

PERANCANGAN DAN IMPLEMENTASI SISTEM INVENTORY MANAGEMENT GUDANG BERBASIS WEBSITE MENGGUNAKAN FIRESTORE DATABASE

Moh Raihan Romadhoni, Nur Cahyo Wibowo, Dhian Satria Yudha Kartika

Sistem Informasi, Universitas Pembangunan Nasional "Veteran" Jawa Timur

Jalan Raya Rungkut Madya, No. 1, Gunung Anyar, Surabaya, Indonesia

mohraihanromadhoni16@gmail.com

ABSTRAK

Sektor manufaktur, yang merupakan pilar utama pembangunan ekonomi, kini mengalami perubahan besar berkat inovasi digital terkini yang dikenal sebagai "Industri 4.0." Transformasi ini telah mengubah secara signifikan cara operasional bisnis dijalankan. CV. Cahaya Mulya Abadi merupakan perusahaan manufaktur yang mengalami permasalahan dalam pengelolaan stok barang yang masih dilakukan secara manual, sehingga sering terjadi keterlambatan pencatatan dan ketidaksesuaian data stok. Penelitian ini bertujuan untuk merancang dan membangun sistem informasi manajemen inventory berbasis website guna memudahkan pengelolaan stok barang. Metode pengembangan sistem yang digunakan adalah model V-Model dengan tahapan analisis kebutuhan, perancangan sistem, implementasi, dan pengujian. Sistem dikembangkan menggunakan framework Laravel dengan database Firestore, dan diuji melalui unit testing, integration testing, system testing, serta acceptance testing. Hasil pengujian menunjukkan seluruh fungsionalitas sistem berjalan dengan baik (pass), serta mampu mendukung pengelolaan stok secara online melalui berbagai perangkat. Dengan diterapkannya sistem ini, diharapkan proses manajemen inventory di CV. Cahaya Mulya Abadi menjadi lebih cepat, akurat, dan efisien.

Kata kunci : Website, Manajemen, Gudang, Manufaktur, V Model

1. PENDAHULUAN

Perkembangan teknologi informasi dan komunikasi telah mendorong perusahaan untuk mengadopsi teknologi demi meningkatkan daya saing. Globalisasi memperluas pasar dan menjadikan bisnis semakin kompetitif, sehingga persaingan pun semakin kompleks. Dalam kondisi ini, penerapan sistem informasi yang efektif sangat penting untuk menunjang kesejahteraan dan keberlangsungan hidup perusahaan [1]. Sektor manufaktur, yang merupakan pilar utama pembangunan ekonomi, kini mengalami perubahan besar berkat inovasi digital terkini yang dikenal sebagai "Industri 4.0." Transformasi ini telah mengubah secara signifikan cara operasional bisnis dijalankan [2].

CV. Cahaya Mulya Abadi adalah perusahaan manufaktur yang berdiri sejak tahun 2012 dan berfokus pada produksi kemasan untuk produk kecantikan. Perusahaan memproduksi berbagai jenis kemasan seperti botol, jar dan tube yang berbahan dasar plastik. Dalam menjalankan proses bisnisnya, perusahaan mengandalkan Spreadsheet untuk mencatat stok dan keluar masuk barang di gudang. Namun metode pencatatan ini sering menimbulkan berbagai kendala, termasuk kesulitan memantau stok secara real-time dan risiko kesalahan yang dapat menyebabkan ketidaksesuaian antara stok fisik dan catatan administrasi. Selain itu, ketergantungan pada spreadsheet juga menyulitkan dalam hal pelaporan, karena proses rekapitulasi data sering kali memakan waktu dan rentan terhadap kesalahan manusia.

Untuk mengatasi masalah tersebut, solusi yang diusulkan adalah pengembangan sistem *inventory* gudang yang dirancang khusus untuk membantu admin gudang dalam menjalankan tugas-tugas

operasional. Sistem akan dikembangkan menggunakan platform website agar dapat diakses dengan mudah dan fleksibel oleh pengguna. Sistem dirancang untuk mengatasi permasalahan kesulitan dalam pengorganisasian dan pengecekan stok. Fitur utama dari sistem ini adalah pengelolaan data stok secara real-time, di mana setiap penambahan dan pengurangan stok akan terintegrasi dengan data barang masuk, barang keluar, serta retur barang. Selain itu, sistem mencakup pengelolaan pemesanan, sehingga admin gudang dapat mencatat status pesanan dari awal hingga selesai.

Pengembangan sistem *inventory* berbasis website untuk manajemen *inventory* gudang telah dilakukan oleh beberapa peneliti terdahulu. Salah satunya adalah penelitian yang berjudul " *Designing a Web-Based Inventory Application at General Steel Supplier Using Extreme Programming Method*" [3]. Hasil penelitian ini menghasilkan aplikasi *inventory* berbasis web yang berhasil membantu General Steel Supplier dalam mengelola stok barang dengan lebih efisien dan terstruktur. Aplikasi ini memudahkan pencatatan barang masuk, keluar, dan retur, serta mempercepat proses pembuatan laporan. Penelitian lain yang berjudul "Sistem Informasi Inventaris Barang Berbasis Website Menggunakan Metode Garis Lurus (Studi Kasus: Rumah Sakit Jiwa Prof. Dr. M. Ildrem)". Hasil penelitian ini berupa sistem yang dikembangkan dapat mencatat inventaris secara lengkap, meliputi berbagai aspek seperti pengadaan, penempatan, mutasi, peminjaman, pengembalian, serta pelaporan kerusakan dan perbaikan barang [4].

Metode pengembangan yang digunakan dalam penelitian ini adalah V-Model, sebuah pendekatan pengembangan perangkat lunak yang menekankan

validasi dan verifikasi di setiap tahap siklus hidup pengembangan sistem (SDLC) [5]. V-Model dipilih karena keunggulannya dalam memberikan struktur yang jelas serta memungkinkan pengujian yang paralel pada setiap tahap pengembangan. Metode ini menitikberatkan pada pengujian di setiap tahap siklus pengembangan, dengan keterlibatan pengguna dalam proses pengujian penerimaan sistem pada tahap akhir [6].

Dalam Pengembangan sistem ini, digunakan laravel, sebuah framework PHP yang memudahkan dan mempercepat pengembangan aplikasi web. Laravel mengadopsi arsitektur MVC (*Model-View-Controller*) yang memisahkan logika aplikasi, antarmuka pengguna, dan pengelolaan data, sehingga pengembangan lebih terstruktur dan mudah dikelola [7]. Laravel menyediakan berbagai fitur utama seperti routing yang mudah, *Eloquent ORM* untuk mengelola database secara intuitif, migration untuk pengaturan skema database dan blade templating untuk pembuatan antar muka yang dinamis [8].

Berdasarkan latar belakang tersebut, tujuan dari penelitian ini adalah untuk membangun sistem yang dapat memudahkan admin gudang dalam memonitor stok barang dan pergerakan inventory yang dikelola oleh CV. Cahaya Mulya Abadi. Melalui sistem ini, admin gudang diharapkan dapat dengan mudah mengakses informasi terkait stok, memantau pergerakan barang masuk dan keluar, serta menghasilkan laporan secara real-time.

2. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Penelitian terkait sistem *inventory management* telah banyak dilakukan. Penelitian pertama berjudul *Sistem Informasi Inventory Gudang (SI-IGUN) pada CV. Mega Kencana* menghasilkan sistem yang meningkatkan efisiensi dan akurasi pengelolaan inventaris dengan mengurangi kesalahan input manual serta mengintegrasikan data secara real-time [9]. Penelitian kedua, *Rancang Bangun Aplikasi Android Mysmartfishfeeding Menggunakan Flutterflow dan Firebase Firestore*, menunjukkan bahwa penggunaan Firebase Firestore memiliki tingkat keakuratan data sebesar 100%, membuktikan aplikasi mampu menampilkan data sesuai hasil tangkapan alat [10]. Penelitian ketiga, *Pengembangan Sistem Inventory Barang Perusahaan Dagang Berbasis Website (CV. Agung Nugraha)*, menghasilkan sistem yang berfungsi dengan baik berdasarkan uji Black-Box dan memiliki tingkat penerimaan pengguna sebesar 86,25%, serta mampu meningkatkan efisiensi kerja [11]. Berdasarkan penelitian-penelitian tersebut, penggunaan Firebase Firestore sangat cocok untuk mendukung sistem *inventory management* karena mampu menyajikan data secara real-time dengan tingkat akurasi yang tinggi. Selain itu, metode pengujian *Black-Box* terbukti efektif dalam mengevaluasi fungsionalitas sistem.

2.2. Sistem Inventory

Sistem *inventory* melibatkan pengelolaan, pengawasan, dan pengendalian stok barang dalam bisnis, dengan tujuan memastikan persediaan selalu mencukupi untuk memenuhi permintaan pelanggan sekaligus mengurangi biaya yang timbul akibat kelebihan atau kekurangan stok [12]. Sistem ini memungkinkan perusahaan untuk memantau stok barang, mengatur pembelian, serta mengelola persediaan secara efisien. Dalam penerapannya, penting untuk memperhatikan kebutuhan integrasi dengan sistem keuangan dan penjualan. Selain itu, pemilihan sistem yang sesuai dengan skala dan jenis usaha sangatlah krusial.

Dalam pengelolaan persediaan barang, terdapat beberapa komponen penting yang harus diperhatikan untuk memastikan sistem *inventory* berfungsi dengan baik. Komponen-komponen ini mencakup berbagai aspek yang saling terkait, dan setiap elemen memiliki peran krusial dalam meningkatkan efisiensi dan efektivitas manajemen persediaan.

2.3. Firebase

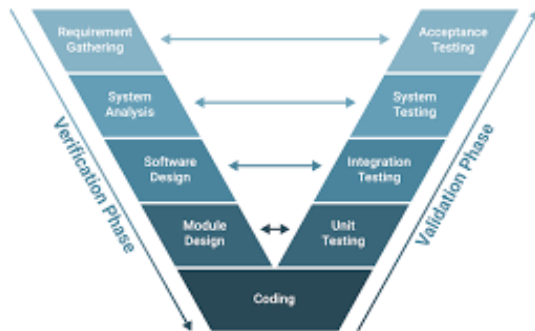
Firebase merupakan platform Backend as a Service (BaaS) milik Google yang dirancang untuk memudahkan pengembang dalam membangun aplikasi mobile dan web. Platform ini pertama kali didirikan pada tahun 2011 oleh Andrew Lee dan James Tamplin, sebelum akhirnya diakuisisi oleh Google pada tahun 2014 [10]. Berkat berbagai fitur yang tersedia, Firebase memungkinkan pengembang untuk lebih fokus pada pengembangan logika aplikasi tanpa harus repot mengelola infrastruktur server secara langsung.

Salah satu fitur andalannya adalah Realtime Database, yang menyimpan data dalam format JSON dan memungkinkan sinkronisasi data secara real-time ke seluruh klien yang terhubung. Fitur ini sangat ideal untuk aplikasi yang membutuhkan pembaruan data secara instan, seperti aplikasi pesan instan. Selain itu, Realtime Database juga mendukung penyimpanan data secara lokal ketika perangkat dalam kondisi offline, sehingga data dapat disinkronkan kembali saat koneksi internet tersedia [13].

2.4. V-Model

V Model, atau Verification and Validation Model, adalah salah satu metodologi dalam Software Development Life Cycle (SDLC) yang menggunakan pendekatan terstruktur untuk proses pengembangan perangkat lunak, serta merupakan pengembangan dari model waterfall [14]. Model ini menggambarkan hubungan antara aktivitas penjaminan kualitas (*quality assurance*) dengan proses komunikasi, pemodelan, dan pengembangan awal. Di sisi kiri huruf V, tim pengembang menyusun kebutuhan utama menjadi bentuk yang lebih rinci dan teknis untuk memahami masalah dan solusinya [15]. Setelah tahap pengkodean selesai, tim kemudian beralih ke sisi kanan V untuk melakukan serangkaian pengujian (*quality assurance*)

yang bertujuan memvalidasi setiap model yang telah dibuat sebelumnya di sisi kiri. Secara umum, siklus hidup pengembangan perangkat lunak pada model ini tidak berbeda jauh dengan model klasik, namun Model V memberikan representasi visual yang lebih jelas tentang bagaimana verifikasi dan validasi dilakukan pada tiap tahap pengembangan.



Gambar 1. Metode V-Model

2.5. Black Box Testing

Black box testing adalah metode pengujian perangkat lunak yang menilai fungsionalitas sistem tanpa memperhatikan struktur internal atau kode sumbernya. Dalam metode ini, penguji hanya berfokus pada input yang diberikan dan output yang dihasilkan, tanpa mengetahui proses yang terjadi di dalam sistem. Tujuan utama dari pengujian ini adalah memastikan bahwa perangkat lunak berjalan sesuai dengan spesifikasi yang ditetapkan, dengan cara menguji berbagai kombinasi input dan memeriksa apakah hasil keluarannya sesuai dengan yang diharapkan[16]. Pengujian sistem menggunakan pendekatan black box dapat mengungkap berbagai jenis kesalahan, termasuk fungsi yang tidak sesuai atau hilang, masalah pada antarmuka pengguna, kesalahan dalam struktur data atau interaksi dengan database eksternal, gangguan pada kinerja, serta masalah dalam proses inisialisasi dan terminasi. Teknik ini dilakukan dengan menggunakan test case yang secara khusus dirancang untuk mendukung metode black box testing.

2.5 Laravel

Laravel adalah salah satu framework PHP paling populer untuk pengembangan aplikasi web modern. Dikembangkan oleh Taylor Otwell dan dirilis pada tahun 2011, Laravel dirancang untuk menyederhanakan proses pembuatan aplikasi web dengan menyediakan berbagai fitur serta alat yang kuat[17]. *Framework* ini bertujuan untuk menyederhanakan berbagai tugas umum dalam pengembangan web seperti routing, caching, dan autentikasi, sehingga pengembang dapat lebih fokus pada logika bisnis inti. Salah satu kelebihan utama Laravel adalah kemudahan dalam pengelolaan basis data. Dengan ORM (*Object-Relational Mapping*) yang disebut *Eloquent*, Laravel memungkinkan pengembang berinteraksi dengan basis data melalui

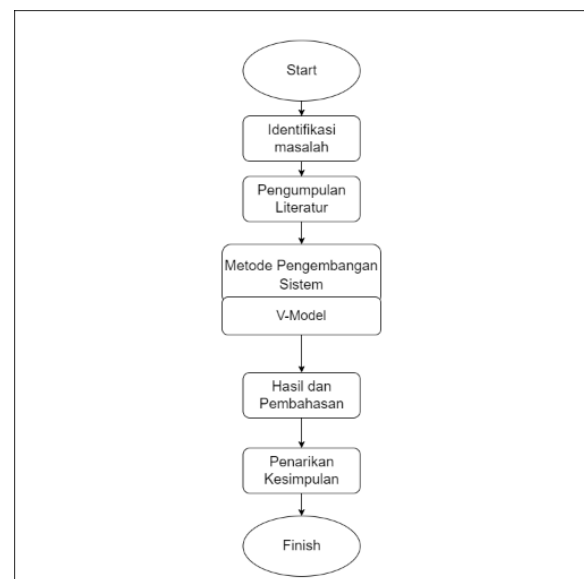
sintaks yang sederhana dan intuitif, tanpa perlu menulis query SQL secara langsung.

3. METODE PENELITIAN

Metode penelitian merupakan pendekatan yang disusun secara sistematis untuk data dikumpulkan, informasi dianalisis, dan tujuan yang telah ditetapkan dicapai dalam suatu kegiatan penelitian.

3.1. Tahapan Penelitian

Terdapat sejumlah tahapan yang dilalui dalam proses penelitian, dimulai dari pengumpulan data, perancangan sistem, hingga proses pengambilan keputusan. Rangkaian tahapan tersebut dapat dilihat pada Gambar 2.



Gambar 2. Tahapan Penelitian

Metode pengembangan sistem yang diterapkan dalam perancangan sistem ini adalah V-Model. Pemilihan metode ini didasarkan pada kemampuannya untuk menggambarkan urutan dasar pengembangan dan aktivitas pengujian. V-Model menekankan pentingnya berbagai tingkat pengujian yang diintegrasikan langsung dengan fase pengembangan.

3.2. Requirement Modeling

Tahap Requirements Modeling merupakan tahap untuk mendefinisikan kebutuhan, baik untuk kebutuhan pengguna maupun kebutuhan sistem. Pada tahap ini, dilakukan analisis mendalam untuk mengidentifikasi dan memahami semua kebutuhan yang diinginkan pengguna serta kebutuhan teknis sistem yang akan dikembangkan.

3.3. Architectural Design

Pada tahap ini, desain arsitektur basis data dilakukan dengan membuat perancangan Document Store Model. Model ini digunakan untuk menggambarkan bagaimana data akan disimpan dalam bentuk dokumen. Entitas-entitas dalam database diidentifikasi dan dikelompokkan berdasarkan tipe

data yang relevan. Setiap dokumen akan berisi data yang terstruktur sesuai dengan kebutuhan aplikasi.

3.4. Component Design

Tahapan ketiga merupakan perancangan sistem yang akan didetailkan menjadi modul-modul kecil dengan fungsi masing-masing. Modul merupakan atribut tunggal dari perangkat lunak yang berisi instruksi-instruksi untuk melakukan fungsi tertentu dan struktur yang jelas. Hal ini dilakukan untuk pemfokusan terhadap sistem yang dibuat terhadap setiap user yang akan menggunakan sistem. Hasil dari tahapan ini adalah *Usecase Diagram*, *Class Diagram*.

3.5. Code Generation

Tahapan selanjutnya, desain yang telah disusun akan diimplementasikan ke dalam sistem sesuai dengan rancangan yang telah dibuat. Database yang digunakan adalah No-SQL dengan Firestore sebagai alat untuk mengelola data. Sistem ini dikembangkan menggunakan framework laravel untuk memudahkan pengembangan dan memastikan performa yang optimal. Integrasi dengan Firestore juga memungkinkan pengelolaan data secara real-time, mendukung kebutuhan sistem yang dinamis. Penggunaan NoSQL memberikan keuntungan dan menangani data yang tidak terstruktur dan skalabilitas yang tinggi, sesuai dengan kebutuhan sistem.

3.6. Executable Software

Tahap berikutnya adalah *executable software*, di mana website yang telah dikembangkan akan dideploy ke server sementara untuk pengujian. Pada tahap ini, spesifikasi server yang akan digunakan menjelang fase validasi (testing) akan dijelaskan.

3.7. Unit Testing

Tahapan selanjutnya merupakan tahapan pengujian terhadap pengkodean yang telah dilakukan. Hal ini diterapkan untuk mengatasi permasalahan ketika pengkodean, yaitu *error* dari kode program. Jika terdapat *error* maka akan dilakukan perbaikan.

3.8. Integration Testing

Tahapan ini melibatkan modul-modul yang telah dibuat dan kesesuaiannya terhadap *Graphical User Interface* (GUI) yang telah dirancang. Pada tahap ini, setiap modul diuji bersama untuk memastikan bahwa alur kerja antar modul berjalan sesuai dengan desain yang diinginkan.

3.9. System Testing

Tahapan ini dilakukan untuk memastikan bahwa website yang dibuat telah memenuhi kebutuhan yang ditetapkan. Pengujian sistem dilaksanakan dengan menggunakan salah satu teknik dalam metode black box testing, yaitu validation testing. Pada pengujian ini, masukan dan keluaran dari setiap kebutuhan

fungsional sistem yang telah didefinisikan sebelumnya menjadi fokus utama.

3.10. Acceptance Testing

Pada *acceptance testing* akan dilakukan dengan cara mengimplementasikan langsung program yang sudah jadi terhadap user yang akan menggunakan program. Hal ini ditujukan untuk validasi terhadap kebutuhan yang telah dilakukan pada tahap requirement modelling sehingga diperoleh hasil apakah program ini sudah sesuai dengan kebutuhan user.

4. HASIL DAN PEMBAHASAN

Hasil dan pembahasan dalam penelitian ini meliputi output dari proses pengembangan dan pengujian, serta analisis yang mencakup tahap perencanaan kebutuhan, desain, hingga penerapan sistem.

4.1. Requirement Modeling

Berdasarkan proses bisnis inventory management yang sedang berjalan, berikut adalah kebutuhan-kebutuhan yang perlu dikembangkan dalam sistem inventory management yang diusulkan.

Tabel 1. Requirement Modelling

<i>Requirement</i>	<i>Role</i>
Mengelola Supplier	Admin
Mengelola Bahan Baku	Admin
Mengelola Pesanan	Admin
Mengelola Retur Pesanan	Admin
Mengelola Barang Produksi	Admin

4.2. Architectural Design

Architectural Design dilakukan dengan merancang Document Store Model. Model ini digunakan untuk menggambarkan cara penyimpanan data dalam format dokumen. Berikut adalah Document Store Model dari sistem yang dirancang:

```
{
  productId : "String",
  namaBarang : "String",
  jumlahBarang : "String",
  deskripsiBarang : "String",
  imageId : "String",
  imageUrl : "String",
  timestamp : "timestamp",
}
```

Gambar 3. Document Store Model Product

Gambar 3. Merupakan Document Store Model untuk produk yang menggambarkan bagaimana data terkait produk disimpan dalam bentuk dokumen.

```
{
  materialId : "String",
  namaBarang : "String",
  jumlahBarang : "String",
  unit : "String",
  deskripsiBarang : "String",
  imageId : "String",
  imageUrl : "String",
  timestamp : "timestamp",
}
```

Gambar 4. Document Store Model Bahan Baku

Gambar 4. Merupakan Document Store Model untuk bahan baku yang menggambarkan bagaimana data terkait bahan baku disimpan dalam bentuk dokumen.

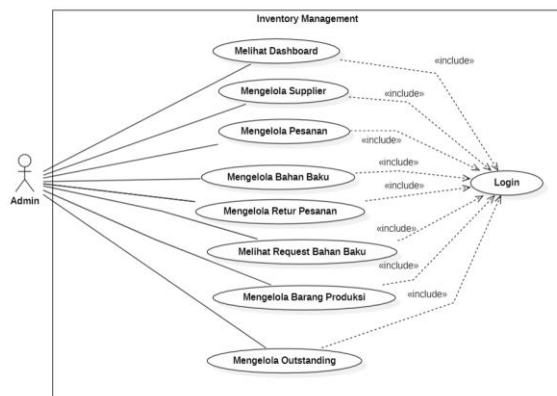
```
{
  orderId : "String",
  customerName : "String",
  deskripsi : "String",
  orderItemName : "Array",
  orderItemQuantity : "Array",
  status : "String",
  orderLeftOvers : "String",
  timestamp : "timestamp",
}
```

Gambar 5. Document Store Model Pesanan

Gambar 5. Merupakan Document Store Model untuk pesanan yang menggambarkan bagaimana data terkait pesanan disimpan dalam bentuk dokumen.

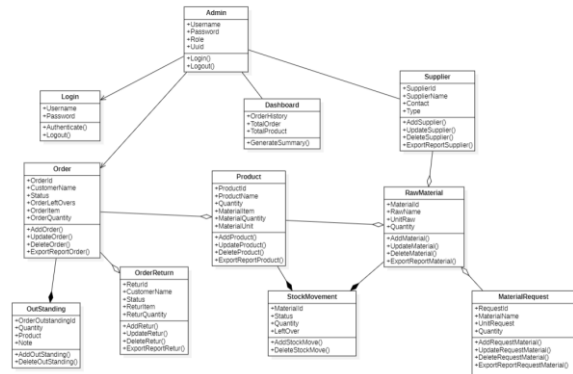
4.3. Component Design

Pada tahap Component Design, sistem besar yang telah dianalisis dan dirancang dipecah menjadi komponen-komponen kecil untuk memudahkan pengembangan. Setiap komponen dirancang agar memiliki fungsi spesifik, dapat berdiri sendiri, serta mampu berinteraksi dengan komponen lain sesuai kebutuhan sistem.



Gambar 6. Usecase Diagram

Berikut adalah Use Case Diagram yang dirancang berdasarkan hasil dari analisis kebutuhan sistem, dalam sistem ini dikhususkan untuk aktor admin.



Gambar 7. Class Diagram

Berikut Merupakan class diagram sistem inventory management yang digunakan untuk menggambarkan class yang ada di dalam sistem.

4.4. Code Generation

```
public function getCollection( $collectionName ) {
  $url = "https://firestore.googleapis.com/v1/projects/{this->projectId}/databases/{default}/documents/{collectionName}";

  $response = Http::withToken( $this->authToken )->get( $url );

  return $response->json();
}
```

Gambar 8. Source Code Get Collection

Berikut ditampilkan potongan kode yang digunakan untuk mengambil data dari Firestore. Data diambil dari koleksi yang telah disimpan di dalam database.

```
public function addDocument( $collectionName, array $data ) {
  $url = "https://firestore.googleapis.com/v1/projects/{this->projectId}/databases/{default}/documents/{collectionName}";

  $response = Http::withToken( $this->authToken )->post( $url, [
    'fields' => $this->formatFields( $data ),
  ] );

  return $response->json();
}
```

Gambar 9. Source Code Add Document

Gambar 9 menunjukkan source code untuk menambahkan dokumen. Kode ini digunakan untuk menyimpan data baru ke dalam koleksi di Firestore.

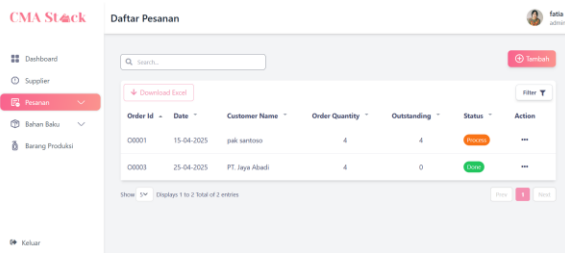
```
public function updateDocument( $collectionName, $documentId, array $data ) {
  $url = "https://firestore.googleapis.com/v1/projects/{this->projectId}/databases/{default}/documents/{collectionName}/{documentId}";

  $response = Http::withToken( $this->authToken )->patch( $url, [
    'fields' => $this->formatFields( $data ),
  ] );

  return $response->json();
}
```

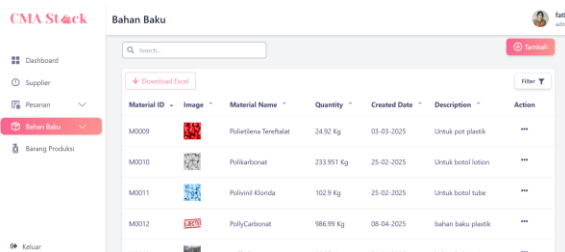
Gambar 10. Source Code Update Document

Gambar 10 menunjukkan source code untuk memperbarui dokumen. Kode ini digunakan untuk melakukan pembaruan data pada dokumen yang ada di Firestore.



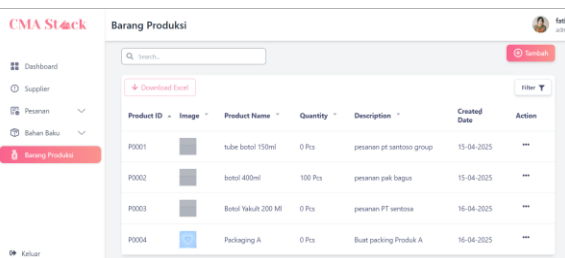
Gambar 11. Tampilan Pesanan

Gambar 11 menunjukkan tampilan website untuk pesanan. Tampilan ini digunakan untuk menampilkan rincian pesanan yang telah dibuat oleh pengguna di website.



Gambar 12. Tampilan Bahan Baku

Gambar 12 menunjukkan tampilan bahan baku. Tampilan ini digunakan untuk menampilkan daftar bahan baku yang tersedia di sistem.



Gambar 13. Tampilan Barang Product

Gambar 13 menunjukkan tampilan barang produk. Tampilan ini digunakan untuk menampilkan daftar produk yang tersedia dalam sistem.

4.5. Executable Software

Pada tahap ini, website yang telah diprogram kemudian di-deploy ke cPanel, yang merupakan panel kontrol berbasis web untuk mengelola hosting dan server. Setelah website di-deploy, pengaturan server dan database kemudian dikonfigurasi sesuai dengan

kebutuhan website, seperti perubahan file konfigurasi dan pengaturan koneksi database. Selain itu, pengaturan domain atau subdomain dilakukan agar website dapat diakses secara publik melalui URL yang telah ditentukan. Terakhir, untuk memastikan keamanan, pengaturan SSL diaktifkan agar komunikasi antara pengguna dan server dijamin keamanannya melalui protokol HTTPS.

4.6. Unit Testing

Pada bagian ini, akan dibahas mengenai Unit Testing, yaitu proses pengujian bagian-bagian terkecil dari perangkat lunak untuk memastikan bahwa setiap unit kode berfungsi sebagaimana mestinya. Unit testing membantu mendeteksi kesalahan lebih awal, meningkatkan keandalan perangkat lunak, dan memudahkan pemeliharaan kode. Berikut ini adalah tabel yang menggambarkan unit testing yang dilakukan :

Tabel 2. Unit Testing

Modul	Parameter	Result
Materials()	-	Pass
AddMaterial()	-	Pass
DeleteMaterial()	Material Id	Pass
MaterialEdit()	Material Id	Pass
Produksi()	-	Pass
AddProduksi()	-	Pass
EditProduksi()	Product Id	Pass
DeleteProduksi()	Product Id	Pass
Daftarpesanan()	-	Pass
Addpesanan()	-	Pass
EditPesanan()	Pesanan Id	Pass
DeletePesanan()	Pesanan Id	Pass

Hasil unit testing untuk beberapa modul dalam website ditunjukkan pada Tabel 2. Modul dan parameter yang diuji, beserta hasil pengujian yang diperoleh, dijelaskan pada setiap baris. Hasil Pass diperoleh pada semua pengujian, yang berarti bahwa setiap unit berfungsi sesuai dengan yang diharapkan tanpa adanya masalah.

4.7. Integration Testing

Integration Testing adalah tahap pengujian yang dilakukan setelah unit testing, di mana modul-modul yang telah diuji digabungkan untuk memastikan bahwa mereka berfungsi dengan baik bersama. Fokus utama adalah memeriksa apakah interaksi antar modul berjalan lancar dan data yang dipertukarkan diproses dengan tepat.

Tabel 3. Integration Testing

Tujuan	Hasil
Periksa alur akses dari Barang Produksi ke pesanan	Sesuai
Periksa alur akses dari Barang Produksi ke Bahan baku	Sesuai
Periksa alur akses dari Pesanan ke Bahan baku	Sesuai
Periksa alur akses dari Pesanan ke barang produksi	Sesuai
Periksa alur akses dari Bahan Baku ke barang produksi	Sesuai
Periksa alur akses dari Bahan Baku ke pesanan	Sesuai

Tabel 3 menunjukkan hasil Integration Testing yang dilakukan pada website, yang mencakup pengujian interaksi antar modul dan hasil pengujian yang diperoleh, yang semuanya sesuai dengan harapan.

4.8. System Testing

Pada tahap System Testing, seluruh sistem diuji untuk memastikan bahwa semua komponen bekerja

dengan baik secara keseluruhan. Pada tahapan ini pengujian dilakukan dengan menggunakan metode black box testing. Tabel 4 berikut menunjukkan hasil dari pengujian sistem yang dilakukan pada website, yang mencakup pengujian terhadap fungsionalitas dan kinerja sistem untuk memastikan bahwa website memenuhi spesifikasi.

Tabel 4. System Testing

Test Case	Expected	Result
Menambahkan Stok Product	Data Product Berhasil Ditambah	Pass
Mengedit Data Product	Data Product Berhasil Di Edit	Pass
Menghapus Product Dari Sistem	Data Product Terhapus Di Database	Pass
Menambah Stok Product Negatif	Menampilkan Alert Error Data Gagal Di Tambah	Pass
Menambah Stok Material	Stok Material Berhasil Ditambah	Pass
Mengedit Data Material	Stok Material Berhasil Di Edit	Pass
Menambah Stok Material Negatif	Menampilkan Alert Material Gagal Di Tambah	Pass
Delete Data Material	Data Material Berhasil Di Delete	Pass
Menambah Data Pesanan	Data Pesanan Berhasil Di Tambah	Pass
Edit Data Pesanan	Data Pesanan Berhasil Di Update	Pass
Delete Data Pesanan	Data Pesana Berhasil Terhapus	Pass

4.9. Acceptance Testing

Pada tahap Acceptance Testing, website diuji untuk memastikan bahwa sistem memenuhi kebutuhan dan harapan pengguna serta persyaratan yang telah disepakati. Tabel 5 berikut menunjukkan hasil dari

Acceptance Testing yang dilakukan pada website, mencakup pengujian fungsionalitas website sesuai dengan kriteria yang ditetapkan oleh pengguna. Hasilnya, semua fitur berfungsi sesuai dengan kebutuhan pengguna.

Tabel 5. Acceptance Testing

Fitur	Hasil	Sesuai	User
Menambah data Bahan baku	Data yang ditambahkan saat menambah bahan baku berhasil tersimpan	Ya	Admin
Mengedit data Bahan baku	Data yang diubah saat mengedit bahan baku berhasil diperbarui	Ya	Admin
Menghapus data Bahan baku	Data bahan baku yang baru ditambah berhasil dihapus	Ya	Admin
Menambah data barang produksi	Data yang ditambahkan saat menambah barang produksi berhasil tersimpan	Ya	Admin
Mengedit data barang produksi	Data yang diubah saat mengedit barang produksi berhasil diperbarui	Ya	Admin
Menghapus data barang produksi	Data barang produksi yang baru ditambah berhasil dihapus	Ya	Admin
Menambah data pesanan	Data yang ditambahkan saat menambah pesanan berhasil tersimpan	Ya	Admin
Mengedit data pesanan	Data yang diubah saat mengedit pesanan berhasil diperbarui	Ya	Admin
Menghapus data pesanan	Data pesanan yang baru ditambah berhasil dihapus	Ya	Admin

5. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian yang telah dilakukan melalui serangkaian tahapan, mulai dari analisis kebutuhan hingga tahap pengujian, dapat disimpulkan bahwa sistem informasi manajemen inventory berbasis web pada CV. Cahaya Mulya Abadi telah berhasil dirancang dan dibangun sesuai dengan tujuan penelitian. Sistem ini dikembangkan dengan menggunakan framework Laravel dan database Firestore, sehingga dihasilkan sebuah aplikasi berbasis web yang dapat digunakan oleh admin gudang untuk memantau stok barang, mengelola pergerakan inventory (barang masuk dan keluar), serta menghasilkan laporan secara real-time. Berdasarkan hasil pengujian yang mencakup unit

testing, integration testing, system testing, dan acceptance testing, seluruh tahapan pengujian telah dilalui dengan hasil valid (pass), sehingga sistem dinyatakan layak untuk digunakan dalam mendukung aktivitas pengelolaan inventory secara online melalui berbagai perangkat dan sistem operasi.

Saran untuk pengembangan ke depannya, diharapkan sistem informasi manajemen inventory ini dapat ditingkatkan dengan menambahkan akses langsung bagi supplier. Dengan demikian, pihak admin tidak perlu lagi menghubungi supplier secara manual untuk kebutuhan pengadaan barang, sehingga proses manajemen inventory dapat berjalan lebih cepat, efisien, dan akurat.

DAFTAR PUSTAKA

- [1] V. D. Anggriani And Firdaus Rayyan, "Penerapan Strategi Transformasi Digital Di Lingkungan Manufaktur Pada Sistem Informasi Manajemen," *Lokawati: Jurnal Penelitian Manajemen Dan Inovasi Riset*, Vol. 2, No. 4, Pp. 140–150, Jun. 2024, Doi: 10.61132/Lokawati.V2i4.1007.
- [2] E. Roso Pristiwaningsih, D. Rizky, T. A. Atmojo, And F. Nadhifah, "Transformasi Digital Di Industri Manufaktur: Dampak Pada Efisiensi," *Jurnal Sains Dan Teknologi Elektro*, Vol. 14, Pp. 2830–3512, 2024, Doi: 10.47709/Elektriese.V14i02.4809.
- [3] A. Andriani And J. Fernandes Andry, "Designing A Web-Based Inventory Application At General Steel Supplier Using Extreme Programming Method Perancangan Aplikasi Inventory Berbasis Web Pada General Steel Supplier Menggunakan Metode Extreme Programming," *Cogito Smart Journal* |, Vol. 9, No. 1, 2023.
- [4] L. Sitorus, J. L. Saragih, T. A. Eka, And M. Sihombing, "Sistem Informasi Inventaris Barang Berbasis Website Menggunakan Metode Garis Lurus (Studi Kasus : Rumah Sakit Jiwa Prof. Dr. M. Ildrem)," *Juki: Jurnal Komputer Dan Informatika*, Vol. 4, 2022.
- [5] Y. Yuningsih, "Penerapan Metode V-Model Dalam Perancangan Sistem Penjualan Online Produk Furniture Menggunakan Php Mysql Di Pd Dua Putri," *Jurnal Informasi Dan Komputer*, Vol. 10, No. 2, 2022.
- [6] D. Hartanti, A. Hiswara, B. Pratama, A. Noeman, And B. Sanjaya, "Rancang Bangun Sistem Informasi Kepegawaian Polres Metro Bekasi Menggunakan Metode V-Model," *Jurnal Jaring Saintek*, Vol. 5, No. 2, Pp. 99–106, 2023, [Online]. Available: <http://ejurnal.ubharajaya.ac.id/index.php/jaring-saintek99>
- [7] A. H. Mirza And Y. Maulana, "Rancang Bangun Aplikasi Buku Tamu Berbasis Website Menggunakan Laravel Di Pt.Kai Divre Iii Palembang," 2024.
- [8] D. Abdul Hakim, R. Bangun Aplikasi Penghitung, And T. Sumarni, "Rancang Bangun Aplikasi Penghitung Hpp Menggunakan Framework Laravel (Studi Kasus: Foodies Steak Grill And Coffee)," *Jifotech (Journal Of Information Technology)*, Vol. 05, No. 01, P. 272, 2025.
- [9] A. S. Karim Et Al., "Sistem Informasi Inventory Gudang (Si-Igun) Pada Cv. Mega Kencana," *Ijccs*, Vol. X, No.X, Pp. 1–5, 2023.
- [10] M. Ilhamsyah And M. Fathurrahman, "Rancang Bangun Aplikasi Android Mysmartfishfeeding Menggunakan Flutterflow Dan Firebase Firestore," *Jurnal Informatika Dan Teknik Elektro Terapan*, Vol. 12, No. 3s1, Oct. 2024, Doi: 10.23960/Jitet.V12i3s1.5269.
- [11] D. Nur Azizah, "Pengembangan Sistem Inventory Barang Perusahaan Dagang Berbasis Website (Studi Kasus : Cv. Agung Nugraha)," 2021.
- [12] H. Wijaya And N. Selamat Riyadi, "Analisa Dan Perancangan Sistem Informasi Inventory Barang Dengan Metode Extreme Programming Pada Pt Pesona Lestari Abadi," 2023.
- [13] M. S. S. Ibrahim, A. Faisol, And R. P. Prasetya, "Rancang Bangun Aplikasi Controlling Dan Monitoring Ruangan Jemuran Pintar Menggunakan Firebase Pada Arduino Dan Android," *Jati (Jurnal Mahasiswa Teknik Informatika)*, Vol. 7, No. 1, Pp. 979–983, Jun. 2023, Doi: 10.36040/Jati.V7i1.6180.
- [14] P. Selvi Armilia And S. Ani Arnomo, "Rancang Bangun Aplikasi E-Commerce Berbasis Web Pada Toko Zifa Beauty," *Jurnal Comasie*, Vol. 10, No. 03, 2024.
- [15] S. Rosetya Wardhana, I. Gede, D. Satya Herdaya, M. Ruswiansari, T. Surabaya, And N. Surabaya, "Pengembangan Aplikasi Classroom Berbasis Mobile Menggunakan Model V," 2024.
- [16] M. Ramadhan And A. Gustalika, "Rancang Bangun Aplikasi Pemesanan Tiket Tempat Wisata Berbasis Android Menggunakan Metode Extreme Programming," *Bulletin Of Information Technology (Bit)*, Vol. 5, No. 2, Pp. 114–124, 2024, Doi: 10.47065/Bit.V5i2.1341.
- [17] R. Yuniarti, I. H. Santi, And W. D. Puspitasari, "Perancangan Aplikasi Point Of Sale Untuk Manajemen Pemesanan Bahan Pangan Berbasis Framework Laravel," *Jurnal Mahasiswa Teknik Informatika*, Vol. 6, No. 1, 2022