

**Analisis dan Perancangan Sistem Informasi *Inventor*  
Menggunakan *Web Service RESTful*  
(Studi Kasus : Toko Kosy Alat dan Bahan Kue Salatiga)**

**Artikel Ilmiah**



**Peneliti:**

**Sri Sutejo (672015070)**

**Yeremia Alfa Susetyo, S.Kom., M.Cs.**

**Program Studi Teknik Informatika  
Fakultas Teknologi Informasi  
Universitas Kristen Satya Wacana  
Salatiga  
Juli 2019**

**Analisis dan Perancangan Sistem Informasi *Inventor*  
Menggunakan *Web Service RESTful*  
(Studi Kasus : Toko Kosy Alat dan Bahan Kue Salatiga)**

**Artikel Ilmiah**

**Diajukan kepada  
Fakultas Teknologi Informasi  
untuk memperoleh Gelar Sarjana Komputer**



**Peneliti:  
Sri Sutejo (672015070)  
Yeremia Alfa Susetyo, S.Kom., M.Cs.**

**Program Studi Teknik Informatika  
Fakultas Teknologi Informasi  
Universitas Kristen Satya Wacana  
Salatiga  
Juli 2019**



### PERNYATAAN TIDAK PLAGIAT

Saya yang bertanda tangan di bawah ini:

Nama : SRI SUTEJO  
NIM : 672015070 Email : 672015070@student.uksw.edu  
Fakultas : FTI Program Studi : TI  
Judul tugas akhir : Analisis dan Perancangan Sistem Informasi Inventory Menggunakan Web Service Rest full (Studi Kasus : Toko Kasy Alat dan Bahan ke Salatiga)  
Pembimbing : 1. Jeremia Alfa Susetyo, S.Kom., M.Cs.  
2. \_\_\_\_\_

Dengan ini menyatakan bahwa:

1. Hasil karya yang saya serahkan ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar keserjanaan baik di Universitas Kristen Satya Wacana maupun di institusi pendidikan lainnya.
2. Hasil karya saya ini bukan saduran/terjemahan melainkan merupakan gagasan, rumusan, dan hasil pelaksanaan penelitian/implementasi saya sendiri, tanpa bantuan pihak lain, kecuali arahan pembimbing akademik dan narasumber penelitian.
3. Hasil karya saya ini merupakan hasil revisi terakhir setelah diujikan yang telah diketahui dan disetujui oleh pembimbing.
4. Dalam karya saya ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali yang digunakan sebagai acuan dalam naskah dengan menyebutkan nama pengarang dan dicantumkan dalam daftar pustaka.

Pernyataan ini saya buat dengan sesungguhnya. Apabila di kemudian hari terbukti ada penyimpangan dan ketidakbenaran dalam pernyataan ini maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya saya ini, serta sanksi lain yang sesuai dengan ketentuan yang berlaku di Universitas Kristen Satya Wacana.

Salatiga, 7 Agustus 2019



Sri Sutejo



### PERNYATAAN PERSETUJUAN AKSES

Saya yang bertanda tangan di bawah ini:

Nama : SRI SUTEJO  
NIM : 672015070 Email : 672015070@student.uksw.edu  
Fakultas : FTI Program Studi : TI  
Judul tugas akhir : Analisis dan Perancangan Sistem Informasi Inventory  
Menggunakan Web Service Rest full (Studi kasus :  
Toko Kosy Alat dan Bahan ke Salatiga

Dengan ini saya menyerahkan hak *non-eksklusif*\* kepada Perpustakaan Universitas – Universitas Kristen Satya Wacana untuk menyimpan, mengatur akses serta melakukan pengelolaan terhadap karya saya ini dengan mengacu pada ketentuan akses tugas akhir elektronik sebagai berikut (beri tanda pada kotak yang sesuai):

- a. Saya mengizinkan karya tersebut diunggah ke dalam aplikasi Repositori Perpustakaan Universitas, dan/atau portal GARUDA
- b. Saya tidak mengizinkan karya tersebut diunggah ke dalam aplikasi Repositori Perpustakaan Universitas, dan/atau portal GARUDA\*\*

\* Hak yang tidak terbatas hanya bagi satu pihak saja. Pengajar, peneliti, dan mahasiswa yang menyerahkan hak *non-eksklusif* kepada Repositori Perpustakaan Universitas saat mengumpulkan hasil karya mereka masih memiliki hak copyright atas karya tersebut.  
\*\* Hanya akan menampilkan halaman judul dan abstrak. Pilihan ini harus dilampiri dengan penjelasan/ alasan tertulis dari pembimbing TA dan diketahui oleh pimpinan fakultas (dekan/kaprodi).

Demikian pernyataan ini saya buat dengan sebenarnya.

Salatiga, 7 Agustus 2019

1956

Yerima Alfa Susetyo, M.G.  
Tanda tangan & nama terang pembimbing I

Mengetahui,

  
Tanda tangan & nama terang mahasiswa

Tanda tangan & nama terang pembimbing II

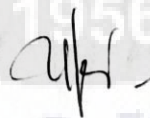
**Lembar Persetujuan**

**Analisis dan Perancangan Sistem Informasi Inventory Menggunakan Web Service Rest Full (Studi Kasus : Toko Kosy Alat dan Bahan Kue Salatiga)**

**Artikel Ilmiah**

**Peneliti :  
Sri Sutejo (672015070)**

**Telah disetujui untuk diuji:  
Tanggal : 08 Juli 2019.....**

1956  


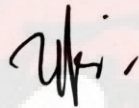
**Yerima Alfa Susetvo, S.Kom., M.Cs.  
Pembimbing**



## Lembar Pengesahan

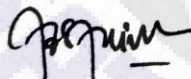
Judul Tugas Akhir : Analisis dan Perancangan Sistem Informasi Inventory  
Menggunakan Web Service Rest Full (Studi Kasus : Toko  
Kosy Alat dan Bahan Kue Salatiga)  
Nama Mahasiswa : SRI SUTEJO  
NIM : 672015070  
Program Studi : Teknik Informatika  
Fakultas : Teknologi Informasi

Menyetujui,

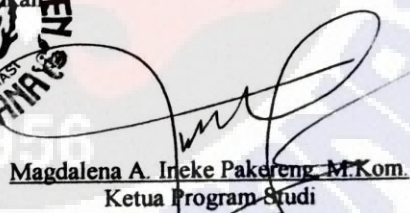


Yerima Alfa Susetyo, S.Kom., M.Cs.

Pembimbing 1



Dr. Wiwin Sulisty, S.T., M.Kom.  
Dekan

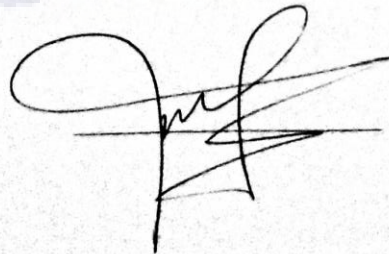


Magdalena A. Ineke Pakere, M.Kom.  
Ketua Program Studi

Dinyatakan Lulus Tanggal: 30 Juli 2019

Reviewer :

- Magdalena A. Ineke Pakere, M. Kom.



**Analisis dan Perancangan Sistem Informasi Inventory Menggunakan Web Service Rest Full (Studi Kasus : Toko Kosy Alat dan Bahan Kue Salatiga)**

Oleh,

**SRI SUTEJO**  
672015070

**ARTIKEL ILMIAH**

Diajukan Kepada Program Studi Teknik Informatika Guna Memenuhi Sebagian Dari  
Persyaratan Untuk Mencapai Gelar Sarjana Komputer

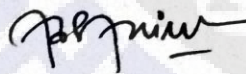
Disetujui oleh,



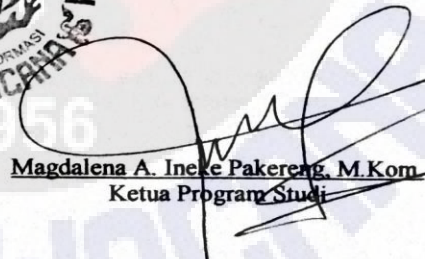
Yerima Alfa Susetyo, S.Kom., M.Cs.  
Pembimbing 1



Disetujui oleh,



Dr. Wiwin Sulisty, S.T., M.Kom.  
Dekan



Magdalena A. Ineke Pakereng, M.Kom.  
Ketua Program Studi

**FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS KRISTEN SATYA WACANA  
SALATIGA  
2019**

**Analisis dan Perancangan Sistem Informasi Inventory Menggunakan Web Service Rest Full (Studi Kasus : Toko Kosy Alat dan Bahan Kue Salatiga)**

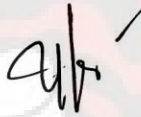
Oleh,

**SRI SUTEJO**  
672015070

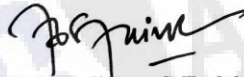
**LAPORAN PENELITIAN**

Diajukan Kepada Program Studi Teknik Informatika Guna Memenuhi Sebagian Dari  
Persyaratan Untuk Mencapai Gelar Sarjana Komputer

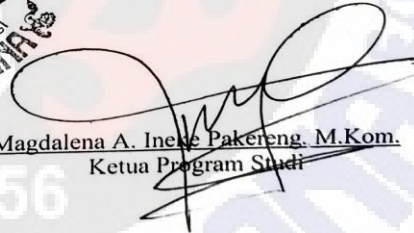
Disetujui oleh,



Yerima Alfa Susetyo, S.Kom., M.Cs.  
Pembimbing 1



Dr. Wiwin Sulistyono, S.T., M.Kom.  
Dekan



Magdalena A. Ineke Pakereng, M.Kom.  
Ketua Program Studi

**FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS KRISTEN SATYA WACANA  
SALATIGA  
2019**



## 1. Pendahuluan

Seiring dengan meningkatnya kebutuhan terhadap pengelolaan item-item yang menjadi persediaan dalam proses produksi, menuntut ada perubahan dalam hal peningkatan sistem administrasi pada bagian *logistic*. Kondisi ini secara langsung menyebabkan perlu dilakukannya sebuah *Continious Improvement* dalam hal peningkatan pengolahan administrasi item-item yang ada pada bagian logistik tersebut. Banyak cara untuk mengontrol informasi internalnya. Informasi tersebut akan tersampaikan dari level kepemimpinan paling bawah ke atas atau sebaliknya secara rapi. Sistem merupakan suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu, sistem dapat juga diartikan sebagai kumpulan dari elemen-elemen yang saling berinteraksi mencapai suatu tujuan tertentu [1].

Toko Kosy merupakan sebuah toko yang melayani penjualan alat dan bahan untuk pembuatan kue. Proses pelayanan dengan cara pelanggan mengambil barang sendiri kemudian membayarkan di kasir. Toko peralatan roti ini merupakan salah satu toko yang melayani alat dan bahan yang cukup ramai sehingga sering melakukan penambahan barang dari gudang ke toko untuk mencukupi permintaan barang dari para pelanggan. Proses pemenuhan barang ini kadang mengalami kendala dalam melihat apakah barang di gudang masih tersedia atau tidak untuk mencukupi permintaan barang di etalase. Apabila proses pencarian dan pemenuhan barang terhambat maka akan berpengaruh kepada pelayanan konsumen yang kurang baik. Mengingat permintaan pelanggan yang banyak dan berbeda barang.

Untuk mengatasi permasalahan tersebut diperlukan aplikasi yang otomatis dan *open platform*. Pada masa sekarang, salah satu teknologi yang banyak digunakan berupa aplikasi *web*. Telah tersedia *metode* untuk mengembangkan web seperti *framework* dan *web service* dengan berbagai bahasa seperti *java*, *PHP*, dan yang lainnya. *Web service* adalah sebuah komputasi yang dapat diakses melalui jaringan intranet maupun standar *protocol* tertentu dalam *platform* dan antarmuka bahasa pemrograman yang independen. *Transfer representasional state* (REST) merupakan abstraksi dari elemen arsitektur dalam sistem *hypermedia* terdistribusi. *Rest* memiliki banyak keuntungan dimana yang pertama adalah segi *performance* yang berasal dari kesederhanaan yang terkandung di dalam *Rest*. *Rest* didasarkan pada standar yang digunakan dalam *web* dan tidak memerlukan standar tambahan, yang menghindari ketergantungan suatu *platform* signifikan khusus dan penurunan okupansi sumber daya sistem [2].

Berdasarkan hal tersebut maka dilakukan penelitian yang bertujuan untuk menganalisis rancangan sistem informasi *inventory* menggunakan *Web Service RESTFull*. Perancangan ini bertujuan untuk memberikan kemudahan pendataan dan informasi bagi pemilik toko Kosy.

## 2. Tinjauan Pustaka

Penelitian dengan judul Sistem Informasi Inventori Barang Menggunakan Metode Object Oriented di PT. Livaza Teknologi Indonesia Jakarta, memiliki tujuan untuk menangani masalah yang terjadi di PT. Livaza Teknologi Indonesia tentang mengelola *inventory* barang. Semua kegiatan tersebut masih dilakukan secara manual dan dalam penerimaan barang masuk dan barang keluar juga masih belum terkontrol dengan baik. Masalah yang timbul adalah adanya ketidaksesuaian dengan catatan stoknya. Sehingga dengan adanya penelitian ini diharapkan dapat memberikan solusi kepada PT. Livaza Teknologi Indonesia dengan membuat suatu aplikasi berbasis *web* untuk mengelola data *inventory* barang tersebut [3].

Terdapat penelitian terdahulu dengan judul Sistem Informasi Inventori Gudang Untuk Mengontrol Persediaan Barang Pada Gudang Studi Kasus : PT. Alaisys Sidoarjo. PT. Alaisys Sidoarjo dalam pengawasan dan pencatatan terhadap barang persediaan memiliki kendala. Pelaporan dari gudang ke kantor pusat dilakukan dengan cara menyalin data dari kartu ke dalam Microsoft Office Excel. Laporan dalam format excel tersebut harus dikirim melalui via email atau gadget. Sehingga pihak kantor kurang efisien dalam melihat data barang di gudang. Adanya aplikasi sistem *inventory* gudang berbasis *web* diharapkan dapat menginventarisasi dengan produk yang ada pada stok-stok di gudang yang meliputi pencatatan, pengelolaan, dan pelaporan data pada persediaan gudang [4].

Hasil Penelitian yang berjudul Perancangan Sistem Informasi *Inventory* di AMIK Lembah Dempo Pagaralam menjelaskan bahwa pada dasarnya suatu organisasi memiliki barang *inventory* karena setiap organisasi pasti mempunyai barang-barang inventaris yang harus dikelola. Pengelolaan *inventory* yang dilakukan masih sangat manual dengan melakukan pembukuan terlebih dahulu terhadap barang inventaris dan *inventory* lainnya yang kemudian akan dipindahkan ke dalam komputer untuk dicatat. Adanya perancangan sistem informasi *inventory* di AMIK lembah Dempo Pagaralam diharapkan nantinya dapat membantu dan mempermudah pengelolaan *inventory* di AMIK lembah Dempo [5].

Penelitian berjudul Analisis dan Perancangan Sistem Informasi *Inventory Sparepart* Kapal Berbasis *Web* (Studi kasus Asia Group Pangkalpinang), menunjukkan bahwa saat ini sistem yang berjalan masih terdapat kekurangan seperti pencarian data, pencatatan data, dan pembuatan laporan yang cukup lama. Hal ini dapat memperlambat karyawan dalam pendataan *inventory sparepart*. Sistem *inventory* ini berbasis *web* sehingga dapat digunakan dengan mudah, *user friendly* dan mengelola data *sparepart* yang bisa dilakukan secara fleksibel dengan berbagai menu yang tersedia untuk *user* cabang dan *Manager Purchasing* [6].

Pada penelitian dengan judul Pembangunan *Web Service* Data Masyarakat Menggunakan *REST API* dengan *Access Token*, menunjukkan bahwa data disajikan merupakan hal yang penting di Indonesia dan dalam era teknologi informasi menunjukkan bahwa saat ini. Pemerintahan masih belum menggunakan data publik secara maksimal untuk tujuan administrasi. Pemanfaatan data populasi besar ini adalah

pembuatan sistem aplikasi layanan *web* dengan *REST API*, tempat data ini akan terbuka dan dapat diakses oleh *user* yang memiliki akses. Aplikasi ini mampu menyediakan dan memfasilitasi banyak pihak, seperti administrator data untuk memantau penggunaan data, registrasi karyawan dalam *input* data, dan orang-orang dapat mendaftar secara mandiri [7].

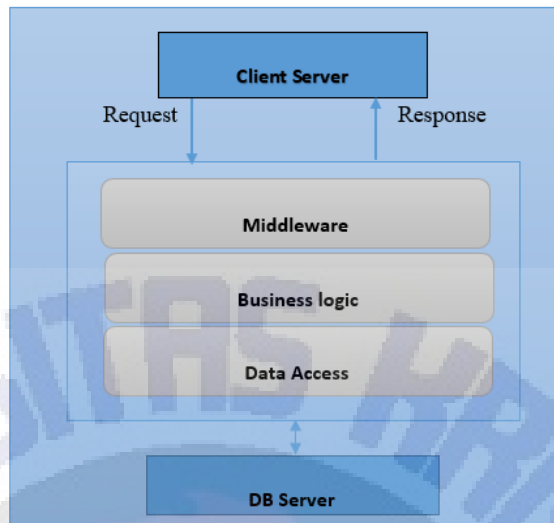
Berdasarkan penelitian-penelitian terdahulu yang pernah dilakukan tentang sistem informasi *inventory*, maka dilakukan penelitian yang membahas tentang perancangan sistem informasi *inventory* Menggunakan *Web Service RESTful*, dengan tujuan untuk memberikan kemudahan pendataan dan pemberi informasi bagi pemilik toko Kosy dalam pengolahan data *inventory*.

*Web Service* memiliki konsep *resource* sebagai komponen dari aplikasi yang perlu digunakan atau dialamatkan. *Web Service RESTful* membangun integritas dengan cara yang lebih ringan dan sederhana, dan berfokus pada sumber daya. *Rest* dapat dijelaskan dalam lima batasan, yaitu:

1. *Resource Identification* : *Web* bergantung pada *Uniform Resource Identifier (URL)* untuk mengidentifikasi sumber daya, sehingga *link* ke sumber daya dapat dibentuk menggunakan skema identifikasi yang mudah untuk dikenali
2. *Connectedness*: artinya klien dari *RestFull Service* seharusnya mengikuti *link* untuk menemukan sumber daya agar dapat berinteraksi dengan *Service*.
3. *Uniform Interface*: artinya sumber daya harus tersedia melalui antarmuka yang seragam dengan semantik yang mendefinisikan interaksi, seperti *Hypertext Transfer Protocol (HTTP)*. *HTTP POST, GET, PUT dan DELETE*.
4. *Self-Describing Messages*: artinya mengekspos *RestFull* menggunakan lebih dari satu format data (XML, JSON, RDF, dan lain-lain) dibandingkan dengan *SOAP (XML)*, namun hal ini tergantung *developer*.
5. *Stateless Interactions*: mengharuskan setiap *request* dari klien lengkap, dalam arti bahwa semua informasi *request* ke *server* harus berisi setiap informasi yang dibutuhkan agar *request* dapat dipahami, dan tidak ada ketergantungan dengan atau penanda dari *client*.

*Rest* menentukan sekumpulan prinsip arsitektur yang dapat digunakan untuk merancang *Web Service* yang berfokus pada sumber daya sistem, termasuk proses transfer data melalui *HTTP* oleh beberapa klien yang ditulis dengan bahasa pemrograman yang berbeda. *REST* dapat mengoperasikan fungsi *CRUD (create, read, update dan delete)* yang dapat dilakukan dengan memanfaatkan metode *HTTP* antara lain *POST, GET, PUT dan DELETE*. Format *application/x-www-form-urlencoded* yang digunakan oleh masing-masing metode *HTTP* di antaranya *GET dan DELETE* berbeda dengan *POST dan PUT* adalah berbeda. *Parsing* data pada Metode *GET dan DELETE* dimulai melalui *URL*. Sedangkan *POST dan PUT* melakukan *parsing* data melalui *payload HTTP* dengan memanfaatkan media type *reapplication/x-www-form-urlencoded* [8].

Konsep arsitektur yang digunakan pada *restful web service* yang digunakan pada sistem *inventory* gudang pada toko kosy ditunjukkan pada Gambar 1.



**Gambar 1** Arsitektur *Restful* [9]

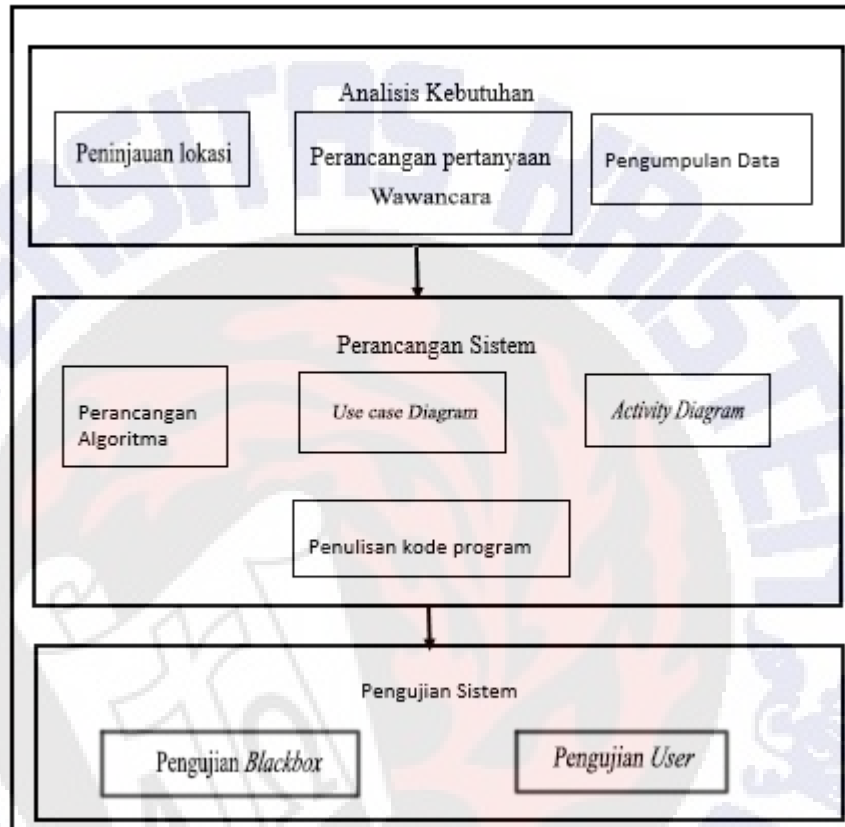
Gambar 1 menunjukkan arsitektur *RESTful* dimana terdapat beberapa lapisan *Middleware*, *Business Logic* dan *Data Acces*. Pada lapisan *Midddware* memiliki fungsi proses komunikasi (*POST*, *GET*, *PUT* dan *DELETE*). Lapisan *Midddware* akan menerjemahkan setiap *request HTTP Body* dan validasi *token* dari *client*. Sedangkan *Business layer* berfungsi mengimplementasikan fungsionalitas inti dari sistem, dan merangkum logika bisnis yang relevan. Lapisan *Data access* berperan untuk mengekspos data berdasarkan batasan-batasan yang dimiliki sistem. Data yang diekspos menggunakan format data *JSON*. *JSON* adalah bahasa independen dan berdasarkan pasangan *key/value* dan mempunyai *list value*. Struktur ini memungkinkan untuk digunakan dalam setiap bahasa pemrograman modern yang membuatnya menjadi pilihan yang baik untuk berkomunikasi di dunia *web* [9]. *JSON* adalah sebuah *token* berbentuk string yang sangat padat ukurannya, informasi mandiri yang memiliki kegunaan autentikasi dan pertukaran informasi. Karena ukurannya kecil, *JSON* dapat dikirim melalui *HTTP POST* atau di dalam *Header HTTP*, sehingga dapat ditransmisikan secara cepat [10].

Web service merupakan sebuah perangkat lunak yang tidak terpengaruh oleh platform, arsitektur, maupun bahasa pemrograman, yang menyediakan layanan atau method untuk pertukaran data yang dapat diakses oleh *network*. Contoh implementasi dari WS antara lain adalah SOAP dan REST [10].



### 3. Metode Penelitian

Tahapan dalam melakukan penelitian yang dilakukan terbagi ke dalam tiga tahap, yaitu : (1) Analisis Kebutuhan, (2) Perancangan Sistem, dan (3) Pengujian sistem.

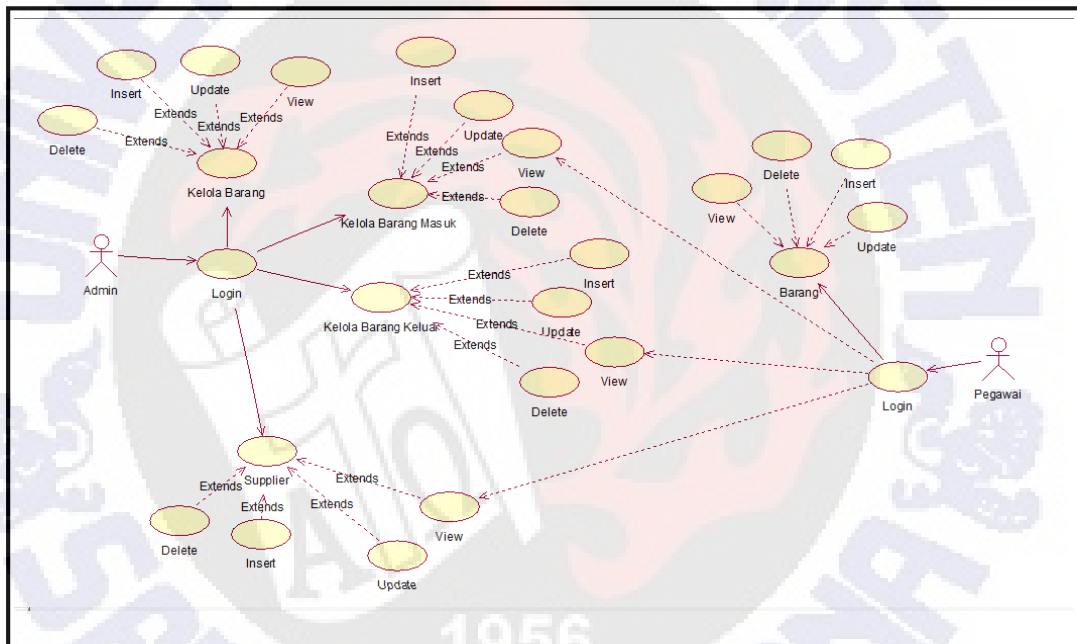


Gambar 2 Tahapan Penelitian

Tahapan penelitian pada Gambar 2, dijelaskan sebagai berikut, tahapan pertama adalah analisis kebutuhan, meliputi peninjauan lokasi untuk melakukan penelitian. Tahapan wawancara dilakukan untuk mendapatkan data dari pemilik toko untuk mengetahui kebutuhan dan yang diharapkan dalam tahap pembuatan aplikasi. Proses wawancara dengan cara tatap muka langsung. Pengumpulan data digunakan untuk menjadi dasar pembuatan aplikasi yang akan dibangun. Tahap kedua adalah perancangan sistem, perancangan ini pertama adalah membuat algoritma program yang akan digunakan dalam pembuatan program. Kemudian *use case* untuk menggambarkan *actor*, *use case*, dan interaksi untuk memperkenalkan suatu sistem. *Use case* memberikan gambaran singkat hubungan singkat *use case*, *actor* dan sistem. *Actor* dua yaitu *admin* dan pegawai. *Activity* diagram yang digunakan untuk menggambarkan

*workflow* (aliran kerja) atau aktivitas dari suatu proses bisnis. Pada tahap ini menggambarkan bagaimana sistem akan berjalan, hal yang menjadi komponen antara lain *activity* atau *state*, *decision*, dan lain-lain. Penulisan kode program, pengkodean sistem merupakan proses penerjemahan bahasa manusia ataupun desain ke dalam bahasa komputer. Pengkodean menggunakan *frame work Java Spring restfull* dan *MySQL* sebagai database untuk mengembangkan program yang nantinya mendukung proses yang dilakukan oleh *user*. Tahap ketiga adalah pengujian program, sebuah pengembangan adalah melakukan sebuah pengujian sistem dengan beberapa langkah uji. Pengujian pertama *black box* dengan melakukan *input* data secara berulang-ulang sampai ditemukan kesalahan sistem

Perancangan sistem ini menggunakan *Unfined Modelling Languange (UML)*. UML bersifat *scalability*, yaitu objek mudah digunakan untuk menggambarkan sistem yang besar dan kompleks.

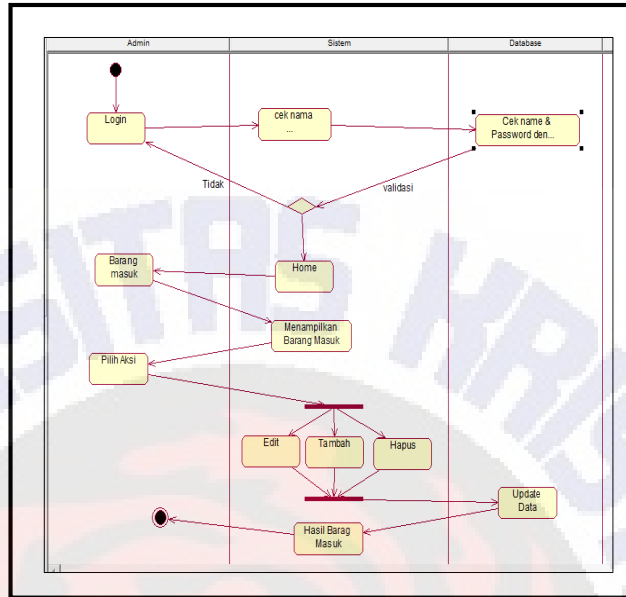


**Gambar 3** Diagram *Use case*

Gambar 3 menunjukkan *use case diagram* sistem *inventory* barang yang melibatkan *admin* dan pegawai. *Admin* disini merupakan orang yang mengolah data barang gudang dan memiliki akses penuh untuk kelola barang. Fungsi kelola barang ini untuk memiliki fungsi menambahkan barang. Kemudian *admin* memiliki fungsi kelola barang masuk dan keluar, serta kelola data *supplier*. Pada pegawai memiliki hak akses untuk melihat data barang masuk, keluar dan data *supplier*. Serta memiliki satu fungsi untuk menambahkan nama barang saja.

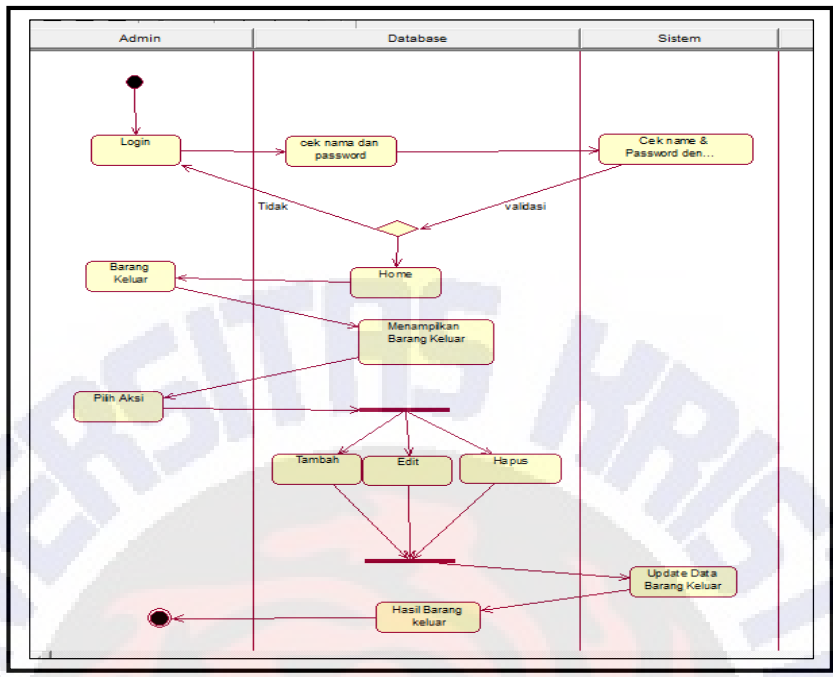
Gambar 4 merupakan diagram alur proses aplikasi pada menu kelola barang masuk, pada diagram tersebut menjelaskan bahwa *admin* terlebih dahulu harus *login* untuk mendapatkan akses halaman utama. Kemudian ada pada menu *home*, *admin*

memilih fungsi barang masuk, pada menu barang masuk 4 aksi yaitu *create*, *read*, *update*, dan *delete*. Setelah aksi dilakukan maka akan menampilkan data di halaman.

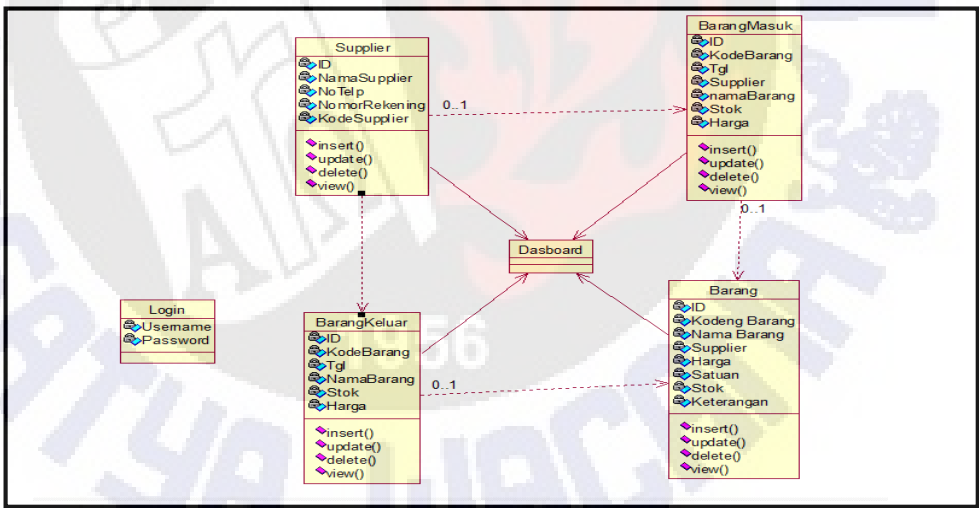


**Gambar 4** Diagram Activity Kelola Barang Masuk

Gambar 5 merupakan diagram alur proses aplikasi pada menu kelola barang keluar, pada diagram tersebut dijelaskan ketika *admin* mendapatkan hak akses pada halaman menu kelola barang keluar maka terdapat 4 aksi juga untuk *create*, *read*, *update* dan *delete*. Pada halaman barang keluar akan menampilkan data yang berada pada *database*.



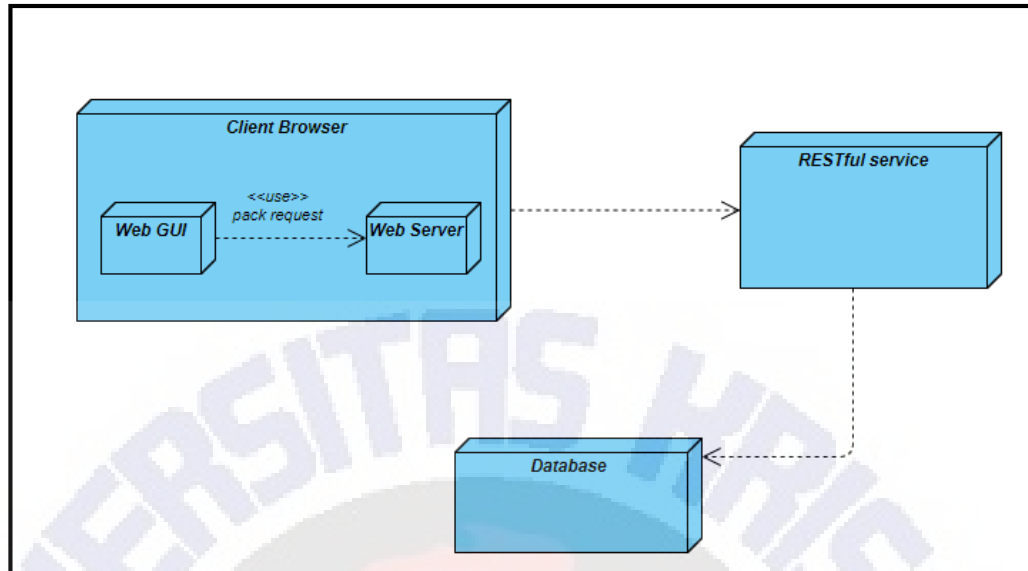
Gambar 5 Diagram Activity Kelola Barang Keluar



Gambar 6 Class Diagram

Gambar 6 merupakan *class diagram* yang akan digunakan dalam pembuatan aplikasi. Relasi dari setiap *class* akan digunakan untuk acuan sebagai pembuatan tabel pada database. *Boundary* berfungsi untuk menghubungkan beberapa tabel dan mengelola data pada database. Sedangkan *entity* berfungsi menggabungkan dan menampilkan fungsi-fungsi yang terdapat pada *boundary* dan *controller class*.





**Gambar 7** Deployment Diagram

Gambar 7 menunjukkan *deployment diagram* yang berawal dari *client*, *web service*, dan *database*. Dalam diagram tersebut *client* akan memberikan perintah yang diakses dengan *web GUI* dan perintah tersebut akan diteruskan ke *web server*. Perintah yang diberikan merupakan bagian dari *RESTful* yang kemudian digunakan untuk melakukan olah data pada *database*.

#### 4. Hasil dan Pembahasan

Penelitian yang dilakukan menghasilkan aplikasi Sistem Informasi *Inventory* menggunakan *Web Service RESTful* Toko Kosy Alat dan Bahan Kue Salatiga, yang dirancang dan dibuat dalam bentuk *web*. Sistem ini diharapkan dapat mempermudah proses pendataan barang yang dilakukan oleh Toko Kosy. Adanya sistem Informasi *Inventory* diharapkan dapat memudahkan pemilik toko mengetahui data barang yang dimiliki serta keluar masuk barang lebih cepat dan efisien melalui aplikasi *Inventory* ini. Hal ini dapat ditunjukkan dalam sistem mulai dari *login*, penginputan data hingga membuat laporan ke dalam bentuk excel setiap barang yang keluar maupun masuk

Inventory Gudang

Home | Barang | Barang Masuk | Barang Keluar | Supplier | Logout

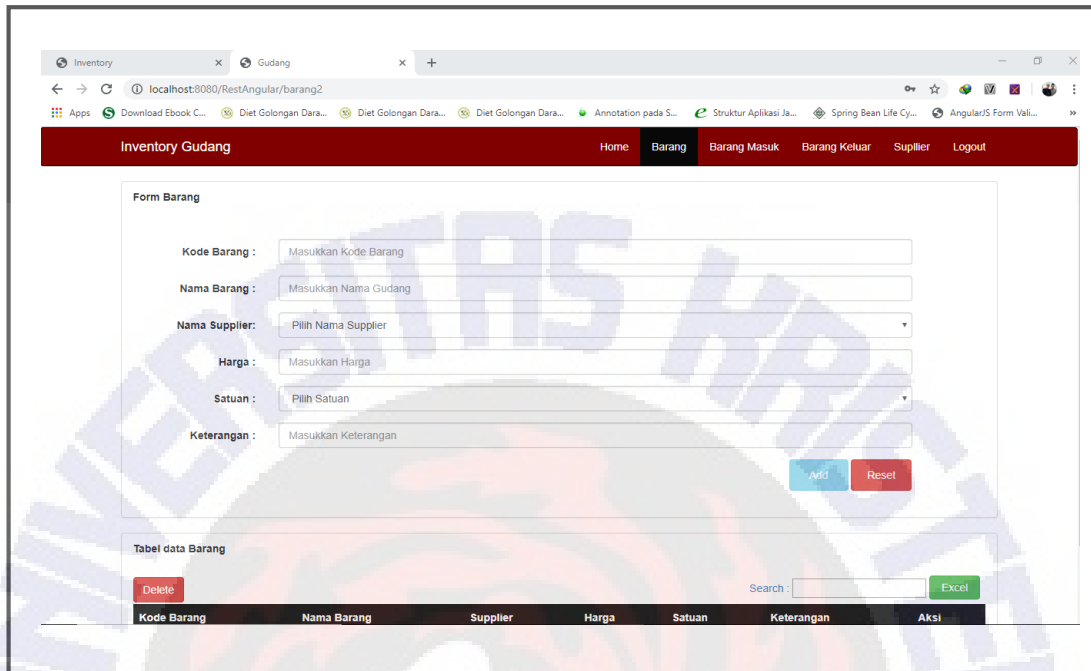
Tabel data Barang

Search:

Kode Barang	Nama Barang	Supplier	Harga	Satuan	Stok	Keterangan
713123	mie	mas too	123	pcs	12	admin
23123	dfsdsdf	adfsa	12312	12	12	admin
0011	terngu	rehwan	100000	1	22	admin
002	minyak	bimoli	2000	pcs	124	admin
63456345	fsdof	zxfzdf	123456	Pcs	0	admin
123	kjdfasik	telo	12000	Pcs	0	admin
123	minyak	telo	12000	Kardus	0	admin
23	kopi	supomo	20000	Pcs	0	admin
123124	plastik	123	24234	Pcs	0	admin
qwewqe	4qwrqwer	rehwan	24234	Pcs	0	admin
123	Mentega	supomo	7000	kilogram	0	admin
12345	minyak	supomo	10000	Pcs	0	admin

**Gambar 8** Halaman Utama *Dashboard* Pada *Admin*

Gambar 8 adalah halaman awal pada aplikasi yang akan menampilkan keseluruhan data. Untuk melakukan pengolahan barang, *admin* dapat memilih sub menu barang untuk melakukan penambahan item barang. Halaman utama ini menampilkan data stok barang keseluruhan yang berada pada *database*, dan bukan data barang masuk atau keluar. Data yang ditampilkan pada halaman utama ini secara otomatis akan berubah sesuai dengan data barang masuk dan keluar.



**Gambar 9** Halaman Barang

Gambar 9 merupakan tampilan halaman barang untuk mengolah data item barang. Halaman barang hanya diperuntukkan untuk menambah item barang berdasarkan value-nya, terdapat empat *text field* dan dua *drop down*. Pilihan pada *drop down supplier* diambil dari tabel *supplier*. Pada *form* ini semua data harus terisi. Apabila masih terdapat yang kosong maka data belum dapat disimpan. Kemudian pada halaman ini dapat melihat data yang disimpan. Tabel barang dapat dilihat pada Gambar 9. Gambar 10 menampilkan gambar tabel barang yang telah disimpan. Tabel barang ini berbeda dengan tabel barang yang berada pada halaman utama. Pada tabel barang ini diberikan informasi mengenai nama barang. Untuk menambahkan stok dan melakukan pengolahan barang keluar dapat memlih sub menu barang masuk dan barang keluar.

Kode Barang	Nama Barang	Supplier	Harga	Satuan	Keterangan	Aksi
713123	mie	mas too	123	pcs	admin	
23123	dfsdsdf	adsfa	12312	12	admin	
0011	terigu	rehwan	100000	1	admin	
002	minyak	bimoli	2000	pcs	admin	
63456345	fsdgr	zxfzdf	123456	Pcs	admin	
123	kjdfasik	telo	12000	Pcs	admin	
123	minyak	telo	12000	Kardus	admin	
23	kopi	supomo	20000	Pcs	admin	
123124	plastik	123	24234	Pcs	admin	
qweqwe	4qwrqwer	rehwan	24234	Pcs	admin	
123	Mentega	supomo	7000	kilogram	admin	
12345	minyak	survmm	10000	Pcs	admin	

Gambar 10 Tabel Barang

#### Kode Program 1 Controller Barang

```

1. @value = "GET LIST Barang DATA", response = Barang.class)
2. @RequestMapping(value = "/barang/export", method = RequestMethod.GET)
3. public ResponseEntity<List<Barang>> listAllBarangExport(Model model
4. HttpServletRequest request, HttpServletResponse response) {
5. System.out.println("masuk mapping barang");
6. List<Barang> master = barangService.getAllBarang();
7. if (master.isEmpty()) {
8. return new ResponseEntity<List<Barang>>(HttpStatus.NO_CONTENT);}
9. String dataMasuk = master.toString();

```

Kode Program 1 merupakan *sourcecode service* barang pada *controller*, pada aplikasi ini memanfaatkan *mapping* untuk menjalankan fungsi yang berbeda. Fungsi *sourcecode* pada Kode Program 1 adalah untuk menampilkan data barang dari *database* ke halaman *jsp*. Kemudian jika ingin menggunakan fungsi lain seperti *update* dan *delete* dapat menggunakan *mapping* yang sesuai. Perintah pada baris kedua berfungsi untuk menjadi fungsi *mapping* saat *request* dari *front end* ke *controller*. Perintah pada baris ketiga memiliki fungsi untuk menampung *entity* yang sudah dideklarasikan pada *class* barang. Perintah pada baris keenam merupakan *sourcecode* untuk mengambil data sesuai *entity* barang dari *database* dan dicek apakah datanya ada atau kosong.



## Kode Program 2 Controller Barang

```
1. @RequestMapping(value = "/barang/{id}", method = RequestMethod.PUT)
2. public ResponseEntity<Barang> updateBarang(@PathVariable("id") long id,
3. @RequestBody Barang mfc) {
4. System.out.println("Updating Data Barang with ID : " + id);
5. barangService.editBarang(mfc);
6. return new ResponseEntity<Barang>(HttpStatus.OK);
7. @RequestMapping(value = "/barang1/{kodebarang}", method
8. RequestMethod.PUT)public ResponseEntity<Barang>
9. DeleteBarang(@PathVariable("kodebarang") String KodeBarang) {
10. System.out.println("DELETE BARANG WITH KODE BARANG : "+KodeBarang);
11. barangService.deleteBarangByKodeBarang(KodeBarang); return new
12. ResponseEntity<Barang>(HttpStatus.OK); }
```

Kode Program 2 merupakan *sourcecode* untuk *edit* dan *delete*, menggunakan *mapping* yang berbeda untuk *service* yang berbeda. Sehingga dalam aplikasi tidak terdapat *mapping* yang sama untuk menghindari perintah yang ambigu. Pengecekan data barang dengan *database* menggunakan *id*. Baris pertama *mapping* untuk memanggil perintah *controller* dari *front end* dan menggunakan *RequestMethod.PUT* karena mengambil data dari *database*. Baris kedua merupakan perintah untuk *update* data barang. Baris ketujuh merupakan *mapping* untuk perintah *delete*, sama seperti *update* saat akan mengambil data untuk dihapus menggunakan *PUT*. Baris kesebelas untuk melakukan perintah *delete* Barang berdasarkan kode barang.

The screenshot shows a web browser window displaying the 'Inventory Gudang' application. The page has a dark red navigation bar with links for 'Home', 'Barang', 'Barang Masuk', 'Barang Keluar', 'Supplier', and 'Logout'. The main content area is titled 'Form Barang' and contains several input fields: 'Kode Barang' (a dropdown menu), 'Tgl' (a date picker), 'Nama Supplier' (a dropdown menu), 'Nama Barang' (a dropdown menu), 'Stok' (a text input), and 'Harga' (a text input). There are 'Add' and 'Reset' buttons at the bottom right of the form. Below the form is a section titled 'Tabel data Barang' which includes a 'Delete' button, a search bar, and an 'Excel' button. The table header shows columns for 'Tgl', 'Supplier', 'Nama Barang', 'Stok', 'Harga', and 'Aksi'.

Gambar 11 Halaman Barang Masuk

Gambar 11 menunjukkan halaman barang masuk, yang terdapat pada beberapa *text field* dan *drop down*. Pada halaman barang masuk, *user* harus memasukkan data berupa kode barang yang tinggal memilih sesuai data yang sudah dimasukkan di menu barang, tanggal barang masuk, nama *supplier* dan nama barang menggunakan *drop down* dengan data yang ada pada *database*. Kemudian pada halaman ini *user* bisa memasukkan stok barang yang masuk, dan yang terakhir adalah harga. Halaman barang masuk terdapat tabel yang menampilkan hasil pengolahan data yang telah dilakukan.

**Gambar 12** Halaman Barang Keluar

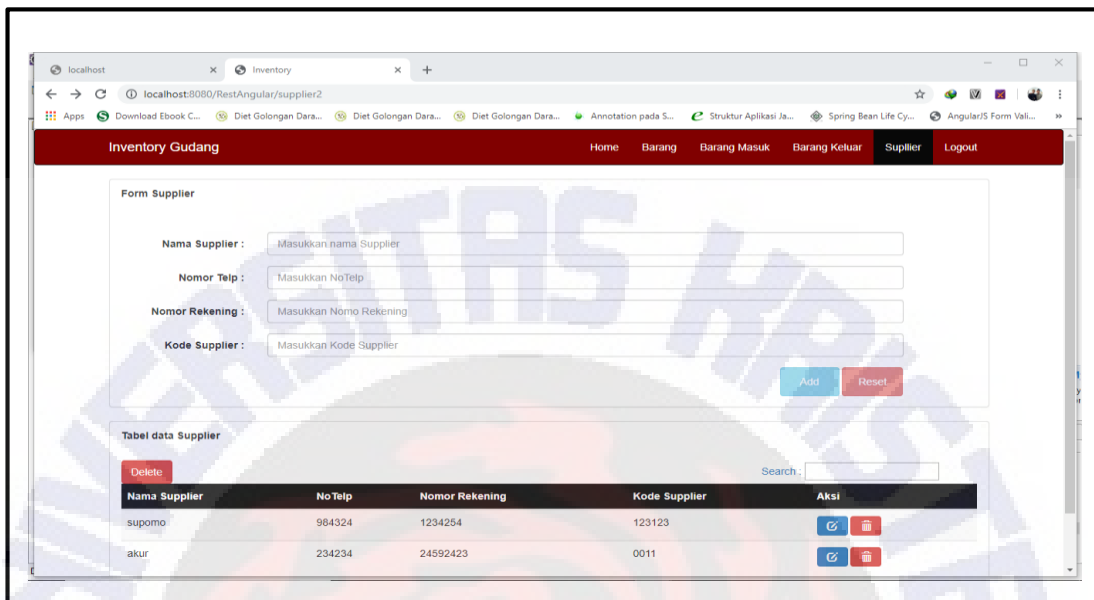
The screenshot shows a web application interface for 'Inventory Gudang'. The navigation bar includes 'Home', 'Barang', 'Barang Masuk', 'Barang Keluar', 'Supplier', and 'Logout'. The 'Form Barang' section contains the following fields:

- Kode Barang:
- Tgl:
- Nama Barang:
- Stok:
- Harga:

Buttons for 'Add' and 'Reset' are located at the bottom right of the form. Below the form is the 'Tabel data Barang' section, which includes a 'Delete' button, a search input, and an 'Excel' button. The table has the following columns:

Tgl	Nama Barang	Stok	Harga	Aksi
-----	-------------	------	-------	------

Halaman barang keluar ditunjukkan pada Gambar 12 dimana terdapat beberapa *text field* dan *drop down*. Pada halaman ini terdapat beberapa variabel yaitu kode barang, tgl keluar, nama barang, stok, dan harga barang. Pada halaman ini juga menyajikan data barang yang sudah diolah



**Gambar 13** Halaman *Supplier*

Gambar 13 merupakan halaman *supplier*, pada halaman ini terdapat *form* dengan beberapa *text field*. *Text field* ini berupa nama *supplier*, nomor telepon, nomor rekening, dan kode *supplier*. Pada halaman ini juga terdapat tabel untuk melihat hasil pengolahan data *supplier*.

Tahapan selanjutnya yang perlu dilakukan adalah pengujian sistem. Pengujian sistem yang dilakukan pada aplikasi ini menggunakan metode *blacbox* yang merupakan pengujian fungsionalistas dari perangkat lunak untuk menentukan beberapa kebutuhan yang diharapkan tanpa memperhatikan alur eksekusi program melainkan apakah setiap fungsi pada aplikasi berjalan dengan semestinya.. Pengujian sistem ini dapat ditunjukkan pada Tabel 1.

**Tabel 1** Hasil Pengujian *Black-Box*

Fungsi yang diuji	Kondisi	Output yang Diharapkan	Output yang dihasilkan sistem	Status Pengujian
<i>Login</i>	<i>Username dan password benar</i>	Sukses <i>Login</i>	Sukses <i>Login</i>	<i>Valid</i>
	<i>Username dan password salah</i>	Gagal <i>Login</i>	Gagal <i>Login</i>	<i>Valid</i>

<b>Fungsi yang diuji</b>	<b>Kondisi</b>	<b>Output yang Diharapkan</b>	<b>Output yang dihasilkan sistem</b>	<b>Status Pengujian</b>
Tambah data barang	<i>Form</i> diisi dengan benar	Sukses tambah data baru	Sukses tambah data baru	<i>Valid</i>
Tambah data barang masuk	<i>Form</i> diisi dengan benar	Sukses tambah data barang masuk baru	Sukses tambah data barang masuk baru	<i>Valid</i>
Tambah data barang keluar	<i>Form</i> diisi dengan benar	Sukses tambah data barang keluar baru	Sukses tambah data barang keluar baru	<i>Valid</i>
Tambah data <i>supplier</i>	<i>Form</i> diisi dengan benar	Sukses tambah data <i>supplier</i> baru	Sukses tambah data <i>supplier</i> baru	<i>Valid</i>
<i>Edit</i> data barang	<i>Form</i> diisi dengan benar	Sukses edit data barang	Sukses <i>edit</i> data barang	<i>Valid</i>
<i>Edit</i> data barang masuk	<i>Form</i> diisi dengan benar	Sukses <i>edit</i> data barang masuk	Sukses <i>edit</i> data barang masuk	<i>Valid</i>
<i>Edit</i> data Barang keluar	<i>Form</i> diisi dengan benar	Sukses <i>edit</i> data barang Keluar	Sukses <i>edit</i> data barang keluar	<i>Valid</i>
Tampil data barang	<i>Form</i> diisi dengan benar	Sukses tampil data barang	Sukses tampil data barang	<i>Valid</i>
Tampil data barang masuk	<i>Form</i> diisi dengan benar	Sukses tampil data barang masuk	Sukses tampil data barang masuk	<i>Valid</i>
Hapus data barang	Pilih data barang pilih <i>button</i> hapus	Sukses hapus data barang	Sukses hapus data barang	<i>Valid</i>
Hapus data barang masuk	Pilih data barang masuk <i>klik button</i> hapus	Pilih data barang masuk <i>klik button</i> hapus	Sukses hapus data barang masuk	<i>Valid</i>

Berdasarkan hasil pengujian pada Tabel 1, dapat disimpulkan bahwa sistem telah berjalan sesuai dengan kebutuhan sistem yang diharapkan. Pengujian beta adalah pengujian yang dilakukan secara objektif dengan kata lain pengujian ini adalah pengujian secara langsung di lapangan atau tempat dimana aplikasi yang dibuat



diimplementasikan, pengujian ini bertujuan untuk mengetahui apakah sistem telah memenuhi kebutuhan pengguna. Untuk mengetahui hasil dari pengujian *beta* digunakan kuesioner yang diberikan kepada pengguna sistem. *Skala likert* merupakan skala yang digunakan untuk mengukur pendapat atau persepsi seseorang maupun kelompok mengenai sebuah peristiwa atau fenomena sosial, dengan menggunakan skala 1 sampai 4.

**Tabel 2** Skala *Likert*

Nomor	Keterangan
1	Sangat Setuju
2	Setuju
3	Kurang setuju
4	Tidak Setuju

Responden yang digunakan berjumlah 5 orang. Pertanyaan untuk kuisisioner dapat dilihat pada Tabel 3. Presentase masing-masing jawaban dicari dari data hasil kuisisioner dengan menggunakan rumus kuisisioner :

$$Y = P/Q*100$$

Keterangan :

P : Banyaknya Responden dari setiap soal

Q : Jumlah responden

Y : Nilai persentase

**Tabel 3** Daftar pertanyaan Kuisisioner

Nomor	Pertanyaan
1	Apakah aplikasi dapat membantu proses Pendataan barang?
2	Dapatkah aplikasi membantu proses pengambilan keputusan?
3	Apakah aplikasi Mudah digunakan?
4	Tampilan aplikasi menarik?
5	Apakah membantu proses pengolahan data barang?

Berdasarkan hasil pengujian yang telah dilakukan, dapat disimpulkan bahwa pertanyaan pertama 50% responden memberikan pendapat sangat setuju dan 50% responden setuju. Pada pertanyaan kedua responden memberikan tanggapan 50% sangat setuju dan 50% setuju. Sedangkan pada pertanyaan ketiga responden memberikan tanggapan setuju dengan presentase 100%. Kemudian pada pertanyaan ketiga responden memberikan tanggapan 50% sangat setuju dan 50% setuju. Pada pernyataan keempat responden memberikan hasil presentase 100% pada kurang setuju. Sedangkan pada pertanyaan kelima responden memberikan presentase 50% sangat setuju dan 50% setuju. Berdasarkan hasil pengujian beta dapat disimpulkan bahwa sistem dapat membantu proses pengambilan keputusan dan pengelolaan barang.

## 5. Simpulan

Berdasarkan penelitian dan pengujian aplikasi yang telah dilakukan dapat diambil kesimpulan bahwa semua fungsi dapat berjalan baik dan sesuai dengan rancangan. Sistem yang dibangun dapat mempermudah proses pendataan barang, barang masuk, barang keluar, dan *supplier*. Sehingga memudahkan dalam pencarian data yang diperlukan dan adanya sistem ini dapat mengurangi terjadinya penumpukan kertas karena menggunakan *RESTful web service* sebagai *backend* baik untuk mengoptimalkan kinerja ketika sistem diakses, dan sistem informasi yang dibangun dapat mendata stok barang masuk, barang keluar, dan *supplier* yang sudah terkomputerisasi dan dengan adanya laporan berupa tabel dapat mengetahui data dan informasi yang lebih cepat, tepat dan akurat. Saran dalam pengembangan sistem perlu ditambahkan informasi data barang menggunakan grafik agar data yang ditampilkan lebih mudah dimengerti untuk dipahami oleh user. Penambahan fitur laporan transaksi yang dilakukan berupa barang masuk dan keluar dengan memiliki jangka waktu seperti mingguan, bulanan dan tahunan.

## 6. Daftar Pustaka

- [1] A. Anthony, A. R. Tanaamah, A. F. Wijaya, F. T. Informasi, U. Kristen, and S. Wacana, "BERDASARKAN STOK GUDANG BERBASIS *CLIENT SERVER* ( STUDI KASUS TOKO GROSIR ' RESTU ANDA ')," vol. 4, no. 2, pp. 136–147, 2017.
- [2] N. I. Masyhur, "Rancang Bangun *Web Service* ( Studi Kasus : Layanan SIM Inventaris Barang )," pp. 51–55, 2015.
- [3] O. Pahlevi, A. Mulyani, and M. Khoir, "Sistem Informasi Inventori Barang Menggunakan Metode *Object Oriented* Di PT. Livaza Teknologi Indonesia Jakarta," *Prosisko*, vol. 5, no. 1, pp. 27–35, 2018.
- [4] H. Agusvianto, "Sistem Informasi Inventori Gudang Untuk Mengontrol Persediaan Barang Pada Gudang, Studi Kasus : PT . Alaisys Sidoarjo," *J. Inf. Eng. Educ. Technol.*, 2017.
- [5] L. Rahmadi and K. Yusmiarti, "Perancangan Sistem Informasi *Inventory* Di AMIK Lembah Dempo Pagaralam," in *Seminar Nasional Teknologi Informasi Dan Multimedia*, 2016.
- [6] S. Suprawiro, "Analisis dan Perancangan Sistem Informasi *Inventory* Sparepart Kapal Berbasis Web : Studi Kasus Asia Group Pangkalpinang," vol. 06, no. September, pp. 121–129, 2017.
- [7] M. I. Perkasa and E. B. Setiawan, "Pembangunan *Web Service* Data Masyarakat Menggunakan *REST API* dengan *Access Token*," vol. X, no. 1, pp. 19–26, 2018.
- [8] P. F. Tanaem, D. Manongga, and A. Iriani, "*RESTful Web Service* Untuk Sistem Pencatatan Transaksi Studi Kasus PT . XYZ," vol. 2, no. April, 2016.
- [9] D. I. Sensuse, "Model Integrasi Sistem dengan Pendekatan Metode *Service Oriented Architecture* dan *Model View Controller* pada Pusat Penelitian Perkembangan IPTEK Lembaga Ilmu Pengetahuan Indonesia," vol. 3, no. April, pp. 84–103, 2017.
- [10] A. Rahmatulloh, H. Sulastri, and R. Nugroho, "Keamanan *RESTful Web Service* Menggunakan

