

RANCANG BANGUN SISTEM INFORMASI MANAJEMEN INVENTORI GUDANG BERBASIS MOBILE MENGGUNAKAN METODE WATERFALL PADA CV. CAHAYA MULYA ABADI

Dhiya' Ulhaq Ahmad Ramsi, Nur Cahyo Wibowo, Nambi Sembilu
Sistem Informasi, Universitas Pembangunan Nasional "Veteran" Jawa Timur
Jalan Rungkut Madya No. 1 Gunung Anyar, Surabaya, Indonesia
21082010066@student.upnjatim.ac.id

ABSTRAK

Peningkatan teknologi telah membawa dampak yang signifikan terhadap cara kerja perusahaan. Sistem informasi menjadi sangat berharga, karena dapat membantu perusahaan dalam mengelola data secara efisien. Salah satu pemanfaatan teknologi tersebut yaitu dalam pengelolaan inventori barang. CV. Cahaya Mulya Abadi memiliki permasalahan dalam proses manajemen inventori, karena proses pengelolaan kemasan tube, pot cream, pot lulur, botol, serta kresak dilakukan dengan cara staff gudang mencatat di selembar kertas, kemudian menyerahkannya ke admin. Cara tersebut menjadikan proses menjadi lambat dan tidak efisien. Untuk mengatasi hal tersebut, maka dibangunlah sebuah aplikasi mobile manajemen inventori gudang pada CV. Cahaya Mulya Abadi dengan menggunakan bahasa pemrograman Dart dan *framework* Flutter, sementara database yang digunakan adalah Firebase dan Supabase. Perancangan yang digunakan dalam pembuatan perangkat lunak dalam penelitian ini adalah metode waterfall yang digabung dengan ICONIX Process untuk pembuatan UML. Berdasarkan sistem yang telah dirancang, terdapat fitur untuk mengelola bahan baku, produk, dan pesanan. Hasil pengujian aplikasi dengan menggunakan *black box testing* dan *User Acceptance Testing* menunjukkan bahwa semua skenario terpenuhi dan sesuai dengan yang diharapkan.

Kata kunci : *flutter, ICONIX process, inventori, metode waterfall, user acceptance testing*

1. PENDAHULUAN

Teknologi informasi telah membawa perubahan besar dalam cara kerja dan memberikan dampak yang signifikan. Pengguna teknologi ini mampu mengelola data dengan lebih efisien, sehingga meningkatkan produktivitas di berbagai sektor [1].

Salah satu elemen yang krusial dalam dunia perdagangan adalah pengelolaan inventori barang. Perusahaan pada umumnya melakukan penghitungan persediaan secara berkala, baik itu setiap bulan, triwulan, enam bulan, maupun tahunan. Aktivitas penghitungan persediaan barang pada setiap perusahaan membutuhkan waktu yang relatif lama berdasarkan jumlah komoditas dan aset yang dimiliki perusahaan. Akibatnya, perusahaan hanya dapat melakukan penghitungan persediaan barang pada saat libur atau dalam situasi yang tidak ramai (*non peak season*). Kondisi ini menyebabkan perusahaan melakukan penghitungan persediaan barang secara tidak teratur. Dan juga rawan terjadi kesalahan pendataan apabila barang yang dicatat jumlahnya relatif banyak [2].

CV. Cahaya Mulya Abadi merupakan sebuah perusahaan yang didirikan pada tahun 2008. Berlokasi di Jl. Bulusidokare No. 569, Sidoarjo, Jawa Timur. Perusahaan ini bergerak di bidang supplier plastik untuk pembuatan kemasan kosmetik dan farmasi yang terbuat dari PP, PE, PS, PET, PETG dan acrylic. Selama ini, proses pengelolaan barang dilakukan dengan cara staff gudang mencatat di selembar kertas, kemudian diberikan ke admin. Kemudian, admin memasukkan satu per satu data ke dalam laporan di

Microsoft Excel. Cara tersebut menjadikan proses manajemen inventori berjalan dengan lambat, dan dapat menyebabkan kerugian bagi perusahaan. Metode pengembangan perangkat lunak yang digunakan di penelitian ini adalah metode waterfall. Sistem yang dibuat akan memiliki kualitas yang baik, karena pelaksanaannya dilakukan secara bertahap. Serta, dokumen pengembangan sistem yang dihasilkan akan terorganisir, karena setiap fase harus diselesaikan terlebih dahulu sebelum lanjut ke fase berikutnya, membuat metode ini menjadi pilihan yang cocok dalam pengembangan sistem informasi [3].

Berdasarkan permasalahan tersebut, maka dibuat sebuah sistem informasi manajemen inventori berbasis mobile dengan menggunakan bahasa pemrograman Dart, dan menggunakan *framework* Flutter, serta menggunakan Metode Waterfall. Diharapkan, dengan adanya aplikasi tersebut dapat membantu CV. Cahaya Mulya Abadi untuk melakukan pengelolaan inventori pada barang masuk dan barang keluar serta dapat mengatasi permasalahan yang terdapat pada perusahaan CV. Cahaya Mulya Abadi.

2. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Penelitian yang dilakukan merujuk pada beberapa tinjauan pustaka, yang pertama yakni mengenai *The Design Of A Stock Taking Inventory Application Based On Android*. Perbedaan terdapat pada metode yang digunakan yaitu metode SDLC (*Software Development Life Cycle*), sementara pada penelitian ini menggunakan metode *waterfall*.

Persamaan dengan penelitian ini yaitu sama-sama meneliti mengenai manajemen inventori gudang [2].

Penelitian kedua yaitu tentang Rancang Bangun Aplikasi Mobile Berbasis Android Untuk Manajemen Persediaan Barang Dan Kasir. Perbedaan terdapat pada bahasa pemrograman dan *framework* yang digunakan yaitu Java, sementara penelitian ini menggunakan Dart dengan *framework* Flutter. Persamaan dengan penelitian ini yaitu sama-sama menggunakan metode *waterfall* [1].

Penelitian ketiga yaitu tentang Rancang Bangun Pendataan Stok Barang Operasional Dengan Metode Iconix Process Berbasis Web. Perbedaan terdapat pada platform yang digunakan yaitu web, sementara penelitian ini menggunakan mobile. Persamaan dengan penelitian ini yaitu sama-sama menggunakan ICONIX Process untuk pembuatan UML (*Unified Modeling Language*) [4].

2.2. Dart

Dart adalah bahasa pemrograman *open source* yang sangat cocok untuk aplikasi mobile dan aplikasi web. Dart dapat digunakan untuk berbagai tujuan seperti aplikasi desktop, mobile, backend, dan web [5].

Bahasa, *tools*, dan API Dart memungkinkan pengembang yang inovatif, produktif, dan berbakat untuk membuat aplikasi web *scalable* yang dapat memanfaatkan web modern sebaik-baiknya [6].

2.3. Flutter

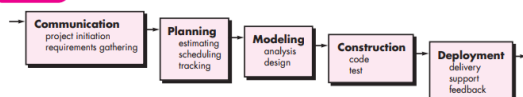
Flutter adalah SDK (Software Development Kit) untuk pengembangan aplikasi mobile, yang dibangun oleh Google dan bersifat *open source* [7].

Untuk rendering, Flutter mengandalkan Skia, sebuah mesin grafis 2D yang kompatibel dengan berbagai platform perangkat keras dan perangkat lunak [8].

2.4. Waterfall Model

Model Waterfall, atau dikenal sebagai model air terjun, sering disebut juga sebagai (*classic life cycle*). Model ini menggambarkan pendekatan pengembangan perangkat lunak yang sistematis dan berurutan (*sekuensial*), di mana setiap tahap harus diselesaikan sebelum melanjutkan ke tahap berikutnya. [9].

FIGURE 2.3 The waterfall model



Gambar 1. Model waterfall pressman [9]

Model ini sering digunakan dalam *Software Engineering* dan menerapkan pendekatan yang terstruktur serta berurutan. Prosesnya dimulai dari tahap identifikasi kebutuhan, kemudian berlanjut ke *Communication*, *Planning*, *Modeling*, *Construction*, hingga akhirnya mencapai *Deployment*. Berikut ini

adalah penjelasan dari tahap – tahap yang dilakukan di dalam Model Waterfall:

a. Communication

Tahap pertama dimulai dengan berkomunikasi dengan konsumen atau pengguna.

b. Planning

Setelah tahap *communication* selesai, langkah berikutnya adalah menyusun rencana pengembangan perangkat lunak.

c. Modeling

Dalam tahap ini, kebutuhan yang telah ditetapkan sebelumnya akan diterjemahkan ke dalam rancangan perangkat lunak.

d. Construction

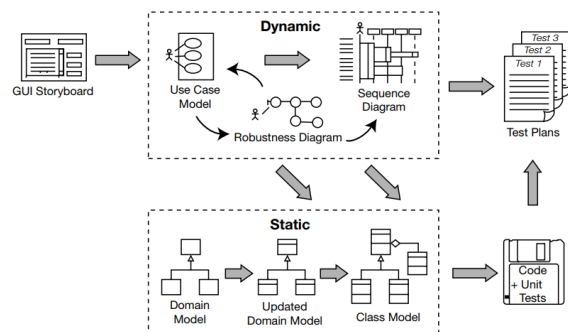
Construction adalah tahap di mana kode program dibuat melalui proses *code generation*. Setelah proses pengkodean selesai, dilakukan testing untuk menguji sistem yang telah dikembangkan.

e. Deployment

Setelah melalui proses analisis, perancangan, dan pengkodean, sistem yang telah selesai akan digunakan oleh pengguna.

2.5. ICONIX Process

Dalam pengembangan sistem informasi, diperlukan sebuah konsep dasar sebagai pedoman dalam Software Development Life Cycle (SDLC). Salah satu metode SDLC yang digunakan adalah Use Case Driven Object Modeling with UML, yang dikenal sebagai ICONIX Process [4].



Gambar 2. ICONIX process [10]

Gambar 2 menampilkan ICONIX Process yang terdiri dari empat tahap [10]. Tahapan-tahapan tersebut, yaitu:

1. Requirements

a. Functional Requirements

Kegiatan dimana data dikumpulkan dan diproses sesuai dengan kebutuhan fungsional berdasarkan informasi yang didapatkan dari observasi dan wawancara.

b. Domain Modeling

Tahap dalam aspek statis UML di mana data yang diperoleh dari kebutuhan fungsional dan non-fungsional akan diekstrak menjadi beberapa bagian, lalu dihubungkan sesuai dengan kebutuhan perangkat lunak.

- c. *Behavioral Requirements*
Tahapan pada bagian ini perlu mendefinisikan interaksi antara sistem dan pengguna.
2. *Analysis/Preliminary Design*
 - a. *Robustness Analysis*
Pengembangan dari tahapan analisa kemudian dilakukan proses tahapan desain.
3. *Detailed Design*
 - a. *Sequence Diagram*
Tahapan pemodelan sequence diagram dimana disusun terhadap diagram alir yang dilanjutkan dari tahapan robustness diagram.
 - b. *Class Diagram*
Tahapan pemodelan class diagram menggambarkan secara jelas struktur, atribut, kelas, hubungan, dan metode setiap objek. Diagram kelas menyediakan data sebagai hubungan antar kelas.
4. *Implementation*
 - a. *Coding/Unit Testing*
Tahapan dimana dimulainya pengimplementasian sistem dan akan dilakukan proses pengujian kode atau unit tests setelah pengembangan model yang dirancang sebelumnya.
 - b. *Integration and Scenario Testing*
Tahapan dimana dilakukan pengujian sesuai dengan use cases, agar pengujian sesuai dengan kondisi yang diinginkan.
5. *Code Review and Model Update*
Pada tahapan ini dilakukan peninjauan terhadap sistem untuk mempersiapkan pengembangan sistem selanjutnya.

3. METODE PENELITIAN

Metode yang digunakan dalam penelitian ini adalah metode Waterfall. Metode tersebut digunakan dalam mengembangkan sistem manajemen inventori berbasis Android. Adapun langkah-langkahnya dilakukan sebagai berikut:

- a. *Communication*
Tahapan awal pengerjaan penelitian adalah wawancara dengan staff gudang, staff admin, dan staff marketing dari pihak CV. Cahaya Mulya Abadi untuk mengumpulkan informasi serta data yang dibutuhkan untuk merancang dan membangun aplikasi Sistem Informasi Manajemen Inventori. Pada saat ini, pencatatan barang dilakukan secara manual, sehingga sering terdapat data yang tidak akurat.
- b. *Planning*
Pada tahap ini, dilakukan perencanaan tentang rencana pengerjaan aplikasi, sumber daya yang dibutuhkan, hasil yang akan dibuat, dan membuat jadwal pengerjaan aplikasi.
- c. *Modeling*
Tahap ini dilakukan perancangan struktur data, tampilan aplikasi, dan algoritma program. Hal tersebut bertujuan untuk memperoleh gambaran besar dari apa yang akan dikerjakan. Penerapan

model ini menggunakan tahap Requirements, Analysis and Preliminary Design dan Detailed Design pada ICONIX Process.

- d. *Construction*

Tahap Construction merupakan tahap membangun aplikasi sesuai dengan analisa kebutuhan dan desain untuk membuat aplikasi Sistem Informasi Manajemen Inventori berbasis mobile. Dengan menggunakan bahasa pemrograman Dart, serta framework Flutter. Penerapan model ini menggunakan tahap Implementation dari ICONIX Process.

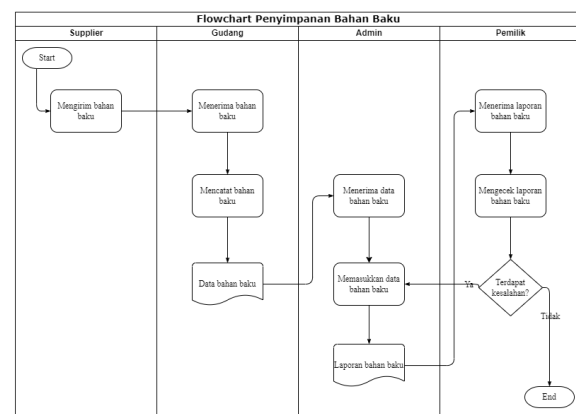
- e. *Deployment*

Tahap deployment merupakan tahap implementasi aplikasi kepada pihak CV. Cahaya Mulya Abadi.

4. HASIL DAN PEMBAHASAN

4.1. Communication

Informasi yang digunakan pada penelitian ini didapatkan secara langsung melalui wawancara dan observasi terhadap objek terkait dengan penelitian secara langsung. Berdasarkan wawancara yang telah dilakukan, didapatkan hasil proses bisnis lama yang digambarkan dengan menggunakan *Cross Functional Flowchart* (CFF).



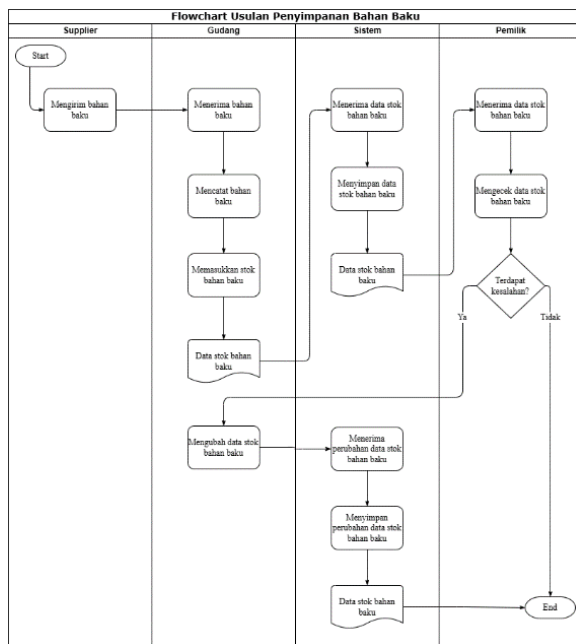
Gambar 3. CFF proses penyimpanan bahan baku saat ini

Gambar 3 menampilkan CFF dari proses penyimpanan bahan baku saat ini. Proses dimulai dari *supplier* yang mengirim bahan baku, lalu diterima oleh staff gudang. Staff gudang akan mencatat apa saja bahan baku yang diterima, kemudian mengirimkannya ke admin. Admin akan memasukkan data bahan baku tersebut ke laporan, dan akan mengirimkan laporan tersebut ke pemilik. Pemilik menerima laporan tersebut, dan akan melakukan pengecekan. Jika terdapat kesalahan, maka admin perlu melakukan revisi. Jika tidak, maka proses telah selesai.

4.2. Planning

Berdasarkan hasil wawancara, langkah selanjutnya yaitu melakukan perencanaan rancang bangun aplikasi. Maka dari itu, dirancanglah sebuah

usulan alur proses bisnis yang baru dengan menggunakan CFF.



Gambar 4. Usulan CFF proses penyimpanan bahan baku

Gambar 4 menampilkan usulan CFF dari proses penyimpanan bahan baku. Proses dimulai dari *supplier* yang mengirim bahan baku, lalu diterima oleh staff gudang. Staff gudang akan mencatat apa saja bahan baku yang diterima, kemudian memasukkan stok bahan baku ke dalam sistem. Sistem akan menerima dan menyimpan data stok bahan baku tersebut. Kemudian, pemilik menerima data stok bahan baku, dan melakukan pengecekan apakah terdapat kesalahan. Jika ya, maka staff gudang perlu mengubah data stok bahan baku. Kemudian, sistem akan menerima dan menyimpan perubahan data stok bahan baku. Jika tidak, maka proses sudah selesai.

4.3. Modeling

4.3.1. Requirements

a. *Functional Requirements*

Tabel 1. *Functional requirements*

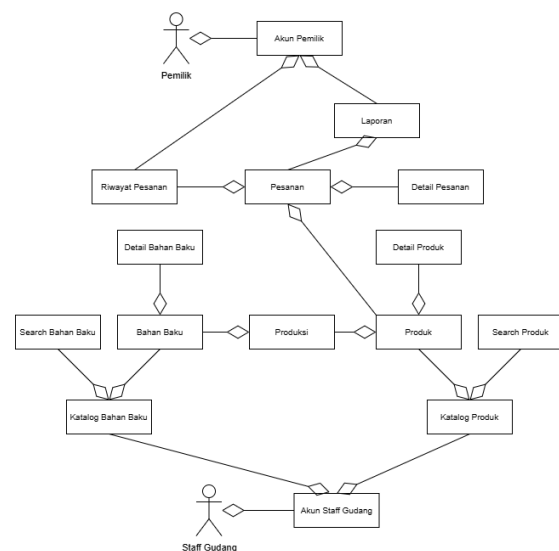
Code	Fitur
FIT-1	Login
FIT-2	Melihat daftar bahan baku
FIT-3	Mencari bahan baku
FIT-4	Menambah bahan baku
FIT-5	Mengubah bahan baku
FIT-6	Menghapus bahan baku
FIT-7	Mengubah stok bahan baku
FIT-8	Melihat daftar request bahan baku
FIT-9	Mencari request bahan baku
FIT-10	Menambah request bahan baku
FIT-11	Menghapus request bahan baku
FIT-12	Approve request bahan baku
FIT-13	Melihat daftar produk
FIT-14	Mencari produk

Kode	Fitur
FIT-15	Menambah produk
FIT-16	Mengubah produk
FIT-17	Menghapus produk
FIT-18	Mengubah stok produk
FIT-19	Melihat daftar pesanan
FIT-20	Mencari pesanan
FIT-21	Menambah pesanan
FIT-22	Mengubah pesanan
FIT-23	Menghapus pesanan
FIT-24	Melihat detail pesanan
FIT-25	Melihat laporan bahan baku dan produk
FIT-26	Melihat laporan pesanan
FIT-27	Melihat profile
FIT-28	Mengubah profile
FIT-29	Mengatur Alert bahan baku
FIT-30	Melihat Alert Bahan Baku

Tabel 1 merupakan Hasil dari analisis dan specification of requirements dari informasi yang didapatkan sebelumnya, terdapat dua aktor yang menggunakan aplikasi ini, yaitu staff gudang dan pemilik. User dapat login, mengelola bahan baku, mengelola request bahan baku, mengelola pesanan, melihat dan mengubah profile, serta mengatur dan melihat alert bahan baku.

b. *Domain Modeling*

Berikut merupakan rancangan *domain model* yang telah dibuat dari aplikasi.



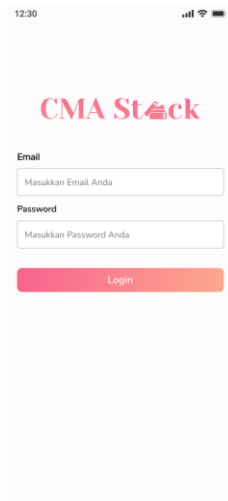
Gambar 5. Domain model

Pada domain model Gambar 5, terdapat 2 pengguna yaitu staff gudang dan pemilik. Staff gudang memiliki akun staff gudang yang mempunyai akses ke katalog bahan baku dan katalog produk. Katalog bahan baku berisi search bahan baku dan bahan baku, kemudian bahan baku berisi detail bahan baku. Sementara katalog produk berisi search produk dan produk, kemudian produk berisi detail produk. Produksi menggunakan bahan baku, lalu hasil produksi menjadi sebuah produk. Kemudian pesanan berisi produk. Pemilik memiliki akun pemilik yang

mempunyai akses ke laporan dan riwayat pesanan. Pesanan berisi riwayat pesanan dan detail pesanan. Sementara laporan berisi pesanan.

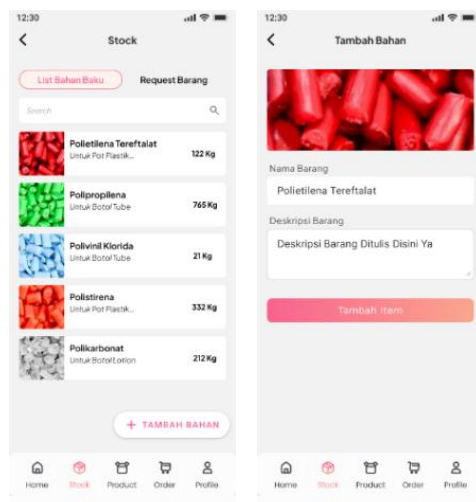
c. Behavioral Requirements

Berikut merupakan beberapa gambar dari GUI Storyboard pada aplikasi yang akan dibuat.



Gambar 6. GUI login

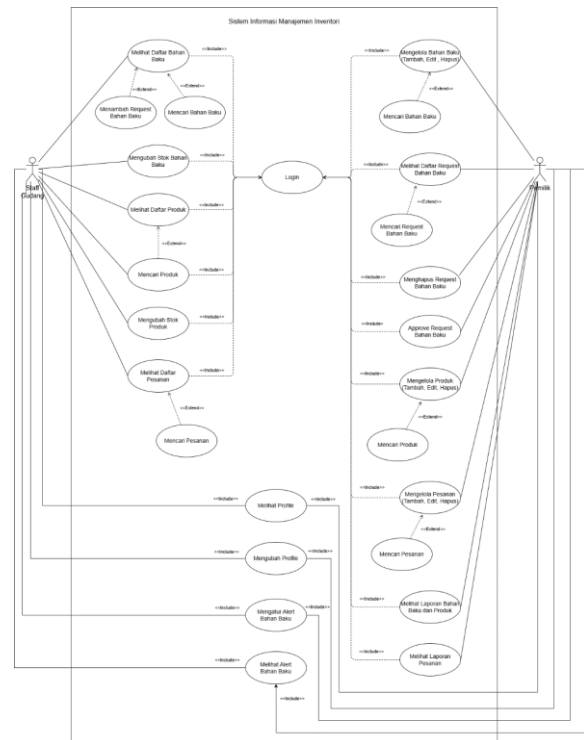
Gambar 6 merupakan tampilan antar muka dari login. User perlu memasukkan email dan password untuk melakukan login.



Gambar 7. GUI menambah bahan baku

Untuk menambah bahan baku, pemilik dapat menggunakan fitur tambah bahan baku seperti pada Gambar 7 bagian kiri, kemudian pada Gambar 7 bagian kanan, pemilik dapat memasukkan foto, nama, dan deskripsi bahan baku.

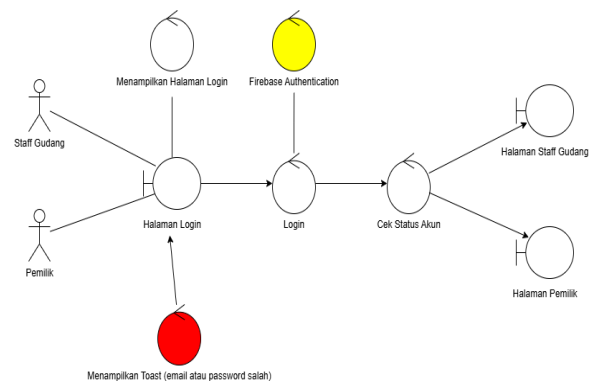
Gambar 8 merupakan Use Case Diagram yang memperlihatkan relasi antara aktor dengan masing-masing Use Case dan relasi antar Use Case.



Gambar 8. Use case diagram

4.3.2. Analysis/Preliminary Design

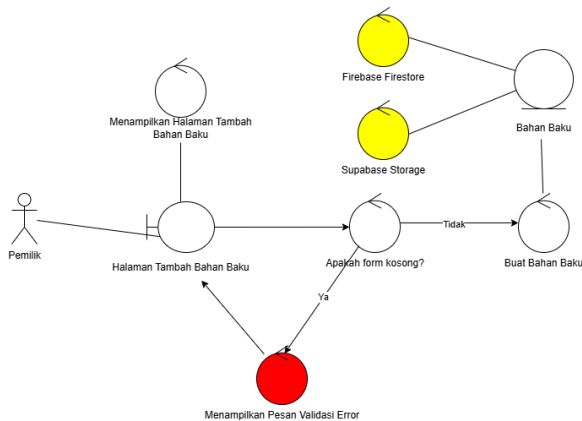
a. Robustness Analysis



Gambar 9. Robustness login

Gambar 9 menampilkan proses login dari staff gudang dan pemilik dengan mengisi email dan password pada form login. Data login tersebut akan dikirimkan ke Firebase Authentication untuk memvalidasi data, setelah Firebase Authentication memvalidasi kecocokan data, proses akan dilanjutkan dengan mengarahkan ke halaman sesuai role yang ada.

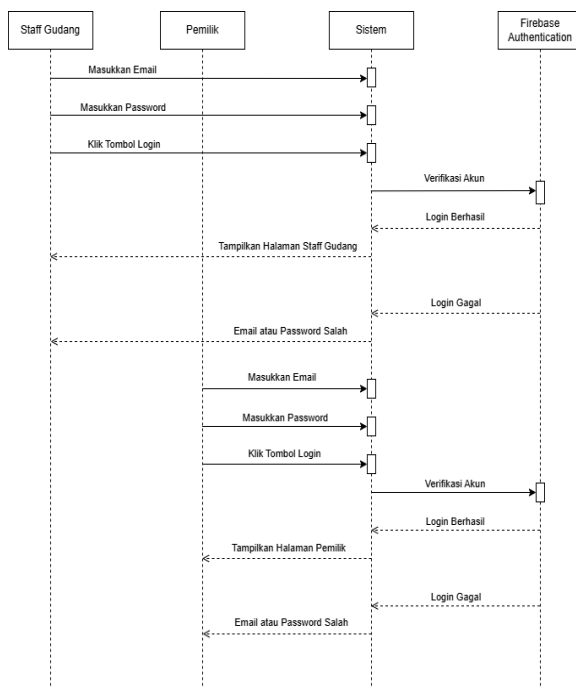
Gambar 10 menampilkan proses menambah bahan baku, dimana pemilik dapat mengakses halaman tambah bahan baku. Kemudian, sistem akan mengecek apakah terdapat form kosong. Jika iya, maka sistem akan menampilkan pesan validasi error. Jika tidak, maka sistem akan menambahkan bahan baku ke dalam Firebase, serta gambar bahan baku ke Supabase.



Gambar 10. Robustness menambah bahan baku

4.3.3. Detailed Design

a. Sequence Diagram

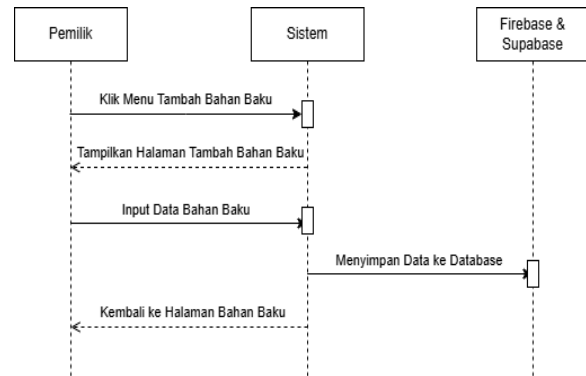


Gambar 11. Sequence diagram login

Gambar 11 menampilkan sequence diagram login, dimana terdapat beberapa objek, yaitu staff gudang dan pemilik sebagai aktor, sistem, dan Firebase. Dua aktor tersebut memiliki akses yang berbeda. Setelah memasukkan email dan password, aktor dapat menekan tombol login. Sistem akan melakukan pengecekan ke Firebase Authentication, apakah verifikasi akun berhasil atau tidak. Jika tidak, Firebase Authentication akan memberikan respon login gagal, dan sistem akan menampilkan pesan email atau password salah. Jika login berhasil, Firebase Authentication akan memberikan respon login

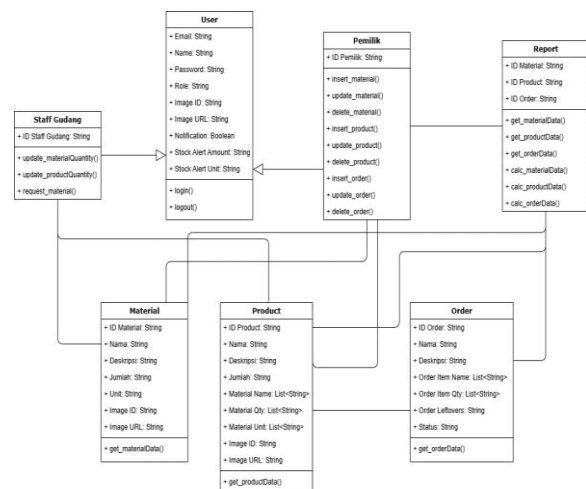
berhasil, dan sistem akan menampilkan halaman utama untuk masing-masing user.

Gambar 12 menampilkan sequence diagram menambah bahan baku, dimana terdapat tiga objek, yaitu pemilik sebagai aktor, sistem, serta Firebase dan Supabase sebagai database. Aktor dapat mengakses menu tambah bahan baku. Sistem akan menampilkan halaman tambah bahan baku, setelah itu aktor perlu memasukkan data bahan baku. Data tersebut akan disimpan ke dalam Firebase dan gambar bahan baku ke Supabase, sementara sistem akan mengembalikan aktor ke halaman bahan baku.



Gambar 12. Sequence diagram menambah bahan baku

b. Class Diagram



Gambar 13. Class diagram

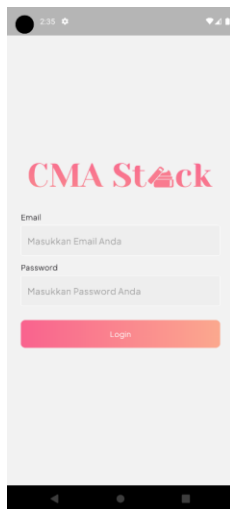
Gambar 13 menunjukkan class diagram berdasarkan use case diagram yang telah dibuat sebelumnya. Terdapat beberapa entitas, yaitu entitas user, entitas staff gudang, entitas pemilik, entitas bahan baku, entitas produk, entitas pesanan, dan entitas laporan.

4.4. Construction

4.4.1. Implementation

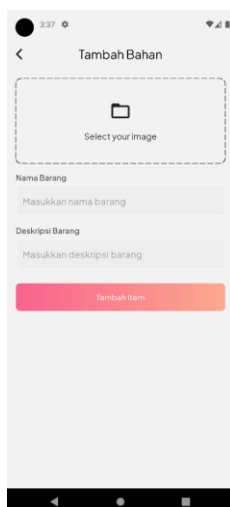
a. Coding

Pada tahap ini, akan ditampilkan hasil dari program yang telah dibuat.



Gambar 14. Tampilan antarmuka login

Gambar 14 merupakan tampilan antarmuka dari login. Terdapat kolom untuk user memasukkan email dan password.



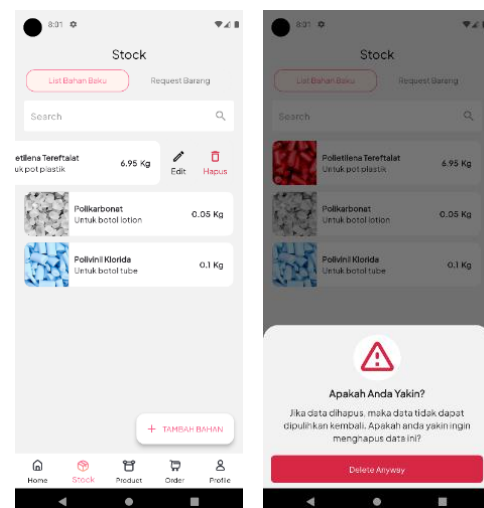
Gambar 15. Tampilan antarmuka menambah bahan baku

Gambar 15 merupakan tampilan antarmuka dari menambah bahan baku. Halaman tersebut berisi kolom untuk mengisi gambar, nama, dan deskripsi bahan baku.



Gambar 16. Tampilan antarmuka mengubah bahan baku

Gambar 16 merupakan tampilan antarmuka dari mengubah bahan baku. Halaman tersebut berisi kolom untuk mengubah gambar, nama, dan deskripsi bahan baku.



Gambar 17. Tampilan antarmuka menghapus bahan baku

Gambar 17 merupakan tampilan antarmuka dari menghapus bahan baku. User perlu melakukan swipe kiri pada salah satu bahan baku, kemudian klik tombol hapus. Setelah itu akan muncul alert hapus bahan baku, jika menekan tombol delete anyway, maka bahan baku tersebut akan terhapus dari database.

b. Testing

Pengujian aplikasi CMA Stock dilakukan menggunakan black box testing dengan berbagai macam skenario, yang didasarkan pada use case yang telah dibuat sebelumnya.

Tabel 2. Pengujian login

Skenario	Pengguna	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
Pengguna masuk ke halaman login	Pemilik	Aplikasi menampilkan halaman login	Sesuai yang diharapkan	Terpenuhi
	Staff Gudang			
Pengguna input kolom email dan password dengan benar	Pemilik	Aplikasi akan memproses, dan akan meneruskan ke halaman utama	Sesuai yang diharapkan	Terpenuhi
	Staff Gudang			
Pengguna melakukan salah input pada kolom email dan password	Pemilik	Aplikasi menampilkan pesan pop up error	Sesuai yang diharapkan	Terpenuhi
	Staff Gudang			
Terdapat kolom yang belum diisi	Pemilik	Aplikasi menampilkan pesan pop up error	Sesuai yang diharapkan	Terpenuhi
	Staff Gudang			

Tabel 2 merupakan pengujian terhadap login. Terdapat 4 skenario, hasil dari pengujian tersebut terpenuhi dan sesuai dengan yang diharapkan.

Tabel 3. Pengujian menambah bahan baku

Skenario	Pengguna	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
Pengguna masuk ke halaman tambah bahan baku	Pemilik	Aplikasi menampilkan halaman tambah bahan baku	Sesuai yang diharapkan	Terpenuhi
Pengguna input seluruh form tambah bahan baku, kemudian klik simpan	Pemilik	Aplikasi berhasil menyimpan data bahan baku, lalu muncul pop up “Tambah Barang Berhasil”	Sesuai yang diharapkan	Terpenuhi
Kolom input nama barang belum diisi	Pemilik	Aplikasi menampilkan pesan error “Nama Barang Perlu Diisi!”	Sesuai yang diharapkan	Terpenuhi

Tabel 3 merupakan pengujian terhadap menambah bahan baku. Terdapat 3 skenario, hasil dari pengujian tersebut terpenuhi dan sesuai dengan yang diharapkan.

4.5. Deployment

Tahap terakhir dari metode waterfall adalah

tahap deployment. Pada tahap ini, dilakukan build pada project flutter yang telah dibuat sebelumnya agar mendapatkan file apk. File tersebut akan dikirimkan melalui link google drive, yang akan dibagikan dan diunduh oleh hp android yang ada pada CV. Cahaya Mulya Abadi.

Tabel 4. User Acceptance Testing

Skenario sistem	Pengguna	Sesuai dengan kebutuhan pengguna	Dapat membantu permasalahan pengguna
Login	Pemilik	Ya	Ya
	Staff Gudang		
Menambah Bahan Baku	Pemilik	Ya	Ya
Mengubah Bahan Baku	Pemilik	Ya	Ya
Menghapus Bahan Baku	Pemilik	Ya	Ya
Mengubah Stok Bahan Baku	Staff Gudang	Ya	Ya

Tabel 4 menunjukkan pengujian terhadap pengguna. Hasil dari pengujian UAT menunjukkan bahwa sistem dapat memenuhi kebutuhan pengguna dan dapat membantu dalam menyelesaikan masalah pengguna.

5. KESIMPULAN DAN SARAN

Berdasarkan penelitian yang telah dilakukan, maka dapat ditarik kesimpulan yaitu, perancangan sistem diawali dengan melakukan wawancara dan observasi kepada mitra CV. Cahaya Mulya Abadi, serta mencari penelitian terdahulu yang sesuai dengan penelitian ini. Penelitian ini dibuat dengan menggunakan model waterfall, serta ICONIX Process untuk pembuatan UML. Kemudian, sistem yang dihasilkan memiliki fitur untuk mengelola bahan baku, produk, dan pesanan. Sementara itu, sistem diuji

dengan menggunakan *Black Box Testing* dan *User Acceptance Testing* (UAT). Sistem dibuat dengan menggunakan bahasa pemrograman Dart dan framework UI milik Google yaitu Flutter. Database yang digunakan dalam sistem ini yaitu Firebase Firestore dan Supabase Storage. Alasan digunakan database tersebut adalah karena integrasi yang mudah serta dapat menyimpan dan menampilkan data secara *realtime*. Berikut merupakan saran yang diberikan yaitu, sistem informasi ini masih dalam tahap awal. Diharapkan dapat dikembangkan lebih baik lagi dengan menambahkan beberapa fitur pada aplikasi. Seperti pada *role* pemilik dapat ditambahkan fitur mengelola *supplier* dan mengelola pendapatan serta pengeluaran keuangan perusahaan.

DAFTAR PUSTAKA

- [1] M. I. Syahib, "RANCANG BANGUN APLIKASI MOBILE BERBASIS ANDROID UNTUK MANAJEMEN PERSEDIAAN BARANG DAN KASIR," *Jurnal Informatika Teknologi dan Sains (Jinteks)*, vol. 6, no. 1, pp. 1–7, 2024.
- [2] R. Haerani and P. Desianasari, "The Design Of A Stock Taking Inventory Application Based On Android," *JURTEKSI (Jurnal Teknologi dan Sistem Informasi)*, vol. 8, no. 3, pp. 313–320, 2022.
- [3] A. A. Wahid, "Analisis metode waterfall untuk pengembangan sistem informasi," *J. Ilmu-ilmu Inform. dan Manaj. STMIK*, no. November, vol. 1, no. 1, pp. 1–5, 2020.
- [4] W. A. Pratama and Z. Sitorus, "Rancang Bangun Pendataan Stok Barang Operasional Dengan Metode Iconix Process Di HRP Foto Copy Berbasis Web," *Bulletin of Information Technology (BIT)*, vol. 4, no. 1, pp. 105–111, 2023.
- [5] S. Sinha, *Quick Start Guide to Dart Programming: Create High-Performance Applications for the Web and Mobile*. Apress, 2019. [Online]. Available: <https://books.google.co.id/books?id=6RfBDwAAQBAJ>
- [6] D. Mitchell, S. Akopkokhyants, and I. Balbaert, *Dart: Scalable Application Development*. Packt Publishing, 2017. [Online]. Available: <https://books.google.co.id/books?id=IHg5DwAAQBAJ>
- [7] E. Windmill, *Flutter in Action*. Manning, 2020. [Online]. Available: <https://books.google.co.id/books?id=EzgzEAAQBAJ>
- [8] M. L. Napoli, *Beginning Flutter: A Hands On Guide to App Development*. Wiley, 2019. [Online]. Available: <https://books.google.co.id/books?id=ex-tDwAAQBAJ>
- [9] R. S. Pressman, *Software Engineering: A Practitioner's Approach*, 7th ed. New York: McGraw-Hill Higher Education, 2010.
- [10] D. Rosenberg and M. Stephens, *Use Case Driven Object Modeling with UML: Theory and Practice*. New York: Springer, 2007.