

Optimalisasi Manajemen Stok UMKM Kosmetik Melalui Aplikasi Inventaris dengan Notifikasi Kadaluarsa Otomatis

Nabila Farah Hanani¹, Clarisah Razendri Ingnasia², Lovina Putri Pratama³, Salman Alfarizi⁴, Achmad Zaki Al Akbar⁵, Fawwaz Ali Akbar^{6*}

^{1,2,3,4,5}Informatika, Universitas Pembangunan Nasional Jawa Timur

123081010003@student.upnjatim.ac.id

223081010016@student.upnjatim.ac.id

323081010018@student.upnjatim.ac.id

423081010020@student.upnjatim.ac.id

523081010022@student.upnjatim.ac.id

⁶Informatika, Universitas Pembangunan Nasional Jawa Timur

*Corresponding author email: fawwaz.ali.fik@upnjatim.ac.id

Abstrak— transformasi digital pada Usaha Mikro, Kecil, dan Menengah (UMKM) menjadi faktor krusial untuk meningkatkan daya saing di era modern. Namun, banyak UMKM, khususnya yang bergerak di bidang produk dengan masa simpan terbatas seperti *skincare*, masih mengandalkan manajemen inventaris manual. Praktik ini berisiko tinggi terhadap kesalahan data, ketidakefisienan operasional, dan kerugian finansial akibat produk kadaluarsa. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sebuah sistem informasi manajemen produk berbasis desktop sebagai solusi digital bagi UMKM “*Qeemla Skin & Body Care*”. Pengembangan sistem ini menggunakan metodologi *waterfall* dan dibangun dengan bahasa pemrograman *Java* yang berorientasi objek (*OOP*) serta *library Java Swing* untuk antarmuka pengguna grafis (*GUI*). Fitur-fitur utama yang dikembangkan mencakup autentikasi pengguna yang aman, manajemen data produk (*CRUD*), dan sistem peringatan dini otomatis untuk produk yang mendekati dan telah melewati tanggal kadaluarsa. Hasil akhir dari penelitian ini adalah sebuah aplikasi yang terstruktur, responsif, dan ramah pengguna, yang secara efektif dapat menggantikan proses manual. Implementasi sistem ini diharapkan mampu mengoptimalkan kontrol stok, meminimalkan kerugian, serta menyediakan data yang akurat untuk mendukung pengambilan keputusan bisnis yang lebih baik bagi UMKM.

Kata Kunci— Digitalisasi UMKM, *Java Swing*, Manajemen Inventaris, Pemrograman Berorientasi Objek, Sistem Informasi, Hasil Pengujian.

I. PENDAHULUAN

Di tengah persaingan bisnis yang semakin ketat, adopsi teknologi digital bukan lagi pilihan melainkan keharusan bagi Usaha Mikro, Kecil, dan Menengah (UMKM) untuk dapat bertahan dan berkembang. Proses digitalisasi terbukti mampu meningkatkan efisiensi operasional dan memperluas jangkauan pasar secara signifikan, serta meningkatkan kinerja pemasaran UMKM [1][2]. Meskipun demikian, banyak UMKM di Indonesia masih menghadapi berbagai tantangan dalam

implementasi teknologi, terutama dalam pengelolaan proses bisnis internal seperti manajemen inventaris[3].

Manajemen inventaris yang dilakukan secara manual menggunakan pencatatan buku atau spreadsheet sederhana sangat rentan terhadap human error, seperti kesalahan input data dan kelalaian dalam pemantauan stok [4]. Bagi bisnis yang menjual produk dengan masa simpan terbatas, seperti industri makanan, farmasi, dan kosmetik, tantangan ini menjadi lebih kritis. Kegagalan dalam memantau tanggal kadaluarsa produk secara akurat dapat menyebabkan kerugian finansial yang signifikan dan menurunkan kepercayaan konsumen terhadap kualitas produk [5]. Studi kasus dalam penelitian ini adalah “*Qeemla Skin & Body Care*”, sebuah UMKM di bidang *skincare* yang menghadapi tantangan operasional tersebut.

Untuk mengatasi permasalahan ini, diperlukan sebuah sistem informasi terkomputerisasi yang mampu mengelola data produk secara terstruktur dan otomatis [6]. Penelitian ini mengusulkan pengembangan sebuah aplikasi manajemen produk berbasis desktop yang dirancang khusus untuk kebutuhan UMKM. Sistem ini dibangun dengan bahasa pemrograman *Java* dan antarmuka grafis (*GUI*) menggunakan *library Java Swing*. Pemilihan *Java* didasarkan pada keandalannya dan dukungan kuat terhadap paradigma Pemrograman Berorientasi Objek (*OOP*) [7]. Pendekatan *OOP* memungkinkan pengembangan perangkat lunak yang terstruktur, modular, dan mudah untuk dikembangkan di masa depan [11]. Fokus utama sistem adalah menyediakan fungsionalitas manajemen produk yang komprehensif, dengan fitur unggulan berupa sistem peringatan dini masa kadaluarsa produk secara otomatis [8].

II. TINJAUAN PUSTAKA

Dalam penelitian ini, terdapat sejumlah istilah teknis dan konsep kunci yang menjadi fondasi dalam perancangan, pengembangan, serta implementasi aplikasi manajemen inventaris pada UMKM *Qeemla Skin & Body Care*.

Pemahaman yang komprehensif terhadap istilah-istilah tersebut sangat penting untuk memperjelas ruang lingkup penelitian, memastikan konsistensi terminologi, serta menghubungkan penelitian ini dengan referensi ilmiah yang relevan. Berikut adalah penjelasan mengenai istilah-istilah utama yang digunakan dalam penelitian ini:

- **Digitalisasi UMKM:** merujuk pada proses transformasi aktivitas bisnis Usaha Mikro, Kecil, dan Menengah (UMKM) dari sistem manual ke sistem berbasis teknologi digital. Digitalisasi bertujuan meningkatkan efisiensi operasional, memperluas akses pasar, serta meminimalisir risiko kesalahan data dan kerugian finansial. Dalam konteks penelitian ini, digitalisasi diwujudkan melalui penerapan aplikasi inventaris berbasis *desktop* yang menggantikan pencatatan manual.
- **Manajemen Inventaris:** proses pengelolaan data stok barang, mulai dari pencatatan, pemantauan, hingga pengendalian persediaan produk. Tujuan utama manajemen inventaris adalah memastikan ketersediaan barang yang cukup, menghindari kelebihan atau kekurangan stok, serta meminimalisir kerugian akibat produk kadaluarsa atau rusak.
- **Sistem Informasi:** kombinasi antara teknologi, prosedur, dan sumber daya manusia yang digunakan untuk mengumpulkan, mengelola, menyimpan, dan mendistribusikan informasi guna mendukung pengambilan keputusan dalam organisasi. Pada penelitian ini, sistem informasi diwujudkan dalam bentuk aplikasi *desktop* yang memfasilitasi pengelolaan inventaris secara otomatis dan terstruktur.
- **Pemrograman Berorientasi Objek (Object-Oriented Programming/OOP):** paradigma pemrograman yang menggunakan konsep "objek" untuk merepresentasikan data dan fungsi dalam perangkat lunak. OOP memudahkan pengembangan sistem yang modular, terstruktur, dan mudah dikembangkan di masa depan. Konsep inti OOP meliputi enkapsulasi, pewarisan (*inheritance*), dan *polimorfisme*.
- **Java:** bahasa pemrograman tingkat tinggi yang bersifat multiplatform dan banyak digunakan untuk pengembangan aplikasi *desktop*, *web*, dan *mobile*.
- **Java Swing:** merupakan *library* di Java yang digunakan untuk membangun antarmuka pengguna grafis (GUI) pada aplikasi *desktop*. Swing menyediakan berbagai komponen visual seperti tombol, tabel, form, dan notifikasi *pop-up*.
- **CRUD:** adalah singkatan dari *Create*, *Read*, *Update*, dan *Delete*, yaitu empat operasi dasar yang digunakan untuk mengelola data dalam aplikasi. Pada sistem inventaris, CRUD digunakan untuk menambah, menampilkan, memperbarui, dan menghapus data produk.
- **Notifikasi kadaluarsa otomatis:** fitur yang memberikan peringatan secara *real-time* kepada admin ketika ada produk yang mendekati atau telah melewati tanggal kadaluarsa. Fitur ini membantu pemilik usaha untuk segera mengambil tindakan, seperti melakukan promosi atau menarik produk dari stok, sehingga dapat mengurangi potensi kerugian.
- **Autentikasi pengguna:** proses verifikasi identitas pengguna sebelum diberikan akses ke sistem. Dalam aplikasi ini, autentikasi dilakukan melalui halaman *login* yang memvalidasi email dan kata sandi admin, sehingga hanya pengguna yang berwenang yang dapat mengakses data inventaris.
- **Log aktivitas:** catatan sistem yang merekam setiap aksi atau perubahan yang dilakukan oleh pengguna, termasuk waktu dan identitas pengguna. Log ini digunakan untuk meningkatkan keamanan, memudahkan *audit*, dan memastikan akuntabilitas dalam pengelolaan data.
- **Metodologi *waterfall*:** model pengembangan perangkat lunak yang bersifat berurutan dan sistematis. Setiap tahapan, mulai dari analisis kebutuhan, perancangan, implementasi, pengujian, hingga pemeliharaan, dilakukan secara berurutan dan tidak dapat kembali ke tahap sebelumnya tanpa menyelesaikan tahap yang sedang berjalan.
- ***Black-box testing*:** metode pengujian perangkat lunak yang berfokus pada pengujian fungsionalitas sistem dari perspektif pengguna, tanpa memperhatikan struktur internal kode program. Tujuannya adalah untuk memastikan bahwa aplikasi berjalan sesuai dengan spesifikasi dan kebutuhan pengguna.
- ***User Interface (UI)*:** adalah tampilan visual yang digunakan pengguna untuk berinteraksi dengan aplikasi. Desain UI yang baik berperan penting dalam kemudahan penggunaan, kenyamanan, dan penerimaan aplikasi oleh pengguna akhir.
- ***User-Friendly*:** mengacu pada kemudahan penggunaan sebuah aplikasi, di mana sistem dirancang agar intuitif dan mudah dipahami oleh pengguna, bahkan bagi mereka yang tidak memiliki latar belakang teknis.
- ***Barcode Scanner*:** fitur ini memungkinkan aplikasi untuk membaca dan mengenali kode batang (*barcode*) pada produk menggunakan perangkat keras seperti *webcam*. Dengan *barcode scanner*, input data produk menjadi lebih cepat dan akurat.
- ***ProductManager*:** sebuah kelas atau modul dalam aplikasi yang bertugas mengelola semua operasi terkait data produk, seperti penambahan, pembaruan, penghapusan, dan penampilan data produk.
- ***Event Handling*:** proses pemrograman yang memungkinkan aplikasi merespon aksi pengguna, seperti klik tombol, input data, atau interaksi lain pada antarmuka aplikasi.
- ***JOptionPane*:** merupakan komponen dari *Java Swing* yang digunakan untuk menampilkan dialog, pesan, atau notifikasi *pop-up* kepada pengguna, seperti konfirmasi penghapusan data atau peringatan produk kadaluarsa.
- ***Flash Sale*:** strategi pemasaran di mana produk dijual dengan harga diskon dalam waktu yang sangat terbatas, biasanya untuk menghabiskan stok produk yang mendekati kadaluarsa.
- ***Human Error*:** kesalahan yang disebabkan oleh kelalaian atau kekeliruan manusia dalam proses pencatatan atau pengelolaan data, yang sering terjadi pada sistem manual.

III. METODOLOGI PENELITIAN

Proses pengembangan perangkat lunak ini mengadopsi model *waterfall* [9]. Model ini dipilih karena pendekatannya yang sistematis dan berurutan, sangat cocok untuk proyek dengan lingkup dan kebutuhan yang telah terdefinisi dengan jelas di awal. Tahapan dalam metodologi ini meliputi:

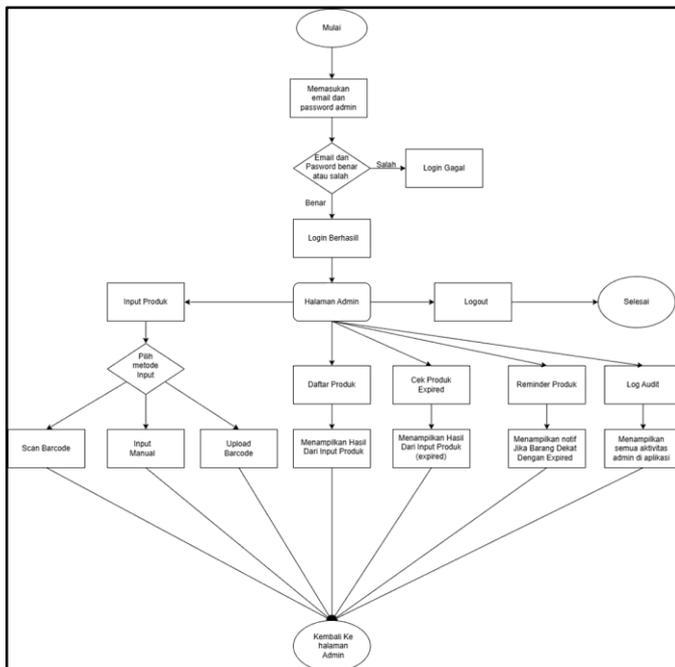
A. Analisis Kebutuhan Sistem

Tahap pertama adalah melakukan analisis mendalam terhadap proses bisnis manual di *Qeemla Skin & Body Care* untuk mengidentifikasi masalah dan kebutuhan fungsional sistem. Kebutuhan utama yang teridentifikasi adalah:

- Sistem autentikasi untuk membatasi akses hanya bagi admin.
- Fungsionalitas *Create, Read, Update, Delete* (CRUD) untuk mengelola data produk (nama, stok, harga, tanggal kadaluarsa).
- Sistem untuk menampilkan seluruh daftar produk yang tersimpan secara akurat.
- Mekanisme otomatis untuk mendeteksi produk yang telah atau akan segera kadaluarsa.
- Antarmuka yang sederhana dan mudah digunakan (*user-friendly*).

B. Perancangan Sistem

Pada tahap ini, dilakukan perancangan arsitektur sistem menggunakan pendekatan Pemrograman Berorientasi Objek (OOB) [7]. OOB memungkinkan pemisahan logis antara data (atribut) dan perilaku (metode) ke dalam objek-objek terpisah, seperti objek Produk dan Pengguna, sehingga menghasilkan kode yang lebih terorganisir dan dapat digunakan kembali (*reusable*). Pada gbr. 1 tertera *flow* fitur yang tersedia dan akan diintegrasikan dengan antarmuka (UI) yang difokuskan pada kemudahan penggunaan dan kejelasan informasi, karena desain antarmuka yang baik merupakan faktor kunci dalam penerimaan sistem oleh pengguna akhir [10][12]. *Library Java Swing* dipilih untuk membangun komponen-komponen visual seperti form, tabel, tombol, dan notifikasi *pop-up* (*OptionPane*).



Gambar 1. Flow aplikasi

Setelah pemaparan mengenai *flow* aplikasi, aspek fundamental lain yang menjadi penopang utama sistem adalah pengelolaan database. Database berperan sebagai pusat penyimpanan seluruh data produk, stok, transaksi, serta informasi penting lainnya yang diperlukan dalam proses manajemen inventaris pada UMKM *Qeemla Skin & Body Care*. Membuat database dapat dilakukan dengan menggunakan *syntax* di aplikasi *SQL* seperti *SQLYog*. Selain database, juga membuat tabel untuk log audit dan products yang hasilnya akan muncul tabel seperti gbr. 3 dan gbr.4. Dengan adanya database yang terstruktur, aplikasi mampu melakukan pencatatan, pemantauan, dan pengelolaan data secara otomatis dan akurat, sehingga meminimalisir risiko kesalahan akibat proses manual serta mendukung pengambilan keputusan yang lebih tepat.

```
CREATE DATABASE qeemla_db;

CREATE TABLE log_audit (
  id INT AUTO_INCREMENT PRIMARY KEY,
  waktu_aksi TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  aksi VARCHAR(100),
  detail TEXT,
  email_pengguna VARCHAR(100)
);

CREATE TABLE products (
  kode VARCHAR(20) PRIMARY KEY,
  nama VARCHAR(100),
  kategori VARCHAR(50),
  harga DECIMAL(10, 2),
  stok INT,
  production_date DATE,
  expiry_date DATE,
  photo_path VARCHAR(255)
);
```

Gambar 2. Syntax pembuatan database, tabel log audit, dan tabel produk

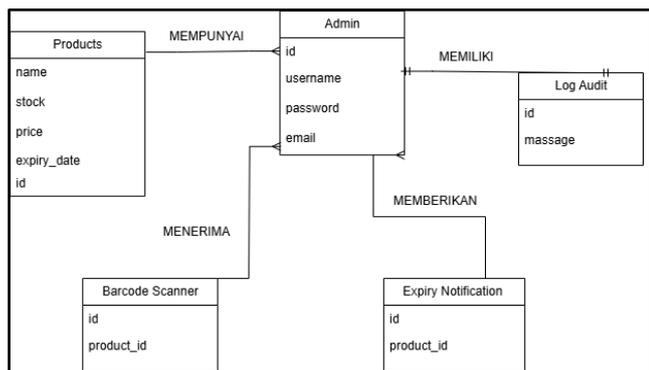
id	waktu_aksi	aksi	detail	email_pengguna
(Auto)	current_times	(NULL)	(NULL)	OK (NULL)

Gambar 3. Tabel log audit

kode	nama	kategori	harga	stok	production_date	expiry_date	photo_path
(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)	(NULL)

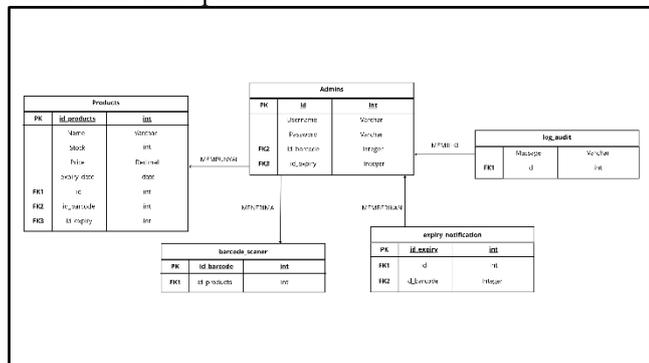
Gambar 4. Tabel produk

Untuk memastikan desain database yang optimal dan sesuai kebutuhan bisnis, dilakukan perancangan model data secara bertahap. Tahap awal dimulai dengan penyusunan *Conceptual Data Model* (CDM) seperti pada gbr.5, yang memetakan entitas utama beserta relasi antar entitas secara konseptual. CDM ini menjadi fondasi dalam memahami kebutuhan data, sebelum dilanjutkan ke tahap desain fisik database melalui *Physical Data Model* (PDM). Dengan pendekatan ini, struktur data yang dihasilkan dapat mendukung seluruh fitur aplikasi secara efektif dan efisien.



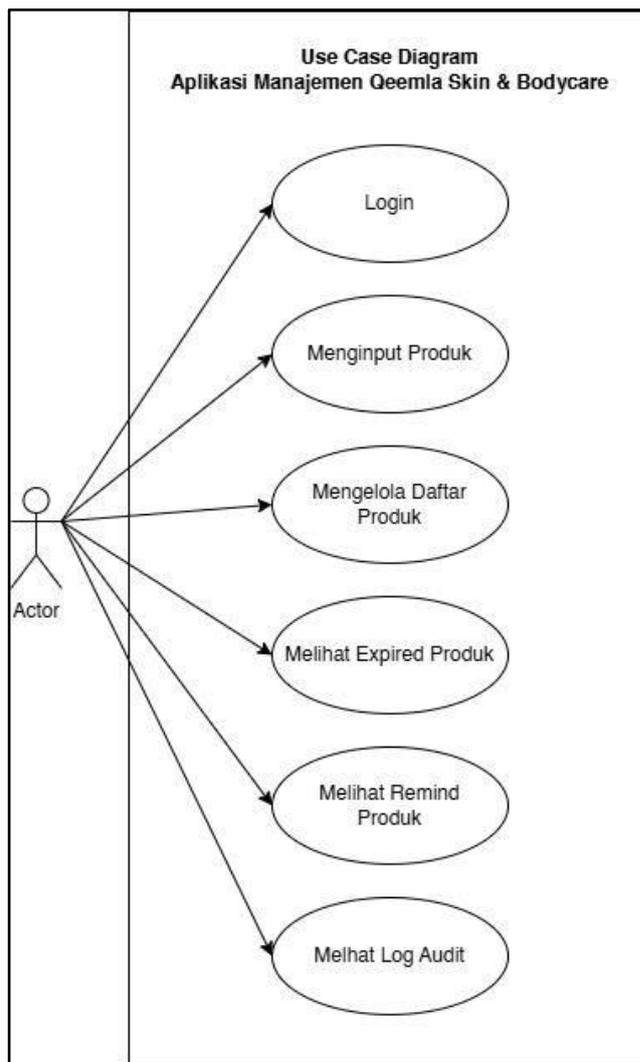
Gambar 5. Conceptual Data Model (CDM) aplikasi

Selanjutnya, pada gbr. 6 yaitu *PDM (Physical Data Model)* dari aplikasi yang disajikan untuk menggambarkan implementasi nyata dari struktur data yang telah dirancang pada tingkat konseptual. *PDM* memuat detail teknis seperti tipe data, struktur tabel, serta relasi antar tabel yang akan digunakan dalam basis data aplikasi.



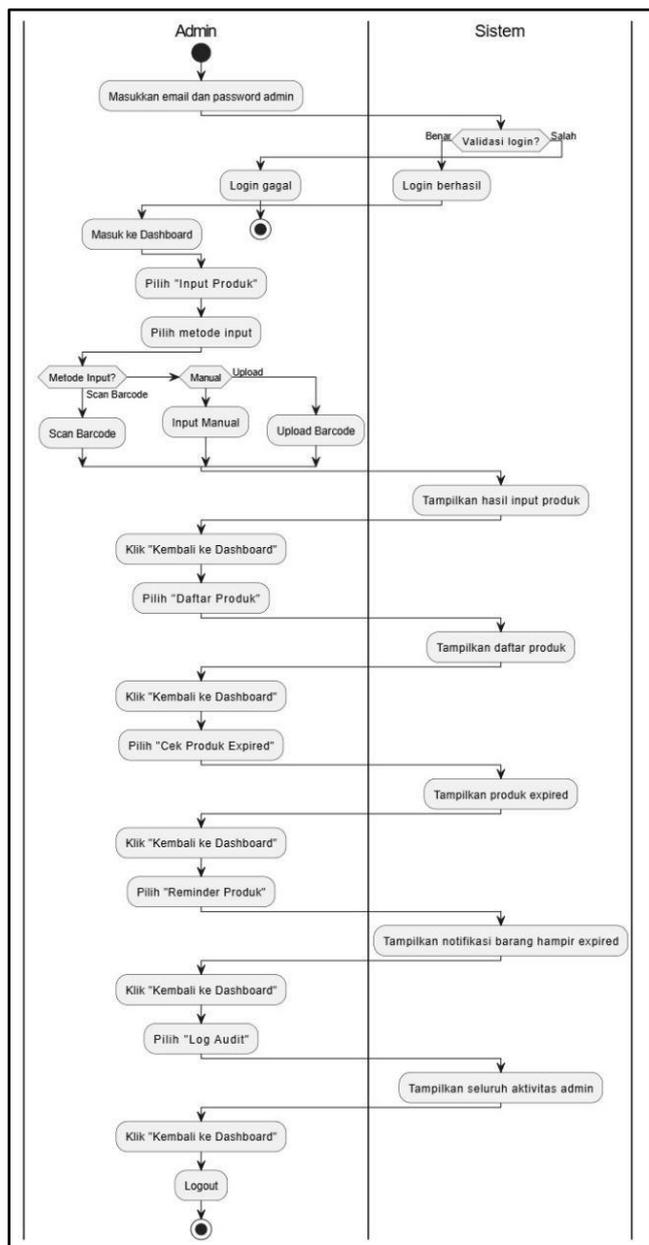
Gambar 6. Physical Data Model (PDM) aplikasi

Untuk memvisualisasikan interaksi antara pengguna dan sistem, digunakan *use case diagram* seperti pada gbr. 7. Diagram ini menggambarkan skenario-skenario utama yang dapat dilakukan oleh aktor (admin) dalam aplikasi, mulai dari proses *login*, pengelolaan data produk, hingga pemantauan notifikasi kadaluarsa. Dengan adanya *use case diagram*, kebutuhan fungsional sistem dapat divisualisasikan secara ringkas dan mudah dipahami.



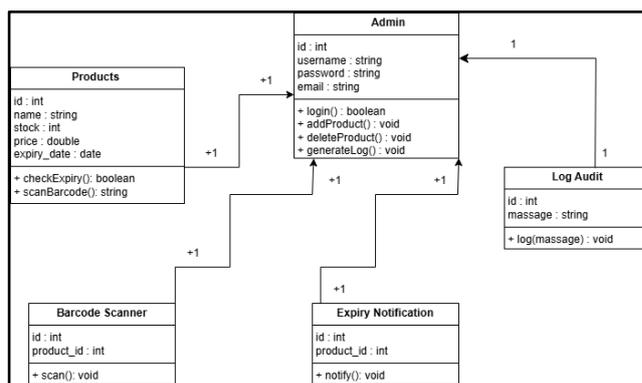
Gambar 7. Use case diagram aplikasi

Selain itu, ada *activity diagram* seperti pada gbr.8 disajikan untuk memperjelas alur aktivitas atau proses bisnis yang terjadi dalam aplikasi, seperti proses *input* data produk, pengecekan otomatis masa kadaluarsa, hingga pengelolaan stok. Diagram ini membantu dalam mengidentifikasi langkah-langkah operasional yang dilakukan oleh sistem maupun pengguna secara berurutan.



Gambar 8. Activity diagram aplikasi

Terakhir, terdapat pula gbr.9 yaitu *class diagram* yang digunakan untuk menggambarkan struktur kelas dan hubungan antar kelas dalam aplikasi berbasis *Java* yang dikembangkan. *Class diagram* ini merepresentasikan desain perangkat lunak secara berorientasi objek (*OOP*), yang memudahkan proses pengembangan, pemeliharaan, serta pengembangan fitur di masa mendatang.



Gambar 9. Class diagram aplikasi

Dengan penyajian model-model perancangan ini, diharapkan pembaca dapat memperoleh pemahaman yang komprehensif mengenai struktur data, alur proses, serta interaksi yang terjadi dalam aplikasi, sehingga dapat menilai kesesuaian solusi yang diusulkan terhadap permasalahan manajemen inventaris pada UMKM *Qeemla Skin & Body Care*.

Manajemen inventaris pada aplikasi ini sekaligus diotomatisasi dengan pencatatan dan pemantauan data secara langsung mengurangi potensi *human error* yang sering terjadi pada sistem manual. Hal ini berdampak pada peningkatan akurasi data inventaris, yang merupakan fondasi untuk pengambilan keputusan yang lebih baik dan terinformasi [14].

Dari segi teknis, penggunaan paradigma *OOP* terbukti sangat efektif dalam membangun aplikasi ini. Struktur modular yang dihasilkan membuat kode lebih mudah dipahami, dimodifikasi, dan dikembangkan. Sebagai contoh, jika di masa depan diperlukan penambahan fitur baru seperti manajemen transaksi, fitur tersebut dapat diintegrasikan sebagai modul baru tanpa mengganggu fungsi yang sudah ada.

Fitur peringatan dini kadaluarsa memberikan nilai tambah yang signifikan bagi bisnis. Dengan mengetahui produk mana yang akan segera habis masa edarnya, pemilik usaha dapat merancang strategi pemasaran seperti flash sale atau diskon untuk menghabiskan stok tersebut. Ini secara langsung mengubah potensi kerugian (biaya produk terbuang) menjadi pendapatan, sehingga meningkatkan efisiensi dan mengurangi pemborosan [15]. Aplikasi ini, dengan demikian, tidak hanya berfungsi sebagai alat pencatatan, tetapi juga sebagai alat bantu pengambilan keputusan strategis.

C. Implementasi dan Pengkodean

Tahap implementasi adalah proses penerjemahan hasil desain ke dalam kode program menggunakan bahasa *Java*.

- Logika Bisnis: Kelas-kelas *Java* dibuat untuk menangani setiap fungsi, seperti *ProductManager* untuk operasi *CRUD* dan *ExpiryNotifier* untuk logika pengecekan tanggal kadaluarsa.
- Manajemen kadaluarsa: Logika inti dari fitur ini adalah membandingkan tanggal kadaluarsa setiap produk dengan tanggal sistem saat ini. Perbandingan ini menggunakan *library* manajemen tanggal di *Java* untuk memastikan akurasi.
- Antarmuka Pengguna (*GUI*): Kelas-kelas dari *Java Swing* digunakan untuk membangun setiap jendela aplikasi, mulai dari layar login, dasbor utama, formulir input produk,

hingga tabel daftar produk. Event handling diimplementasikan untuk membuat aplikasi merespons aksi pengguna, seperti klik tombol.

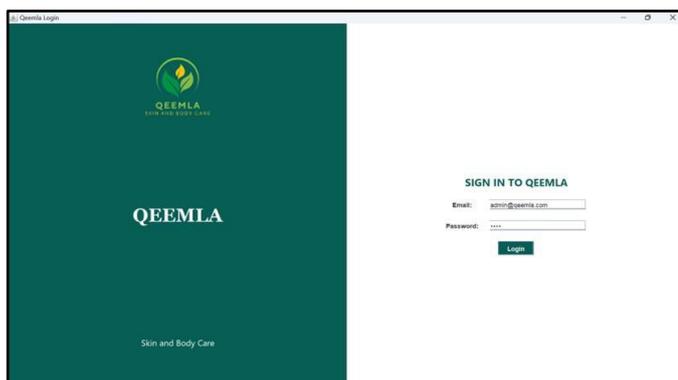
D. Pengujian Sistem

Pengujian dilakukan untuk memastikan semua fitur berjalan sesuai dengan kebutuhan yang telah ditentukan. Pengujian *black-box* diterapkan untuk memverifikasi fungsionalitas dari perspektif pengguna tanpa melihat kode internal. Tujuan pengujian adalah untuk menemukan dan memperbaiki bug sebelum sistem digunakan, karena pengujian merupakan tahap vital dalam menjamin kualitas perangkat lunak [13].

IV. HASIL DAN PEMBAHASAN

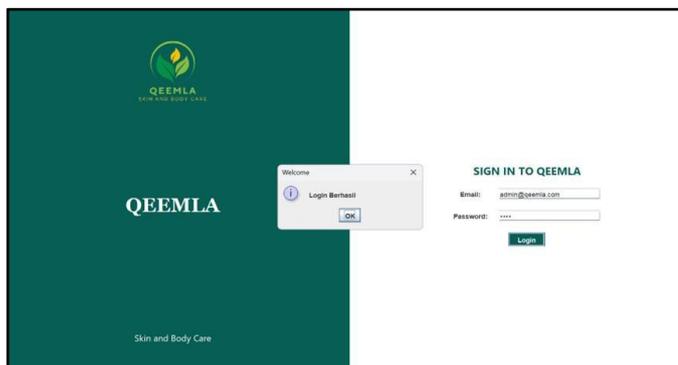
Hasil penelitian ini adalah sebuah aplikasi manajemen produk berbasis desktop yang fungsional dan siap pakai. Aplikasi ini berhasil mengintegrasikan beberapa fitur utama yang dirancang untuk menjawab permasalahan yang dihadapi oleh UMKM. Berikut adalah penjelasan dari berbagai fitur yang ada:

- **Autentikasi Pengguna:** Aplikasi memiliki halaman login yang aman untuk memastikan hanya admin yang berwenang yang dapat mengakses data inventaris.



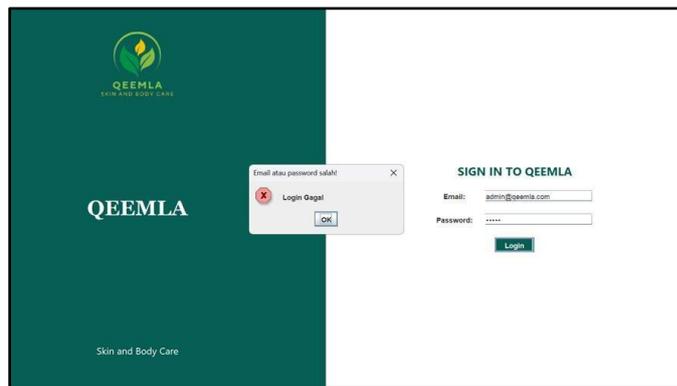
Gambar 10. Tampilan halaman login admin

Pada gbr. 10 terdapat proses login sistem memverifikasi data admin, termasuk email dan kata sandi.



Gambar 11. Tampilan halaman login admin jika berhasil

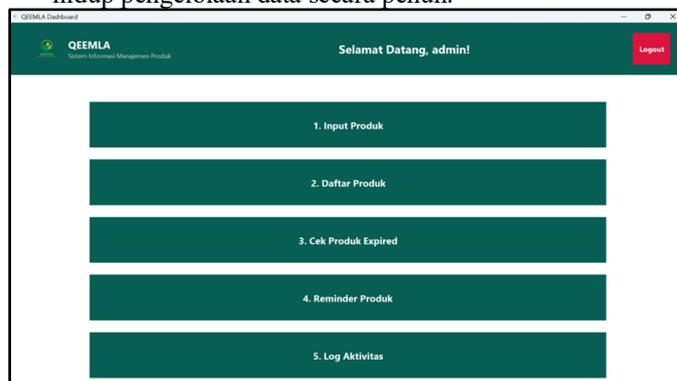
Pada gbr. 11 terdapat proses login berhasil yang akan menampilkan pesan login berhasil akan ditampilkan oleh sistem jika data yang dimasukkan benar, dan admin akan diarahkan ke halaman utama aplikasi.



Gambar 12. Tampilan halaman login admin jika gagal

Pada gbr. 12 menampilkan pesan login gagal, yakni sistem akan memberi tahu admin jika email atau kata sandi salah.

- **Manajemen Produk Komprehensif:** Pengguna (dalam hal ini admin) dapat dengan mudah menambah, melihat, memperbarui, dan menghapus data produk melalui antarmuka formulir dan tabel yang intuitif. Hal ini mendukung siklus hidup pengelolaan data secara penuh.



Gambar 13. Tampilan halaman dashboard aplikasi

Pada gbr. 13 terdapat bagian input produk yang pada halaman ini memungkinkan admin memiliki beberapa opsi untuk memasukkan data produk ke dalam sistem pada fitur ini. Opsi-opsi ini termasuk:

- Scan barcode digunakan untuk admin melakukan scan pada barcode produk dengan menggunakan *webcam* yang ada di aplikasi *Qeemla*.
- Input manual digunakan untuk mengisi data produk secara manual tanpa *barcode*.
- Upload *barcode* memiliki kemampuan untuk mengunggah file gambar *barcode* produk. Sistem akan menampilkan informasi produk yang baru dimasukkan setelah produk dimasukkan.

Dimana ketiga fitur itu masuk ke dalam bagian input produk sesuai dengan tampilan pada gbr. 14.

Gambar 14. Tampilan halaman input produk

Apabila pengisian produk pada form tambah produk sudah terisi dengan benar, maka akan muncul pesan berhasil seperti pada gbr. 15.

Gambar 15. Tampilan halaman input produk telah berhasil

Pada gbr. 16 terdapat fitur daftar produk yang memungkinkan melihat manajemen semua produk yang terdaftar dalam sistem, baik yang telah dimasukkan sebelumnya maupun yang baru saja dimasukkan. Admin dapat memeriksa daftar produk dan memilih untuk kembali ke halaman utama.

Foto	Kode	Nama	Kategori	Harga	Stok	Tgl Produksi	Tgl Expired
	8998800100018	Facial Wash Brightening	Pembersih Wajah	35000.0	100	2025-06-13	2026-06-13
	8998800100025	Serum Qeemia Skin	Pelembab	50000.0	50	2025-06-13	2026-06-13
	8998800100032	Sunscreen SPF 50 PA+++	Day Cream	45000.0	50	2025-06-13	2025-07-07
	8998800100049	Toner Qeemia Skin	Toner Wajah	40000.0	100	2025-06-13	2025-06-25

Gambar 16. Tampilan halaman daftar produk

Admin juga dapat mengedit dan menghapus daftar produk yang ada apabila terdapat kesalahan input seperti pada gbr. 17. Hal ini juga termasuk implementasi dari konsep *Update* dan *Delete* pada *CRUD*. Terdapat pesan konfirmasi sebelum menghapus produk untuk memastikan pilihan admin seperti pada gbr. 18.

Jika memilih “Yes”, maka akan ada notifikasi “Produk berhasil dihapus” seperti pada gbr. 19.

Gambar 17. Tampilan halaman edit produk

Foto	Kode	Nama	Kategori	Harga	Stok	Tgl Produksi	Tgl Expired
	8998800100018	Facial Wash Brightening	Pembersih Wajah	35000.0	100	2025-06-13	2026-06-13
	8998800100025	Serum Qeemia Skin	Pelembab	50000.0	50	2025-06-13	2026-06-13
	8998800100032	Sunscreen SPF 50 PA+++	Day Cream	45000.0	50	2025-06-13	2025-07-07
	8998800100049	Toner Qeemia Skin	Toner Wajah	40000.0	100	2025-06-13	2025-06-25

Gambar 18. Tampilan pop up jika menghapus produk

Foto	Kode	Nama	Kategori	Harga	Stok	Tgl Produksi	Tgl Expired
	8998800100018	Facial Wash Brightening	Pembersih Wajah	35000.0	100	2025-06-13	2026-06-13
	8998800100032	Sunscreen SPF 50 PA+++	Day Cream	45000.0	50	2025-06-13	2025-07-07
	8998800100049	Toner Qeemia Skin	Toner Wajah	40000.0	100	2025-06-13	2025-06-25

Gambar 19. Tampilan pop up jika hapus produk berhasil

- Sistem Notifikasi kadaluarsa Cerdas: Ini adalah fitur paling krusial. Sistem secara proaktif memberikan peringatan visual, misalnya melalui notifikasi *pop-up* atau penandaan warna pada tabel, untuk produk yang mendekati tanggal kadaluarsa. Sistem ini juga dapat memfilter dan menampilkan daftar produk yang sudah kadaluarsa untuk segera ditarik dari stok. Seperti pada gbr. 20 terdapat fitur pengecekan produk kadaluarsa yang memungkinkan sistem menampilkan produk yang telah mencapai atau mendekati tanggal kadaluarsa. Produk yang akhirnya dapat diperiksa oleh admin dan dikembalikan ke halaman utama setelah selesai.

Kode	Nama	Harga	Stok	Tanggal Expired	Status
8998800100032	Sunscreen SPF 50 PA+++	45000.0	50	2025-07-07	📌 PROMO - Exp dalam 23 hari
8998800100049	Toner Qeemla Skin	40000.0	100	2025-06-25	📌 PROMO - Exp dalam 11 hari

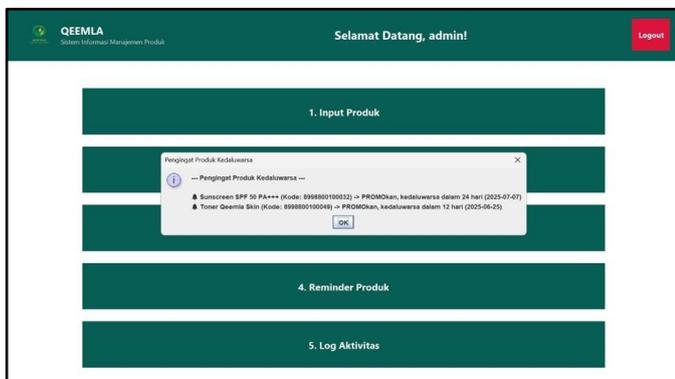
Gambar 20. Tampilan halaman log cek *expired* produk

Pada gbr. 21 terdapat Peningkat Produk yang digunakan untuk memberikan notifikasi pemberitahuan kepada admin tentang produk yang mendekati tanggal kadaluarsa. Peningkat ini membantu admin untuk lebih cepat menangani produk yang perlu diperhatikan lebih lanjut. Setelah admin membaca notifikasi, mereka dapat kembali ke halaman utama.

- Log Audit Aktivitas Pengguna: Fitur ini mencatat seluruh aktivitas pengguna secara detail, termasuk waktu pelaksanaan dan identitas pengguna seperti pada gbr. 22. Data riwayat tersebut disajikan dalam format yang terorganisir, sehingga mempermudah admin dalam memonitor kegiatan sistem, dan menjaga keamanan data.

Waktu	Aksi	Detail	Email Pengguna
2025-06-13 19:41:45.0	Tambah Produk	Produk Sunscreen SPF 50 PA+++ berhasil ditambahkan	admin
2025-06-13 16:42:28.0	Tambah Produk	Produk Toner Qeemla Skin berhasil ditambahkan	admin
2025-06-13 16:17:24.0	Tambah Produk	Produk Sunscreen SPF 50 PA+++ berhasil ditambahkan	admin
2025-06-13 14:23:14.0	Tambah Produk	Produk Facial Wash Brightening berhasil ditambahkan	admin

Gambar 22. Tampilan halaman riwayat log aktivitas

Gambar 21. Tampilan halaman dashboard dengan *Expiry Notifier* otomatis

- *Black-box testing*:

TABEL I
HASIL PENGUJIAN BLACK BOX TESTING

Test Case	Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Menekan tombol "Login" pada halaman login	Klik tombol "Login"	Apabila pengguna tidak memasukkan username atau password maka akan tampil pesan kesalahan	Berhasil	<i>Valid</i>
		Apabila pengguna memasukkan username dan password yang terdaftar maka akan dialihkan ke halaman dashboard admin	Berhasil	<i>Valid</i>
		Apabila pengguna memasukkan username dan password yang tidak terdaftar maka akan tampil pesan kesalahan	Berhasil	<i>Valid</i>
Menuju halaman "Dashboard" saat sudah login	-	Pengguna akan dialihkan ke halaman Dashboard Admin	Berhasil	<i>Valid</i>
Menekan tombol "Input Produk" pada dashboard	Mengisi form input produk dengan cara <i>scan barcode</i> yang telah dibuat dan klik "Simpan"	Data masuk ke daftar produk dengan <i>field</i> lengkap	Berhasil	<i>Valid</i>

Melihat semua produk yang tersimpan	Klik tombol “Daftar Produk” pada dashboard	Menampilkan semua produk dalam tabel dengan detail	Berhasil	Valid
Memfilter produk berdasarkan masa kadaluarsa	Klik tombol “Cek Produk Expired” pada dashboard	Menampilkan riwayat produk yang sudah kadaluarsa	Berhasil	Valid
Menjalankan aplikasi dengan produk yang mendekati <i>Expired</i>	Login ke aplikasi sebagai admin	Menampilkan popup “Peringat Produk Kadaluarsa” yang berisi daftar produk terkait	Berhasil	Valid
Menekan tombol “Remind Produk”	Klik tombol “Remind Produk”	Popup “Peringat Produk Kadaluarsa” muncul kembali dengan isi yang sama/sesuai kondisi	Berhasil	Valid
Aktivitas admin dicatat dalam log audit	Tambah/hapus produk	Sistem mencatat aktivitas ke log dengan waktu dan deskripsi	Berhasil	Valid
Menekan tombol “Logout”	Klik tombol “Logout” pada <i>sidebar</i>	Pengguna akan keluar dan diarahkan ke halaman awal login	Berhasil	Valid

Hasil uji *black-box testing* pada aplikasi manajemen inventaris untuk UMKM *Qeemla Skin & Body Care* menunjukkan bahwa seluruh fitur utama yang diimplementasikan berjalan dengan baik sesuai dengan kebutuhan pengguna. Pengujian ini dilakukan dari perspektif pengguna tanpa melihat kode sumber, sehingga validasi dilakukan berdasarkan respons sistem terhadap berbagai skenario penggunaan yang umum terjadi di lingkungan operasional UMKM.

Selama pengujian, fitur autentikasi pengguna terbukti mampu membedakan dengan tepat antara data admin yang valid dan tidak valid. Sistem memberikan notifikasi yang jelas ketika terjadi kesalahan input, sehingga keamanan akses data inventaris dapat terjaga. Pada fitur manajemen produk, seluruh proses penambahan, pembaruan, penghapusan, dan penampilan data produk berjalan lancar. Admin dapat dengan mudah melakukan input produk baik secara manual, melalui scan *barcode*, maupun dengan mengunggah gambar *barcode*. Setiap aksi yang dilakukan selalu diikuti dengan notifikasi keberhasilan atau pesan konfirmasi, sehingga pengguna dapat memastikan tindakannya telah tercatat dengan benar.

Fitur notifikasi kadaluarsa otomatis juga berjalan efektif. Sistem mampu mendeteksi dan menampilkan produk yang mendekati atau telah melewati tanggal kadaluarsa secara *real-time*. Notifikasi yang diberikan membantu admin untuk segera mengambil tindakan, seperti melakukan promosi atau menarik produk dari stok, sehingga potensi kerugian akibat produk terbuang dapat diminimalkan. Selain itu, sistem juga menyediakan log aktivitas pengguna yang mencatat setiap perubahan data beserta waktu dan identitas admin yang melakukan aksi tersebut. Hal ini meningkatkan transparansi dan akuntabilitas dalam pengelolaan inventaris.

Secara keseluruhan, hasil uji *black-box* membuktikan bahwa aplikasi ini tidak hanya mampu menggantikan proses manual yang rentan terhadap kesalahan, tetapi juga meningkatkan akurasi data inventaris dan mendukung pengambilan keputusan bisnis yang lebih baik dan terinformasi.

V. KESIMPULAN

Penelitian ini telah berhasil merancang dan mengimplementasikan sebuah aplikasi manajemen produk berbasis desktop menggunakan *Java Swing* sebagai solusi digitalisasi untuk UMKM *Qeemla Skin & Body Care*. Dengan menerapkan metodologi *waterfall* dan pendekatan Pemrograman Berorientasi Objek, sistem yang dihasilkan memiliki fungsionalitas yang teruji, mencakup manajemen data produk yang komprehensif dan sistem peringatan kadaluarsa otomatis yang efektif.

Aplikasi ini secara langsung menjawab tantangan operasional yang dihadapi UMKM dengan menggantikan proses manual yang rentan kesalahan, meningkatkan akurasi data stok, dan membantu meminimalkan kerugian finansial. Keberhasilan proyek ini menunjukkan bahwa aplikasi desktop yang dirancang dengan baik tetap menjadi solusi yang relevan dan berdaya guna untuk kebutuhan bisnis spesifik, serta menjadi langkah penting bagi UMKM dalam perjalanan transformasi digital mereka.

Namun, penelitian ini memiliki keterbatasan seperti tidak mencakup integrasi transaksi penjualan, sistem kasir, atau akses berbasis cloud dan mobile. Oleh karena itu, untuk studi selanjutnya disarankan pengembangan sistem ke arah aplikasi web-based atau mobile-based, serta penambahan fitur seperti notifikasi berbasis WhatsApp, integrasi sistem POS, dan penyimpanan berbasis cloud.

UCAPAN TERIMA KASIH

Peneliti mengucapkan terima kasih kepada UPN “Veteran” Jawa Timur dan Dosen Pembimbing atas kesempatan dan dukungan yang diberikan, yang telah memungkinkan kelancaran penyelesaian penelitian ini. Peneliti juga sangat menghargai masukan dan saran yang diberikan oleh para reviewer yang telah berkontribusi dalam meningkatkan kualitas jurnal ini.

REFERENSI

- [1] Tupan, T., et al. (2021). Strategi dan Upaya Digitalisasi UMKM di Indonesia di Era Pandemi COVID-19: Sebuah Tinjauan Literatur. *Jurnal Ekonomi dan Bisnis*, 24(1), 51-68.
- [2] Aprih Wiyono, A., & Hidayah, N. (2022). Peran Digitalisasi dalam Peningkatan Kinerja Pemasaran pada Usaha Mikro Kecil dan Menengah (UMKM). *Jurnal Ilmiah Manajemen dan Bisnis*, 7(2), 112-120.
- [3] Sari, D. P., & Purnomo, S. H. (2021). Analisis Tantangan dan Peluang Digitalisasi UMKM di Indonesia. *Jurnal Administrasi Bisnis (JAB)*, 11(2), 78-87.
- [4] Susanto, A., & Putri, D. E. (2019). Perancangan Sistem Informasi Inventaris Barang untuk Mengurangi Risiko Kesalahan Pencatatan Manual. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, 6(4), 421-428.
- [5] Ristiana, N., & Santoso, A. J. (2020). Sistem Informasi Manajemen Stok dan Monitoring Tanggal kadaluarsa Produk pada Apotek. *Jurnal Sistem Informasi (J-SI)*, 12(1), 35-42.
- [6] Gaol, C. J. L., & Astuti, H. M. (2018). Perancangan Sistem Informasi Inventory Control untuk Optimalisasi Persediaan Barang. *Jurnal Ilmiah FIFO*, 10(2), 135-144.
- [7] Gunawan, H., & Shofa, M. N. (2019). Implementasi Konsep Pemrograman Berorientasi Objek pada Aplikasi Sistem Informasi Akademik Berbasis Desktop. *Jurnal Teknoinfo*, 13(2), 73-78.
- [8] Pratama, Y. A., & Wibowo, A. (2022). Pengembangan Sistem Peringatan Dini Tanggal kadaluarsa Produk Makanan Menggunakan Notifikasi Otomatis. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 6(5), 2345-2352.
- [9] Sommerville, I. (2016). *Software Engineering*, 10th Edition. Pearson Education.
- [10] Setiawan, D., & Nugroho, E. (2018). Pentingnya Perancangan User Interface dan User Experience dalam Pengembangan Aplikasi. *Prosiding Seminar Nasional Teknologi Informasi dan Komunikasi*, 102-108.
- [11] Pressman, R. S., & Maxim, B. R. (2020). *Software Engineering: A Practitioner's Approach*, 9th Edition. McGraw-Hill Education.
- [12] Hidayat, T., & Sukadi. (2017). Analisis Pengaruh Desain Antarmuka Pengguna (User Interface) terhadap Kemudahan Penggunaan Aplikasi Mobile. *Jurnal Informatika*, 4(2), 201-209.
- [13] Mulyani, S. (2018). Peran Pengujian Perangkat Lunak dalam Menjamin Kualitas Sistem Informasi. *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, 7(1), 1-6.
- [14] Laudon, K. C., & Laudon, J. P. (2021). *Management Information Systems: Managing the Digital Firm*, 17th Edition. Pearson.
- [15] Handoko, T. H. (2017). *Manajemen Personalia dan Sumber Daya Manusia*. BPFE-Yogyakarta.