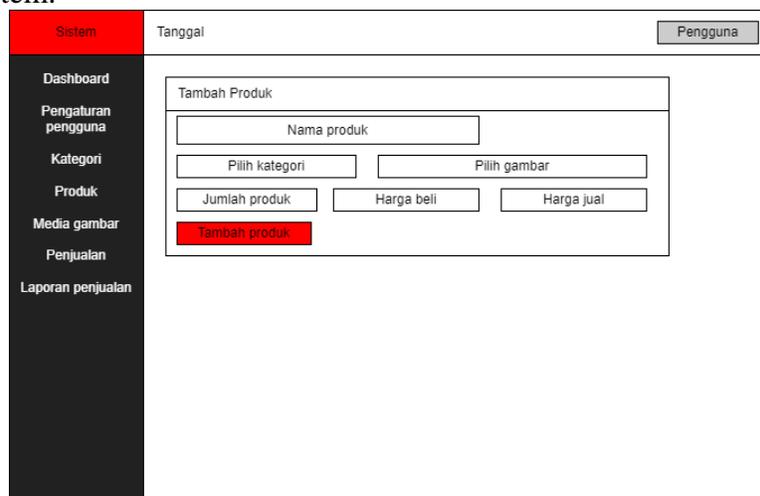
Sketsa ini menunjukkan tampilan pengaturan pengguna, berisi pengguna yang menggunakan sistem inventori bisa melihat terakhir login dan bisa hapus pengguna dan edit pengguna. Basis data pengguna yang diilustrasikan dalam gambar memiliki beberapa kolom dengan tipe data dan panjang data tertentu untuk menyimpan informasi pengguna. Kolom "id" menggunakan tipe data "Int" dengan panjang 11 sebagai Primary Key untuk mengidentifikasi setiap pengguna secara unik. Kolom "name" dengan tipe data "Varchar" dan panjang 60 menyimpan nama lengkap pengguna, sementara kolom "username" dengan tipe data "Varchar" dan panjang 50 menyimpan nama pengguna yang digunakan untuk login ke sistem. Kata sandi pengguna disimpan dalam kolom "password" dengan tipe data "Varchar" dan panjang 255, memastikan bahwa kata sandi dapat disimpan dalam bentuk terenkripsi. Kolom "user\_level," menggunakan tipe data "Int" dengan panjang 11, menyimpan tingkatan pengguna dalam sistem dan bertindak sebagai Foreign Key. Informasi gambar atau foto pengguna disimpan dalam kolom "image" dengan tipe data "Varchar" dan panjang 255. Status pengguna dicatat dalam kolom "status" dengan tipe data "Int" dan panjang 1, menunjukkan apakah pengguna aktif atau tidak. Terakhir, kolom "last\_login" dengan tipe data "Datetime" menyimpan waktu terakhir pengguna melakukan login ke sistem. Keseluruhan struktur ini dirancang untuk memastikan data pengguna tersimpan dengan lengkap dan terorganisir, mendukung fungsi manajemen pengguna dalam sistem. Basis data kelompok pengguna yang dijelaskan memiliki beberapa kolom dengan tipe data dan panjang data tertentu untuk menyimpan informasi terkait kelompok pengguna. Kolom "id" menggunakan tipe data "Int" dengan panjang 11 sebagai Primary Key, yang berfungsi untuk mengidentifikasi setiap kelompok pengguna secara unik. Kolom "group\_name" dengan tipe data "Varchar" dan panjang 150 menyimpan nama kelompok pengguna. Kolom "group\_level," dengan tipe data "Int" dan panjang 11, mencatat tingkatan atau level dari kelompok pengguna, bertindak sebagai Foreign Key untuk menunjukkan hierarki atau kategori dalam sistem. Selain itu, kolom "group\_status" juga menggunakan tipe data "Int" dengan panjang 11 untuk menyimpan status kelompok pengguna, yang mungkin

menunjukkan apakah kelompok tersebut aktif atau tidak. Struktur basis data ini dirancang untuk menyimpan informasi penting tentang berbagai kelompok pengguna secara terorganisir, yang berguna dalam pengelolaan akses dan fungsi dalam sistem.



Gambar 3.3.4 Sketsa kategori

Sketsa ini menunjukkan tampilan kategori dalam tampilan ini bisa menambahkan kategori baru, mengedit kategori yang sudah dan bisa menghapus kategori Basis data kategori memiliki dua kolom penting dengan tipe data dan panjang data tertentu untuk menyimpan informasi mengenai kategori. Kolom pertama adalah "id," yang menggunakan tipe data "Int" dengan panjang 11 dan berfungsi sebagai Primary Key, yang mengidentifikasi setiap kategori secara unik. Kolom kedua adalah "name," yang menggunakan tipe data "Varchar" dengan panjang 60 dan berfungsi untuk menyimpan nama kategori. Selain itu, kolom "name" juga bertindak sebagai Foreign Key, menghubungkan kategori dengan entitas lain dalam sistem. Struktur ini dirancang untuk menyimpan informasi kategori secara efisien dan terorganisir, mendukung fungsi pengelolaan kategori dalam sistem.



Gambar 3.3.5 Sketsa Tambah Produk

Sketsa ini menunjukkan tampilan tambah produk, Tampilan ini berisi tambah produk bisa mencantumkan nama produk yang baru dan dimasukkan dalam kategori lalu bisa menambahkan gambar sesuai produk yang kita masukkan, setelah itu bisa memasukkan jumlah produk harga modal dan harga jual Basis data barang memiliki delapan kolom penting dengan tipe data dan panjang data tertentu untuk menyimpan informasi mengenai barang. *Product* Kolom pertama adalah "id," yang menggunakan tipe data "Int" dengan panjang 11 sebagai Primary Key, berfungsi untuk mengidentifikasi setiap barang secara unik. Kolom kedua adalah "name," yang menggunakan tipe data "Varchar" dengan panjang 255 untuk menyimpan nama barang, dan juga berfungsi sebagai Foreign Key. Kolom "quantity" menggunakan tipe data "Varchar" dengan panjang 50 untuk mencatat jumlah barang yang tersedia. Harga beli barang disimpan dalam kolom "buy\_price" dengan tipe data "Decimal" dan panjang 25.2, sedangkan harga jual barang disimpan dalam kolom "sale\_price" dengan tipe data yang sama. Kolom "categorie\_id" dengan tipe data "Int" dan panjang 11 mencatat ID kategori barang, menghubungkannya dengan entitas kategori dalam sistem. Selain itu, kolom "media\_id" menggunakan tipe data "Int" dengan panjang 11 untuk menyimpan ID file yang diunggah, seperti gambar atau dokumen terkait barang tersebut. Terakhir, kolom "Date" dengan tipe data "Datetime" mencatat waktu saat data barang diinput ke dalam sistem. Struktur ini dirancang untuk menyimpan informasi barang secara efisien dan terorganisir, mendukung fungsi pengelolaan inventori dalam sistem. *Categories* Tabel "Categories" dalam basis data memiliki dua kolom penting dengan tipe data dan panjang data tertentu untuk menyimpan informasi mengenai kategori. Kolom pertama adalah "id," yang menggunakan tipe data "Int" dengan panjang 11 sebagai Primary Key. Kolom ini berfungsi untuk mengidentifikasi setiap kategori secara unik dalam sistem. Kolom kedua adalah "name," yang menggunakan tipe data "Varchar" dengan panjang 60, berfungsi untuk menyimpan nama kategori. Selain itu, kolom "name" juga dapat bertindak sebagai Foreign Key, menghubungkan kategori ini dengan entitas lain dalam sistem. Struktur tabel ini dirancang untuk menyimpan informasi kategori secara efisien dan terorganisir, mendukung pengelolaan kategori dalam sistem.



**Gambar 3.3.6 Sketsa Media Gambar**

Sketsa ini menunjukkan tampilan media gambar. Tampilan ini bisa mengunggah gambar. setelah gambar di unggah akan masuk kedalam media gambar nama gambar dan tipe gambar lalu bisa menghapus dan mengedit gambarnya. Media pertama adalah "id," yang menggunakan tipe data "Int" dengan panjang 11, berfungsi sebagai Primary Key dan juga sebagai Foreign Key,

mengidentifikasi setiap file secara unik dan menghubungkannya dengan entitas lain dalam sistem. Kolom kedua adalah "file\_name," yang menggunakan tipe data "Varchar" dengan panjang 255, berfungsi untuk menyimpan nama file. Kolom ketiga adalah "file\_type," yang menggunakan tipe data "Varchar" dengan panjang 100, berfungsi untuk mencatat tipe atau jenis file tersebut. Struktur tabel ini dirancang untuk menyimpan informasi file secara efisien dan terorganisir, mendukung pengelolaan file dalam sistem.

Sistem		Tanggal	Pengguna														
Dashboard	<input type="button" value="Pilih produk"/> Nama produk																
Pengaturan pengguna	Penjualan																
Kategori	<table border="1"> <thead> <tr> <th>Nama produk</th> <th>Harga</th> <th>Jumlah</th> <th>Total</th> <th>Tanggal</th> <th><input type="button" value="Tambah"/></th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> </tr> </tbody> </table>					Nama produk	Harga	Jumlah	Total	Tanggal	<input type="button" value="Tambah"/>						
Nama produk	Harga	Jumlah	Total	Tanggal	<input type="button" value="Tambah"/>												
Produk																	
Media gambar																	
Penjualan																	
Laporan penjualan																	

**Gambar 3.3.7 Sketsa Tambah Penjualan**

Sketsa ini menunjukkan Tampilan Tambah Penjualan, Tampilan ini berisi produk yang telah dipilih, harga, jumlah, total barang, tanggal barang yang dimasukkan setelah itu bisa menambahkan penjualan. Tabel "Sales" dalam basis data memiliki lima kolom penting dengan tipe data dan panjang data tertentu untuk menyimpan informasi mengenai penjualan. pertama adalah "id," yang menggunakan tipe data "Int" dengan panjang 11 sebagai Primary Key, berfungsi untuk mengidentifikasi setiap transaksi penjualan secara unik. Kolom kedua adalah "product\_id," yang menggunakan tipe data "Int" dengan panjang 11 dan berfungsi sebagai Foreign Key untuk menghubungkan transaksi penjualan dengan barang yang terjual. ketiga adalah "qty," yang menggunakan tipe data "Int" dengan panjang 11, berfungsi mencatat jumlah barang yang terjual dalam setiap transaksi. Kolom keempat adalah "price," yang menggunakan tipe data "Decimal" dengan panjang 25.2, berfungsi mencatat harga barang yang terjual. Terakhir, kolom "date" menggunakan tipe data "date" untuk mencatat waktu saat penjualan dilakukan. Struktur tabel ini dirancang untuk menyimpan informasi penjualan secara efisien dan terorganisir, mendukung pengelolaan data penjualan dalam sistem. Tabel "Sales" dalam basis data memiliki lima kolom penting dengan tipe data dan panjang data tertentu untuk menyimpan informasi mengenai penjualan. pertama adalah "id," yang menggunakan tipe data "Int" dengan panjang 11 sebagai Primary Key, yang berfungsi untuk mengidentifikasi setiap transaksi penjualan secara unik. kedua adalah "product\_id," yang menggunakan tipe data "Int" dengan panjang 11 dan berfungsi sebagai Foreign Key, menghubungkan transaksi penjualan dengan barang yang terjual. ketiga adalah "qty," yang menggunakan tipe data "Int" dengan panjang 11 untuk mencatat jumlah barang yang terjual dalam setiap transaksi. Kolom keempat adalah "price," yang menggunakan tipe data "Decimal" dengan panjang 25.2, mencatat harga barang yang terjual. Terakhir, kolom "date" menggunakan

tipe data "date" untuk mencatat waktu saat penjualan dilakukan. Struktur tabel ini dirancang untuk menyimpan informasi penjualan secara efisien dan terorganisir, mendukung pengelolaan data penjualan dalam sistem.

## HASIL

### Temuan atau Data Tabel Struktur Data

Tabel *categories*

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	Int	11	Id Kategori / Primary Key
2	name	varchar	60	Nama Kategori / Foreign Key

- id: Merupakan primary key yang unik untuk setiap kategori.
- name: Menyimpan nama dari kategori tersebut.

Tabel *media*

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	Int	11	Id File / Primary Key / Foreign Key
2	file_name	Varchar	255	Nama File
3	file_type	Varchar	100	Tipe File

- id: Merupakan primary key yang unik untuk setiap file, bisa juga menjadi foreign key di tabel lain.
- file\_name: Menyimpan nama file.
- file\_type: Menyimpan tipe file (misalnya: jpg, png, pdf).

Tabel *products*

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	Int	11	Id Barang / Primary Key
2	name	Varchar	255	Nama Barang / Foreign Key
3	quantity	Varchar	50	Jumlah Barang
4	buy_price	Decimal	25.2	Harga Beli Barang
5	sale_price	Decimal	25.2	Harga Jual Barang
6	categorie_id	Int	11	Id Kategori
7	media_id	Int	11	Id File yang Diunggah
8	Date	Datetime		Waktu saat data diinput

- id: Merupakan primary key yang unik untuk setiap produk.
- name: Menyimpan nama produk.
- quantity: Menyimpan jumlah produk yang ada.
- buy\_price: Menyimpan harga beli produk.
- sale\_price: Menyimpan harga jual produk.
- categorie\_id: Merupakan foreign key yang merujuk ke `id` dalam tabel `categories`.
- media\_id: Merupakan foreign key yang merujuk ke `id` dalam tabel `media`.
- Date: Menyimpan waktu kapan data produk tersebut diinput.

Tabel *sales*

No	Nama Field	Tipe Data	Panjang Data	Keterangan
----	------------	-----------	--------------	------------

1	id	Int	11	Id Penjualan / Primary Key
2	product_id	Int	11	Id Barang / Foreign Key
3	qty	Int	11	Jumlah Barang
4	price	Decimal	25.2	Harga Barang Terjual
5	date	date		Waktu saat Penjualan Dilakukan

- id: Merupakan primary key yang unik untuk setiap penjualan.
- product\_id: Merupakan foreign key yang merujuk ke `id` dalam tabel `products`.
- qty: Menyimpan jumlah barang yang terjual.
- price: Menyimpan harga barang yang terjual.
- date: Menyimpan waktu kapan penjualan dilakukan.

Tabel *users*

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	Int	11	Id Pengguna / Primary Key
2	name	Varchar	60	Nama Pengguna
3	username	Varchar	50	Nama Pengguna di Sistem
4	password	Varchar	25	Kata Sandi Pengguna untuk Login
5	user_level	Int	11	Tingkatan Pengguna Sistem / Foreign Key
6	image	Varchar	255	Gambar / Foto Pengguna
7	status	Int	1	Status Pengguna
8	last_login	Datetime		Waktu Terakhir Pengguna Login

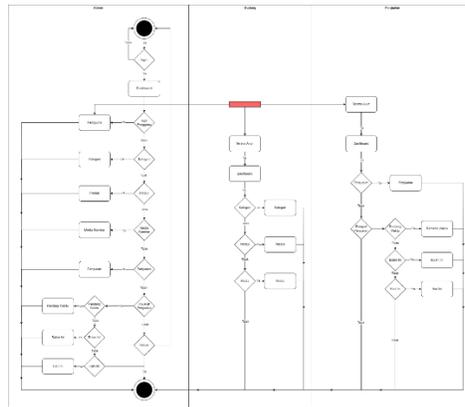
- id: Merupakan primary key yang unik untuk setiap pengguna.
- name: Menyimpan nama lengkap pengguna.
- username: Menyimpan nama pengguna yang digunakan untuk login ke sistem.
- password: Menyimpan kata sandi pengguna untuk login, dienkripsi untuk keamanan.
- user\_level: Merupakan foreign key yang merujuk ke `group_level` dalam tabel `user_groups`, menunjukkan tingkatan atau peran pengguna dalam sistem.
- image: Menyimpan URL atau path ke gambar atau foto pengguna.
- status: Menyimpan status pengguna (misalnya: aktif atau tidak aktif), dengan panjang data 1 (mungkin 0 untuk tidak aktif, 1 untuk aktif).
- last\_login: Menyimpan waktu terakhir pengguna melakukan login ke sistem.

Tabel *user\_groups*

No	Nama Field	Tipe Data	Panjang Data	Keterangan
1	id	Int	11	Id Kelompok Pengguna / Primary Key
2	group_name	Varchar	150	Nama Kelompok Pengguna
3	group_level	Int	11	Tingkatan Kelompok Pengguna / Foreign Key
4	group_status	Int	11	Status Kelompok Pengguna

- id: Merupakan primary key yang unik untuk setiap kelompok pengguna.
- group\_name: Menyimpan nama kelompok pengguna.
- group\_level: Merupakan tingkatan atau peran kelompok pengguna, bisa menjadi foreign key untuk `user_level` dalam tabel `users`.
- group\_status: Menyimpan status kelompok pengguna (misalnya: aktif atau tidak aktif).

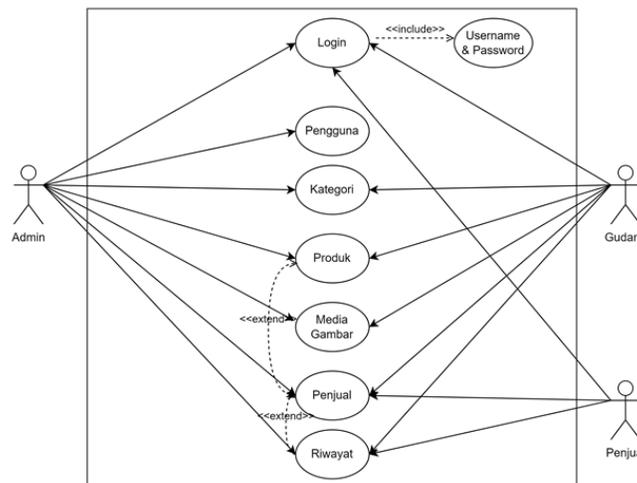
## Activity Diagram



**Gambar 4.1.1 Activity Diagram Penjualan Barang**

ADMIN Login: ADMIN melakukan login untuk mengakses sistem. Dashboard: Setelah login berhasil, ADMIN akan diarahkan ke dashboard. Pengguna: ADMIN dapat mengelola pengguna, termasuk membuat, membaca, memperbarui, atau menghapus pengguna. Kategori: ADMIN dapat mengelola kategori produk. Produk: ADMIN dapat mengelola produk. Media Gambar: ADMIN dapat mengelola media gambar untuk produk. Penjualan: ADMIN dapat melihat data penjualan. Rentang Waktu: ADMIN dapat memilih rentang waktu tertentu. Bulan Ini: ADMIN dapat melihat data penjualan untuk bulan ini. Hari Ini: ADMIN dapat melihat data penjualan untuk hari ini. Gudang Terima Akun: Gudang menerima akun untuk login ke sistem. Dashboard: Setelah login, Gudang akan diarahkan ke dashboard. Kategori: Gudang dapat mengelola kategori produk. Produk: Gudang dapat mengelola produk. Media: Gudang dapat mengelola media untuk produk. Penjualan Terima Akun: Penjualan menerima akun untuk login ke sistem. Dashboard: Setelah login, Penjualan akan diarahkan ke dashboard. Penjualan: Penjualan dapat mengelola data penjualan. Rentang Waktu: Penjualan dapat memilih rentang waktu tertentu. Bulan Ini: Penjualan dapat melihat data penjualan untuk bulan ini. Hari Ini: Penjualan dapat melihat data penjualan untuk hari ini.

## Usecase Diagram



**Gambar 4.1.2 Usecase Diagram**

1. Login
  1. Aktor harus login untuk mengakses sistem.
  2. Use case ini mencakup memasukkan Username & Password.
2. Pengguna
  1. Setelah login, aktor dapat mengelola data pengguna.
  2. Use case ini memungkinkan penambahan, penghapusan, dan pengeditan informasi pengguna.
3. Kategori
  1. Aktor dapat mengelola kategori produk.
  2. Use case ini memungkinkan penambahan, penghapusan, dan pengeditan kategori.
4. Produk
  1. Aktor dapat mengelola data produk.
  2. Use case ini memungkinkan penambahan, penghapusan, dan pengeditan informasi produk.
5. Media Gambar
  1. Use case ini terkait dengan produk, menunjukkan bahwa setiap produk dapat memiliki media gambar.
  2. Ada relasi <<extend>> dari Produk ke Media Gambar, menunjukkan bahwa menambah atau mengedit produk dapat melibatkan media gambar.
6. Penjual
  1. Aktor dapat mengelola data penjual.
  2. Use case ini memungkinkan penambahan, penghapusan, dan pengeditan informasi penjual.
7. Riwayat
  1. Aktor dapat melihat riwayat aktivitas atau transaksi.
  2. Ada relasi <<extend>> dari Produk dan Penjual ke Riwayat, menunjukkan bahwa riwayat melibatkan data produk dan penjual.

Relasi dalam diagram:

1. Include (<<include>>): Menunjukkan bahwa use case Login mencakup Username & Password.
2. Extend (<<extend>>): Menunjukkan bahwa beberapa use case (Media Gambar dan Riwayat) adalah perluasan dari use case lain (Produk dan Penjual).

Secara keseluruhan, diagram ini menggambarkan bagaimana berbagai fungsi dalam sistem pengelolaan inventori terhubung dan berinteraksi satu sama lain. Setiap use case mewakili satu fungsi utama yang dapat diakses oleh aktor setelah login ke sistem.

1. Login:
  1. Use case "Login" digunakan untuk mengautentikasi pengguna yang ingin mengakses sistem. Pengguna harus memasukkan informasi "Username & Password" untuk masuk ke dalam sistem.
  2. Terdapat hubungan <<include>> dengan "Username & Password", yang berarti use case "Login" memerlukan informasi "Username & Password" untuk dapat berfungsi.
2. Pengguna:
  1. Use case "Pengguna" menunjukkan interaksi pengguna dengan sistem. Pengguna mungkin melakukan berbagai tindakan seperti mengelola akun mereka, melihat profil, atau mengatur preferensi mereka.

## PENUTUP

### Kesimpulan

Skripsi ini berhasil merancang dan mengoptimalkan sistem berbasis *web* yang efektif untuk mengelola inventori dan penjualan. Sistem ini mampu mengoptimalkan pengelolaan stok dan transaksi penjualan melalui fitur-fitur seperti pemantauan stok real-time, pembaruan stok, dan pelaporan penjualan yang mudah digunakan. Pendekatan berbasis *web* dan prototyping yang diterapkan memungkinkan aksesibilitas yang lebih luas dan kemudahan penggunaan bagi para pengguna. Hasil pengujian menunjukkan bahwa sistem ini dapat meningkatkan efisiensi operasional dan mengurangi kesalahan dalam pengelolaan inventori. Dengan demikian, sistem yang dikembangkan dapat menjadi solusi yang signifikan bagi perusahaan atau pedagang besar maupun kecil yang ingin meningkatkan efisiensi dan akurasi dalam manajemen inventori dan penjualan mereka.

Berdasarkan hasil kuesioner yang diisi oleh responden, yang semuanya merupakan Staf Gudang, dapat disimpulkan bahwa secara umum mereka merasa puas dengan penggunaan website manajemen inventori Anda. Rata-rata kemudahan penggunaan website dinilai sebesar 4,2 dari skala 5, menunjukkan bahwa website ini cukup mudah digunakan. Tidak ada responden yang melaporkan kesulitan dalam mengakses fitur-fitur tertentu, dan kepuasan terhadap kecepatan loading halaman website mendapatkan rata-rata skor 4,4. Semua responden menyatakan bahwa fitur-fitur yang tersedia sudah sangat lengkap. Tampilan website dinilai cukup memadai dengan rata-rata skor 4,4, sedangkan UI (User Interface) dan UX (User Experience) masing-masing mendapatkan rata-rata penilaian 4,4 dan 4,6. Secara keseluruhan, website ini telah memberikan pengalaman yang positif dan efisien bagi penggunanya.

### Saran

Berdasarkan hasil yang telah dicapai dalam penelitian ini, disarankan agar pengguna terus memperbarui dan memelihara sistem informasi manajemen inventori berbasis web untuk memastikan keandalannya dan keamanannya. Selain itu, pelatihan berkelanjutan bagi pengguna sistem perlu dilakukan untuk meningkatkan kompetensi dan pemahaman mereka dalam menggunakan sistem ini secara optimal. Evaluasi berkala juga penting untuk mengidentifikasi area yang memerlukan peningkatan dan memastikan sistem tetap relevan dengan kebutuhan operasional organisasi yang terus berkembang.

### DAFTAR PUSTAKA

- , C. L., -, A. B., -, W. X., & -, W. J. (2023). The Effects of Inventory Management on Business Efficiency. *International Journal For Multidisciplinary Research*, 5(4), 1–17. <https://doi.org/10.36948/ijfmr.2023.v05i04.4877>
- Amirrudin, A. H., Kamaruddin, N. S., Salehuddin, N., & Ibrahim, S. (2023). Improving Warehouse Efficiency Through Effective Inventory Management Practices. *Social and Management Research Journal*, 20(1), 173–187. <https://ir.uitm.edu.my/id/eprint/77192/1/77192.pdf>
- Anusha, C., Aluguri, S. V., Maddiveni, S., Kandibanda, S. T. G., & Kotakonda, B. (2023). Rfid Based Inventory Management System. *International Journal of Engineering Applied Sciences and Technology*, 8(5), 133–139. <https://doi.org/10.33564/ijeast.2023.v08i05.018>
- Brady, D. (2009). Introduction to UML. *Mobile Computing Principles*, 155–228.

<https://doi.org/10.1017/cbo9780511546969.005>

- Ch. Chandra Mohan, Shaik Shoaib Ahmed, N. Vishnu Priya, M. Jahnavi, & T. Praneeth Babu. (2022). E - Health Centre Maintenance System using PHP with MySQL and XAMPP Web Server. *International Journal of Advanced Research in Science, Communication and Technology*, 2(1), 859–865. <https://doi.org/10.48175/ijarsct-7577>
- Christanto, H. J., Sutresno, S. A., & Suandi, V. S. (2023). Usability Analysis of Atma Jaya Catholic University E-Learning Based on Human Computer Interaction. *Journal of Information Systems and Informatics*, 5(3), 833–847. <https://doi.org/10.51519/journalisi.v5i3.512>
- Dey, B. K., & Seok, H. (2024). Intelligent inventory management with autonomation and service strategy. *Journal of Intelligent Manufacturing*, 35(1), 307–330. <https://doi.org/10.1007/s10845-022-02046-4>
- Dwi Agustina Cahyaningrum, Delfi Arviana Wulan Prastika, & Suseno. (2023). Rancangan Sistem Informasi Pemantauan Stok Barang Dan Penentuan Manajemen Akuntansi Penggajian Karyawan pada UMKM Bakpia. *Jurnal Teknologi Dan Manajemen Industri Terapan*, 2(3), 182–190. <https://doi.org/10.55826/tmit.v2i2.96>
- Eid Bait Bin Saleem, S. Z., & Ullah, A. (2023). The Pros and Cons of Inventory Control Strategies, and How Does it Affect the Company's Performance. *International Journal of Operations Management*, 3(1), 7–14. <https://doi.org/10.18775/ijom.2757-0509.2020.31.4001>
- Fan, Y. (2010). Development of inventory management system. *ICIME 2010 - 2010 2nd IEEE International Conference on Information Management and Engineering*, 6(April), 207–210. <https://doi.org/10.1109/ICIME.2010.5478077>
- Farhad, R. G., Dewayani, E., & Lauro, M. D. (2023). PERANCANGAN DAN PEMBUATAN APLIKASI INVENTORY MANAGEMENT BERBASIS WEB PADA PT. X MENGGUNAKAN METODE EOQ(Economic Order Quantity). *Jurnal Ilmu Komputer Dan Sistem Informasi*, 11(1). <https://doi.org/10.24912/jiksi.v11i1.24097>
- Hwang, H. C., & Kim, W. J. (2023). Design of Enhanced Document HTML and the Reliable Electronic Document Distribution Service. *Electronics (Switzerland)*, 12(10), 1–15. <https://doi.org/10.3390/electronics12102176>
- Khan Mohd, T., Thompson, J., Carmine, A., & Reuter, G. (2022). Comparative Analysis on Various CSS and JavaScript Frameworks. *Journal of Software*, 17(6), 282–291. <https://doi.org/10.17706/jsw.17.6.282-291>
- Komala Sari, R., & Isnaini, F. (2021). Perancangan Sistem Monitoring Persediaan Stok Es Krim Campina Pada Pt Yunikar Jaya Sakti. *Jurnal Informatika Dan Rekayasa Perangkat Lunak (JATIKA)*, 2(1), 151–159.
- Munyaka, J. B., & Yadavalli, V. S. S. (2022). Inventory Management Concepts and Implementations: a Systematic Review. *South African Journal of Industrial Engineering*, 33(2), 15–36. <https://doi.org/10.7166/33-2-2527>
- N., B., S.K., A., & R., R. (2023). Analyzing Grocery Items Inventory and Managing Stock. *International Academic Journal of Innovative Research*, 10(2), 10–17. <https://doi.org/10.9756/iajir/v10i2/iajir1005>
- Oywa, A. I. (2023). Inventory Management at Makerere University Business School and Its Impact on the Revenue Performance of Small and Medium Enterprises. *Idosr Journal of Current Issues in Arts and Humanities*, 9(2), 23–38. <https://doi.org/10.59298/idosrjciah/2023/2.3.3300>
- Perez, H. D., Hubbs, C. D., Li, C., & Grossmann, I. E. (2021). Algorithmic approaches to inventory management optimization. *Processes*, 9(1), 1–17. <https://doi.org/10.3390/pr9010102>

- Prof. Kiran Likhar, Anish Jha, Sudarshan Tiwari, Alice Sunar, Sanjana Shahu, & Shruti Thate. (2023). Machine Learning-Based Sales Prediction and Inventory Management for Grocery Stores. *International Journal of Advanced Research in Science, Communication and Technology*, 24–32. <https://doi.org/10.48175/ijarsct-13605>
- Rachana Beniwal, Arpita Sharma, Saloni Jain, Shashank Vyas. (2023). A Review on Full Stack Web Development. *Tuijin Jishu/Journal of Propulsion Technology*, 44(1), 191–196. <https://doi.org/10.52783/tjjpt.v44.i1.2240>
- San-José, L. A., Sicilia, J., Pando, V., & Alcaide-López-de-Pablo, D. (2022). Optimization of an inventory system with partial backlogging from a financial investment perspective. *International Transactions in Operational Research*, 29(2), 706–728. <https://doi.org/10.1111/itor.13038>
- Shanmugapriya, M. K. (2023). Warehouse Management and Inventory Control. *International Scientific Journal of Engineering and Management*, 02(03), 1–6. <https://doi.org/10.55041/isjem00217>
- Shazna, S. ., & Shazna, S. . (2021). an Effective Procurement and Inventory Management System for Abc Textile Shop. *International Journal of Engineering Applied Sciences and Technology*, 5(9), 76–82. <https://doi.org/10.33564/ijeast.2021.v05i09.011>
- Sridhar, P., Vishnu, C. R., & Sridharan, R. (2021). Simulation of inventory management systems in retail stores: A case study. *Materials Today: Proceedings*, 47(xxxx), 5130–5134. <https://doi.org/10.1016/j.matpr.2021.05.314>
- Šušter, I., & Ranisavljević, T. (2023). Optimization of MySQL database. *Journal of Process Management and New Technologies*, 11(1–2), 141–151. <https://doi.org/10.5937/jouproman2301141q>
- Wibowo, D. S., Ilmadina, H. Z., Ardi, A. S., & Insani, F. F. G. (2023). Apache web server security with security hardening. *Journal of Soft Computing Exploration*, 4(4), 213–221. <https://doi.org/10.52465/josce.v4i4.230>
- Yang, S., Wei, Z., Herman, G. L., & Alawini, A. (2021). Analyzing Patterns in Student SQL Solutions via Levenshtein Edit Distance. *L@S 2021 - Proceedings of the 8th ACM Conference on Learning @ Scale*, 323–326. <https://doi.org/10.1145/3430895.3460979>