

Pengembangan Sistem Informasi Manajemen *Supplier* dan Barang dengan *Extreme Programming*

<http://dx.doi.org/10.28932/jutisi.v6i1.2132>

Astria Hijriani^{#1}, Jannati Asri Safitri [✉]^{#2}, Rd Irwan Adi Pribadi^{#3}, Rico Andrian^{#4}

[#]Jurusan Ilmu Komputer, Universitas Lampung

Jl. Prof. Sumantri Brojonegoro No.1, Gedong Meneng, Bandar Lampung

¹astria.hijriani@fmipa.unila.ac.id

²jannati30fitri@gmail.com

³irwan.adipribadi@fmipa.unila.ac.id

⁴rico.andrian@fmipa.unila.ac.id

Abstract — The case study was taken from one of trading companies in Lampung. The company sells Muslim fashion products from a large number of suppliers. Suppliers data is recorded in detail manually, as well as products recorded. Manual data collection can result in recording errors, data easily tugged, or not recorded. This research develops an information system to help the company in data collection of suppliers and products automatically based on web using Laravel as a framework. This system is built using extreme programming methods and has features that focus on collecting suppliers, products, and product shipments. The results of system testing using the black box testing method shows that the system has fulfilled functional requirements and user needs.

Keywords— Management Information System; Product; Supplier.

I. PENDAHULUAN

Produk merupakan segala sesuatu yang dapat ditawarkan ke pasar atau konsumen untuk mendapatkan perhatian, dibeli, digunakan, atau di konsumsi yang dapat memuaskan keinginan atau kebutuhan [1]. Pakaian merupakan suatu produk yang dibutuhkan oleh setiap individu. Berbagai jenis pakaian tersedia dalam berbagai pilihan mulai dari bahan, model, warna, dan lain-lain. Studi kasus penelitian ini diambil dari salah satu perusahaan yang ada di Lampung. Perusahaan tersebut menjual pakaian muslim dengan berbagai pilihan busana yang dibutuhkan oleh konsumen mulai dari pakaian muslim dewasa hingga anak-anak.

Pakaian tersebut disuplai oleh *supplier*. Perusahaan tersebut memiliki kurang lebih seratus *supplier*. *Supplier* yang terdaftar di perusahaan akan didata secara manual. *Supplier* akan mengirimkan barang saat *owner* menghubungi untuk melakukan pembelian barang. Barang yang dikirim oleh *supplier* dan telah sampai di perusahaan

akan dilakukan pencatatan transaksi pembelian barang dari faktur. Pencatatan transaksi pembelian barang dari *supplier* nantinya dimaksudkan sebagai modul pembelian pada sistem ini. Barang yang telah sampai juga dilakukan pemeriksaan untuk mengetahui kesesuaian barang apakah terjadi kerusakan atau perbedaan jumlah barang yang diminta. Jika terdapat barang yang rusak, maka akan dilakukan *return* barang. Barang yang sudah selesai diperiksa dan tidak ada masalah akan didata secara manual ke dalam *Microsoft Excel*.

Pendataan *supplier* dan barang yang dilakukan secara manual akan menyebabkan terjadinya kesalahan pendataan, data terselip, bahkan terdapat data *supplier* atau barang tidak tercatat di *Microsoft Excel*. Pendataan secara manual juga membutuhkan waktu yang lama dan kurang efisien.

Masalah lainnya yaitu terjadi dalam hal pengiriman barang, proses bisnis yang terjadi dalam hal pengiriman barang di perusahaan yaitu, barang akan dikirim ketika *owner* menghubungi *supplier* untuk melakukan pembelian barang kembali (*restock*). Pengiriman barang yang dikirim oleh *supplier* membutuhkan waktu beberapa hari dari pengemasan hingga barang sampai di perusahaan. Pengiriman yang membutuhkan waktu beberapa hari akan mengakibatkan kekosongan stok barang di gudang.

Beberapa *supplier* di perusahaan bahkan ada yang berinisiatif mengirimkan barang ke perusahaan sebelum dihubungi oleh *owner*. *Supplier* tersebut berpikir, barang yang mereka kirimkan sebelumnya sudah habis di perusahaan tersebut. Karena hal tersebut timbul masalah terjadi penumpukan stok ketika barang yang dikirim masih banyak dan ditambah lagi dengan barang baru, padahal barang tersebut kurang diminati oleh pembeli.

Supplier dan pihak perusahaan harus memiliki hubungan relasi yang baik dalam berbisnis agar kedua belah pihak sama-sama diuntungkan sehingga tidak terjadi kesalahpahaman dalam berkomunikasi seperti halnya

masalah pengiriman barang secara inisiatif oleh *supplier*. Hubungan antara *Supplier* dan perusahaan ini dikenal dengan *Supplier Relationship Management (SRM)*. *Supplier Relationship Management* adalah pendekatan secara komprehensif untuk mengelola interaksi, komunikasi, dan bagaimana bekerja lebih efektif dengan *Supplier* agar sama-sama saling menguntungkan. *Supplier* dan perusahaan harus bersama-sama mencapai tujuan mereka [2].

Penelitian yang membahas tentang SRM salah satunya yaitu “Analisis Pengaruh *Top Manajement Commitment* terhadap *Supplier Performance* dan *Supplier Relationship Manajemen* pada Hotel Budget di Surabaya”. Penelitian tersebut membahas pentingnya *Supplier Relationship Manajemen* yang berdampak pada peningkatan kinerja perusahaan. Pada penelitian tersebut dijelaskan bagaimana terciptanya hubungan yang baik antara perusahaan dan *supplier* mereka. Dengan terciptanya hubungan tersebut, maka memberikan kemudahan dalam proses kerja yang dilakukan perusahaan. Salah satunya adalah dengan mengurangi resiko kesalahan dalam melakukan pesanan barang ke *supplier*, kesalahan pengiriman oleh pihak *supplier*, dapat menurunkan biaya transaksi atau mungkin dapat dihilangkan karena adanya hubungan yang baik di kedua belah pihak. Analisis tersebut menyebutkan sangat penting untuk menggandeng *supplier* yang mampu mengerti visi dan misi perusahaan sehingga dapat menyelaraskannya dalam rangka mengembangkan bisnis hingga masa mendatang [3].

Berdasarkan latar belakang tersebut, yang menjadi tujuan penelitian ini yaitu membuat sistem informasi yang pendataan *supplier* dan barang menggunakan *framework* Laravel. Tujuan lainnya yaitu membuat sistem informasi yang dapat memberikan notifikasi kepada *supplier* ketika stok barang mereka hampir habis. Notifikasi tersebut dibuat agar tidak ada lagi *supplier* yang berinisiatif mengirimkan barang secara random sehingga dapat mengurangi *miss communication* antara *supplier* dan perusahaan serta tercipta hubungan *supplier relationship management* yang baik antara perusahaan dan *supplier*.

Pada pelaksanaannya penelitian ini tergabung dalam satu tim yang beranggotakan empat orang. Penelitian tersebut memiliki empat pembagian modul berbeda yakni pembelian, penjualan, *dashboard*, dan akuntansi. Jadi, pada penelitian ini, yang di fokuskan adalah modul pembeliannya saja yang mencakup pendataan *supplier*, pembelian, dan pendataan barang. Sedangkan untuk modul lainnya akan diimplementasikan oleh anggota tim yang lain sesuai dalam pembagian modulnya.

Pengimplementasian sistem ini akan bermanfaat untuk perusahaan karena dapat meminimalisir kesalahan dalam menginputkan data dan juga data terselip, membantu pegawai dalam mengefisienkan waktu kerja, dan membuat *owner* dapat mengontrol serta mengawasi persediaan barang yang ada di perusahaan. Manfaat lainnya yaitu *supplier* akan dapat melihat jumlah barang mereka yang hampir habis

lewat notifikasi.

Batasan masalah pada penelitian yaitu hanya membuat atau berfokus pada modul pembelian barang. Nantinya sistem informasi ini dapat melakukan pendataan *supplier*, barang, dan juga transaksi pembelian. Penelitian ini akan berbasis *web* dan menggunakan *framework* Laravel. Sistem informasi ini merupakan sistem yang berkaitan dengan tiga modul lainnya yaitu modul penjualan (sistem *customer relationship management*), modul *dashboard* (sistem *business intelligence dashboard*), dan modul akuntansi (sistem informasi akuntansi). Sistem informasi ini akan berfokus pada pendataan *supplier*, pendataan barang, dan pendataan transaksi. Sistem ini juga akan berkaitan dengan hubungan *supplier* dan pihak perusahaan melalui pendekatan SRM.

II. TINJAUAN PUSTAKA

A. Penelitian Terdahulu

Pada jurnal yang berjudul “Sistem Informasi *Inventori Gudang Untuk Mengontrol Persediaan Barang Pada Gudang Studi Kasus: PT Alaisys Sidoarjo*”. Penelitian tersebut menjelaskan pendataan gudang yang masih manual seperti pencatatan informasi penjualan dan persediaan barang menggunakan bon nota buku pencatatan dan laporan yang masih ditulis tangan. Untuk mendapatkan data yang dibutuhkan harus mencari satu persatu nota yang telah disimpan. Penelitian tersebut menghasilkan sebuah sistem inventori gudang berbasis *web* yang dapat mendata barang, transaksi, dan laporan barang. Sistem ini membantu perusahaan dalam mengawasi dan mengontrol barang sehingga dapat meningkatkan efektivitas perusahaan. Sistem ini juga memudahkan karyawan dalam melakukan bagian pengontrolan persediaan barang [4]. Keterkaitan penelitian ini dengan penelitian yang akan dilakukan adalah bagaimana membuat sistem informasi yang dapat mendata barang dan transaksi secara otomatis.

Penelitian lainnya yaitu berjudul “Perancangan Sistem Stok Barang di *Warehouse* Berbasis *Web*”. Penelitian ini mengambil studi kasus pada PT. Mulia Andalan Gemilang. Pendataan stok barang masih dengan cara manual atau masih mencatat kedalam buku. Pencatatan meliputi data barang masuk dari *supplier* dan data barang keluar sebagai barang terjual kepada *customer*. Dengan proses pencatatan data yang tradisional itu perusahaan mengalami banyak kerugian, yaitu kesalahan dalam perhitungan stok barang secara manual dan pembuatan laporan juga terlambat. Penelitian tersebut menghasilkan sistem berbasis *web* yang menerapkan teknologi komputer secara keseluruhan dalam pencatatan stok gudang agar meningkatkan kinerja operasi perusahaan. Manfaat sistem ini yaitu dapat menghemat waktu pemrosesan data dan lebih cepat dalam pembuatan laporan stok barang masuk dan keluar [5]. Keterkaitan antara penelitian ini dengan penelitian yang dilakukan adalah bagaimana membuat sistem yang dapat mendata stok barang secara otomatis agar lebih efektif dan efisien.

Penelitian selanjutnya berjudul “Perancangan dan Implementasi Sistem Informasi Persediaan Bahan Baku dan Suku Cadang di PT. Triteguh Manunggal Sejati Banjarbaru”. Dalam penelitian ini pengelolaan persediaan barang masih dilakukan secara manual dan belum terkomputerisasi. Perusahaan itu masih menggunakan media kertas sehingga pencarian data dilakukan dengan cara menelusuri buku besar yang sangat menyita waktu. Setiap entitas belum terintegrasi yang menyebabkan keterlambatan kinerja dan kurang akuratnya informasi. Masalah lain yaitu belum adanya aplikasi untuk mengingatkan apabila ada stok barang yang sudah mencapai titik pesan kembali (*reorder point*), yang harus segera dipesan untuk menghindari kehabisan stok. Analisis kebutuhan sistem yang dilakukan menghasilkan kebutuhan fungsional yaitu sistem harus dapat melakukan pendataan barang, pendataan karyawan, pendataan *supplier*, dan pencetakan laporan. Keterkaitan penelitian ini dengan penelitian yang akan dilakukan adalah membuat sistem yang dapat melakukan pendataan secara otomatis untuk pendataan barang, pendataan *supplier*, dan pemberitahuan saat stok barang hampir habis [6].

B. Supplier

Menurut Bunafit Nugroho dalam bukunya, *supplier* adalah pihak yang menjadi penyedia barang yang dijual pada perusahaan/ toko, bisa disebut juga dengan distributor atau agen yang mendatangkan barang dagangan ke sebuah perusahaan/ toko. Perusahaan/ toko itu sendiri juga dapat bertindak sebagai *supplier*, misalnya saja ada sebagian produk yang dijual adalah produk rumahan buatan dari perusahaan itu sendiri [7].

C. Supplier Relationship Management (SRM)

Supplier relationship management (SRM) adalah istilah umum tentang memutuskan level intervensi, tingkat, dan hubungan dasar yang dibutuhkan dengan pemasok (*Supplier*). Sumber daya yang diarahkan sebagian besar hanya untuk bekerja dengan para pemasok, namun tidak menyadari nilai yang signifikan dan berharga yang akan kita dapatkan dari pemasok [8].

Supplier Relationship Management adalah salah satu pendekatan untuk menghubungkan rantai pasokan dan pemasok dengan kepentingan strategis organisasi untuk berhasil mengidentifikasi dan mendorong peluang bisnis yang belum tersentuh. Pendekatan SRM memberikan kerangka kerja yang jelas untuk mengelola hubungan di semua tingkat, fungsi yang berbeda, mengurangi tingkat kerumitan, dan meningkatkan transparansi di kedua sisi [9].

SRM yaitu pendekatan sistematis untuk mengembangkan dan mengelola kemitraan. Keberhasilan SRM juga sangat bergantung pada kompetensi pemasok (*supplier*). Kompetensi pemasok (*supplier*) adalah sumber daya, proses dan ukuran pemasok yang ditawarkan kepada perusahaan pembeli. *Supplier relationship management* sering dianggap sebagai proses operasional/taktis yang akan menyelesaikan masalah dan mengurangi biaya [10].

D. Pendataan

Pendataan adalah proses pembuktian yang ditemukan dari hasil penelitian yang dapat dijadikan dasar kajian atau pendapat. Secara teknis pengertian pendataan adalah proses yang lebih berkaitan dengan pengumpulan datanya secara empiris. Menurut penulis pendataan dapat diartikan sebagai proses pembuktian dari hasil penelitian dan lebih dikaitkan dengan pengumpulan secara empiris [11].

E. Extreme Programming

Extreme Programming (XP) adalah model yang banyak digunakan untuk mengembangkan perangkat lunak dengan kualitas tinggi dalam metode pengembangan *Agile*. *Extreme Programming* (XP) berfokus pada peningkatan kualitas perangkat lunak dan juga berfokus pada respons terhadap perubahan kebutuhan [12].

Pada metode *Agile*, sistem dibangun dengan desain yang sederhana, pengujian dilakukan pada tahap pengembangan, dan dokumentasi perangkat lunak yang cukup. *Extreme programming* (XP) cocok untuk individu, tim kecil, dan juga tim menengah. *Extreme programming* dalam pelaksanaannya mengutamakan komunikasi yang baik antar anggota tim. Pengerjaan menggunakan metode ini dilakukan secara *incremental* dan interaktif untuk menghadapi kebutuhan yang berubah dengan cepat [13].

Metode *Extreme Programming* sering juga dikenal dengan metode XP. Metode ini dicetuskan oleh seorang pakar *software engineering* yaitu Kent Beck. XP adalah suatu model pengembangan perangkat lunak yang menyederhanakan berbagai tahapan pengembangan menjadi lebih efisien, adaptif dan fleksibel [14].

Nilai dasar metode *extreme programming* yaitu:

1. *Communication*: memfokuskan komunikasi antara *programmer* dengan *user*.
2. *Courage*: pengembang harus memiliki keyakinan, keberanian dan integritas dalam melakukan tugasnya.
3. *Simplicity*: lakukan semua dengan sederhana.
4. *Feedback*: mengandalkan *feedback*.
5. *Quality work*: proses berkualitas berimplikasi pada perangkat lunak yang berkualitas sebagai hasil akhirnya.

III. METODOLOGI

F. Metode Pengumpulan Data

1. Wawancara

Wawancara merupakan salah satu metode pengumpulan data yang digunakan untuk memperoleh informasi dari narasumber. Wawancara dilakukan dengan berinteraksi atau memberikan pertanyaan kepada narasumber terkait dengan informasi yang dibutuhkan dalam melakukan penelitian. Pengembang telah melakukan wawancara sebanyak 4 kali dengan *owner*

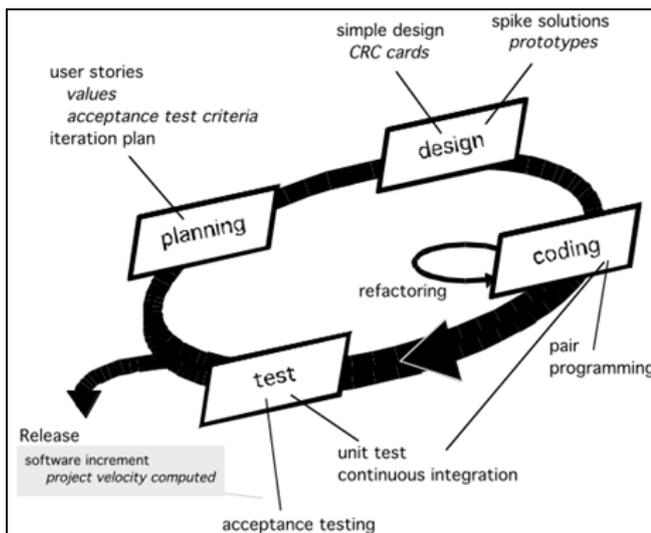
perusahaan. Wawancara pertama mengenai sejarah, wawancara kedua mengenai masalah yang terjadi, wawancara ketiga tentang apa yang dibutuhkan, dan wawancara keempat melakukan validasi apa yang dibutuhkan oleh pihak perusahaan.

2. Observasi

Observasi merupakan salah satu teknik pengumpulan data. Pengembang melakukan pengamatan secara langsung ke objek penelitian untuk melihat dari dekat kegiatan apa saja yang dilakukan dan mendapatkan informasi yang akan dibutuhkan untuk melancarkan penelitian tersebut. Pengembang telah melakukan observasi sebanyak 3 kali. Observasi pertama dilakukan untuk mengetahui proses bisnis perusahaan, observasi kedua dan ketiga dilakukan untuk mendapatkan *fitur-fitur* yang akan dibuat ke dalam sistem.

G. Metode Pengembangan Sistem

Metodologi yang digunakan dalam pembuatan sistem informasi ini yaitu metode *extreme programming* (XP). *Extreme programming* memiliki beberapa tahapan dalam pelaksanaannya yaitu *planning*, *desain*, *coding*, dan *tahap testing* [15]. Tahapan-tahapan yang ada pada metode *extreme programming* tersebut dapat dilihat pada Gambar 1.



Gambar 1. Metode *Extreme Proerammine* (XP) [16]

1. *Planning* (Perencanaan)

Dalam tahap ini akan dimulainya analisis tentang pemahaman konteks sistem informasi yang akan dibuat yang dapat dilihat pada *user stories*, pengembang

mendefinisikan keluaran (*output*), fitur yang akan di buat, fungsi aplikasi, dan juga menjelaskan alur pengembangan aplikasi.

2. *Design* (Perancangan)

Tahap ini dilakukan membuat desain sistem informasi manajemen untuk pendataan *supplier* dan barang secara sederhana. Salah satu cara untuk mendesain aplikasi biasanya digunakan pemetaan untuk membangun kelas-kelas yang akan digunakan seperti *use case diagram*, *class diagram*, atau diagram objek.

3. *Coding* (Pengkodean)

Tahap ini dalam pengembangan aplikasi dengan menggunakan metode *Extreme programming* akan dilakukan proses *coding* untuk membangun sistem informasi manajemen *supplier* dan barang.

4. *Testing* (Pengujian)

Tahap ini memfokuskan pada pengujian fitur-fitur yang ada di dalam sistem informasi manajemen *supplier* dan barang. Sistem akan di uji sehingga tidak ada kesalahan (*error*) dan aplikasi yang dibuat sesuai dengan kebutuhan pengguna (*user*).

H. Analisis Kebutuhan Masalah

Dalam perusahaan tersebut pelaksanaan proses bisnisnya masih dilakukan secara manual sehingga banyak terjadi kesalahan pendataan informasi. Berdasarkan proses bisnis yang terjadi pemilik perusahaan membutuhkan informasi secara akurat dan efisien. Karena hal tersebut pengembang melakukan observasi dan wawancara untuk membuat *user stories*. *User stories* merupakan salah satu bagian dari tahap *planning* pada metode XP. Berikut *user stories owner*:

- Sebagai seorang *owner*, saya ingin adanya pendataan *supplier* secara sistematis karena di perusahaan saya pendataan *supplier* masih secara manual.
- Sebagai seorang *owner*, saya ingin adanya pendataan barang secara sistematis mulai dari data kategori barang, satuan barang, rak barang, jenis barang, dan *pricing* barang.
- Sebagai seorang *owner*, saya ingin adanya pendataan transaksi pembelian barang dari *supplier* ke perusahaan dan pendataan *return/* pengembalian barang dari perusahaan ke *supplier*.
- Sebagai seorang *owner*, saya ingin *supplier* mengetahui saat stok barang mereka hanya tersisa sedikit di perusahaan.

- e. Sebagai seorang *owner*, saya ingin *supplier* mengetahui daftar barang mereka yang tidak perlu mereka kirimkan ke perusahaan.

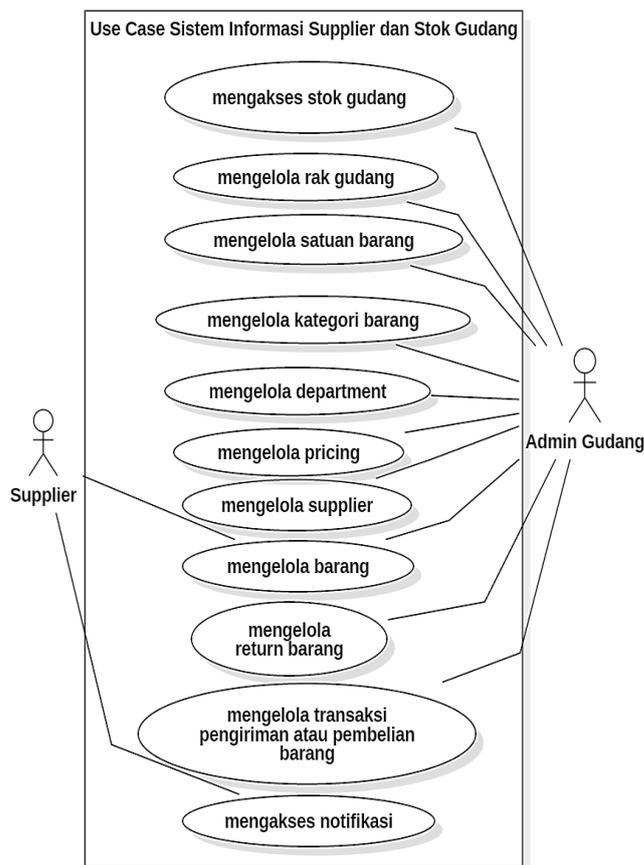
D. Analisis Kebutuhan Sistem

Setelah melihat *user stories* yang sudah disampaikan oleh *owner*, maka akan dibuat sistem informasi manajemen *supplier* dan barang berbasis *web* dimana terdiri dari beberapa komponen dalam pengembangannya yaitu menggunakan *framework* Laravel, CSS dan Bootstrap.

Penerapan sistem pada proses bisnis yang terjadi di perusahaan nantinya akan membuat informasi yang didata secara manual menjadi otomatis sehingga mempermudah dalam menjalankan proses selanjutnya. Sistem ini dapat mendata seluruh data pembelian produk dari *supplier*, data barang, dan data *supplier*. Sistem ini akan memudahkan pihak perusahaan dalam pendataan sehingga dapat meminimalisir kesalahan dalam penginputan data informasi barang ataupun *supplier*.

E. Perancangan Sistem

Perancangan untuk sistem ini akan diperlihatkan pada diagram *use case*. Gambar 2 memperlihatkan *use case diagram* yang menunjukkan bahwa terdapat dua level *user* yaitu Admin Gudang dan *supplier*. Admin Gudang dapat mengakses stok gudang, mengelola barang, rak gudang, satuan barang, kategori barang, transaksi pengiriman pembelian barang, *return* barang, dan mengelola *supplier*. *Supplier* akan dapat mengakses barang dimana didalamnya terdapat fitur barang tunda dan mengakses notifikasi.



Gambar 2. Use case sistem informasi manajemen *supplier* dan barang.

IV. HASIL DAN PEMBAHASAN

A. Hasil Penelitian

Sistem informasi manajemen *supplier* dan barang adalah sistem yang dibuat untuk memudahkan pemilik perusahaan dalam melakukan pendataan stok barang dan *supplier*. Sistem ini juga memudahkan *supplier*/ pemasok untuk melihat pergerakan barang mereka lewat pemberitahuan (*notification*) yang ada di sistem. Sistem informasi manajemen *supplier* dan barang memiliki dua tingkatan *user* yaitu petugas gudang yang bertugas sebagai *admin* dan *supplier*. Petugas gudang bertugas mengelola Sistem informasi manajemen *supplier* dan barang.

Petugas gudang memiliki akses untuk mengelola *supplier*, mengelola barang, mengelola pengiriman barang dari *supplier*, mengelola *return* pembelian barang *supplier*, mengelola kategori, mengelola harga, mengelola stok, mengelola rak di gudang, mengelola satuan, mengelola jenis kode barang, dan mengelola *user*. *Supplier* memiliki akses untuk melihat stok barang dan transaksi pengiriman barang mereka yang ada di perusahaan. *Supplier* juga dapat melihat pemberitahuan yang muncul saat stok barang mereka di perusahaan sudah dalam batas minimal sehingga para *supplier* dapat menghubungi *owner* untuk mengonfirmasi

apakah dilakukan *restock* barang atau tidak. *Role supplier* pada sistem ini ditujukan untuk menciptakan relasi dan komunikasi yang baik antara *supplier* dan perusahaan. Hubungan yang baik antara *supplier* dan perusahaan dapat disebut *Supplier Relationship Management*.

Pengembangan sistem ini menggunakan metodologi *Extreme programming*. Sistem informasi manajemen *supplier* dan barang berbasis *web*, diakses secara *online*, dan dibangun dengan menggunakan bahasa pemrograman PHP juga *Javascript*. PHP (*Hypertext PreProcessor*) digunakan sebagai bahasa pemrograman *server side* yang berjalan pada sisi *server*, dan *Javascript* digunakan sebagai bahasa pemrograman *client side* yang berjalan pada sisi *client* atau *browser*. Sistem informasi manajemen *supplier* dan barang dalam pengembangannya dibangun dengan menggunakan *framework* Laravel.

Atom Text Editor versi 1.25.1 untuk menuliskan kode program. Balsamiq Mockup versi 3.5.8 untuk membuat desain *interface*. XAMPP 7.1.10 sebagai *local server* dalam pengembangan sistem MySQL dan PHP. Star UML untuk membuat diagram model sistem. Web Browser Google Chrome untuk menguji dan menjalankan sistem. Laptop Lenovo Z40-75 sebagai alat bantu mengembangkan sistem.

Windows 10 64 bit sebagai sistem operasi untuk menjalankan berbagai aplikasi.

B. Pembahasan

Pencatatan data supplier dan barang merupakan komponen penting dalam perusahaan. Maka dibuat sistem untuk menangani pendataan *supplier* dan barang agar proses bisnis yang ada dapat berjalan dengan baik. Sistem informasi manajemen *supplier* dan barang dikembangkan menggunakan metode *extreme programming* (XP). XP memiliki tahapan yaitu *planning*, *design*, *coding*, dan *testing*. *User stories* yang mana masuk dalam tahap *planning* dan perancangan sistem yang masuk dalam tahap *design* telah dijelaskan pada bab 3 metodologi.

Sistem ini juga saling terintegrasi atau terhubung dengan tiga sistem yang berbeda, hal ini juga sempat disinggung pada bab pendahuluan. Ketiga sistem yang berbeda itu antara lain Sistem BID (*Business Intelligence Dashboard*), Sistem SIA (Sistem Informasi Akuntansi), dan Sistem CRM (*Customer Relationship Management*). Sistem SDB (*Supplier dan Barang*) yang memiliki data awal *supplier* dan barang sistem SDB (*Supplier dan Barang*) mengirim data ke semua sistem (BID, SIA, dan CRM). Sistem pendataan *supplier* dan barang mengirim data untuk *system customer relationship management* berupa data *supplier*, data barang, data kategori barang, dan data jenis kode barang agar pergerakan stok barang di perusahaan tersebut bisa dilakukan secara otomatis juga pendataan penjualan pada sistem CRM lebih sistematis, sistem ini juga mengirimkan data ke sistem *business intelligence dashboard* berupa data *supplier*, data kategori, dan data barang untuk diolah kembali menjadi informasi yang dapat membantu *owner* dalam melihat pergerakan bisnis di perusahaannya.

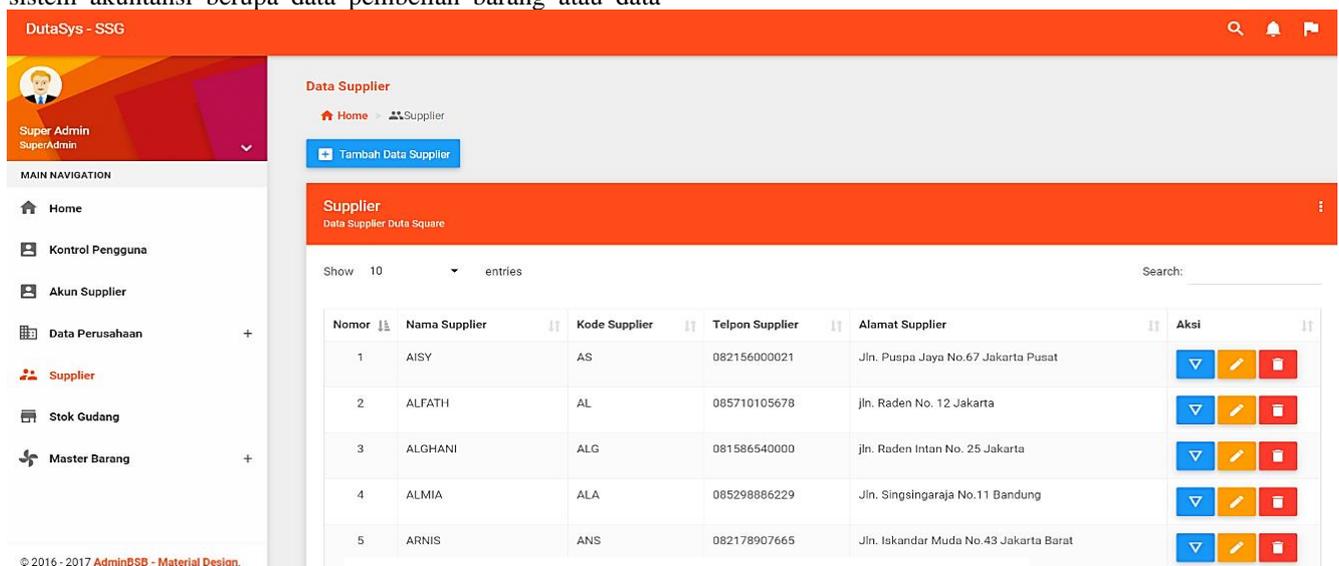
Sistem *supplier* dan barang juga mengirimkan data ke sistem akuntansi berupa data pembelian barang atau data

transaksi pembelian barang dari *supplier* yang mencakup total harga pembelian dan data *supplier* untuk diolah menjadi data yang dibutuhkan dalam hal keuangan perusahaan. Pengiriman data tersebut menggunakan API (*Application Programming Interface*) yang merupakan seperangkat antarmuka (bisa berbentuk fungsi, *method* atau URL *endpoint*) yang bisa digunakan untuk mengembangkan aplikasi, baik dalam satu platform maupun lintas platform. Berikut ini merupakan implementasi tampilan sistem yang berhasil dibuat menggunakan *framework* Laravel.

Sistem ini memiliki fitur pendataan *supplier*, barang, dan juga pembelian barang. Penjualan tidak di implementasikan pada sistem ini karena sudah diperuntukkan oleh anggota tim yang lain. Pada sistem ini terdapat fitur notifikasi sebagai bentuk implementasi dari konsep hubungan *supplier relationship management* atau SRM antara *supplier* dan perusahaan. Dalam fitur notifikasi *supplier* tidak hanya mengetahui sisa stok barang mereka yang ada di gudang perusahaan tapi juga terdapat informasi barang apa saja yang direkomendasikan oleh perusahaan untuk tidak dikirim ulang karena minat barang dari konsumen tidak terlalu tinggi sehingga dapat mengurangi biaya pengiriman yang percuma.

1. Tampilan Menu *Supplier*

Menu *supplier* diperlihatkan pada Gambar 3. Menu *supplier* digunakan untuk melakukan pendataan pada *supplier*. Pada menu *supplier* terdapat data berupa nama *supplier*, kode *supplier*, nomor telepon *supplier*, dan alamat *supplier*. Pada halaman ini juga admin dapat melakukan tambah data *supplier*, mengubah data *supplier*, melihat *detail supplier*, dan menghapus *supplier*. Admin dapat melakukan pencarian data *supplier* lewat *search* dengan memasukkan nama *supplier* atau kode *supplier* yang ingin di cari.



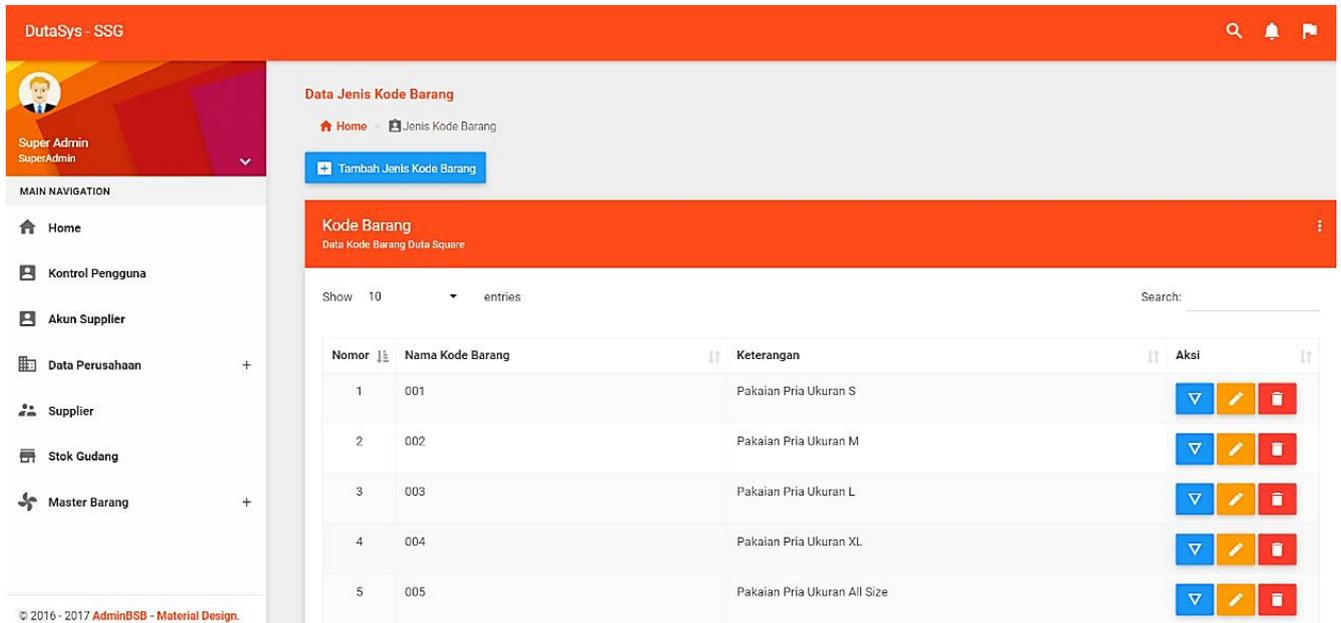
Nomor	Nama Supplier	Kode Supplier	Telpn Supplier	Alamat Supplier	Aksi
1	AISY	AS	082156000021	Jln. Puspa Jaya No.67 Jakarta Pusat	[Dropdown] [Edit] [Delete]
2	ALFATH	AL	085710105678	jln. Raden No. 12 Jakarta	[Dropdown] [Edit] [Delete]
3	ALGHANI	ALG	081586540000	jln. Raden Intan No. 25 Jakarta	[Dropdown] [Edit] [Delete]
4	ALMIA	ALA	085298866229	Jln. Singingaraja No.11 Bandung	[Dropdown] [Edit] [Delete]
5	ARNIS	ANS	082178907665	Jln. Iskandar Muda No.43 Jakarta Barat	[Dropdown] [Edit] [Delete]

Gambar 3. Tampilan halaman *supplier*.

2. Tampilan Menu Jenis Kode Barang

Halaman tampilan menu jenis kode barang diperlihatkan pada Gambar 4. Menu jenis kode barang di gunakan untuk membuat kode barang misal kode 001 sebagai kode untuk pakaian pria ukuran S. Pada menu jenis kode barang terdapat data berupa nomor, nama kode

barang, dan keterangan tentang jenis kode barang yang diinputkan. Pada halaman ini admin dapat melakukan tambah data jenis kode barang, mengubah, dan melihat *detail* jenis kode barang, dan menghapus jenis kode barang. Admin dapat melakukan pencarian data kode barang lewat *search* dengan memasukkan kode barang.

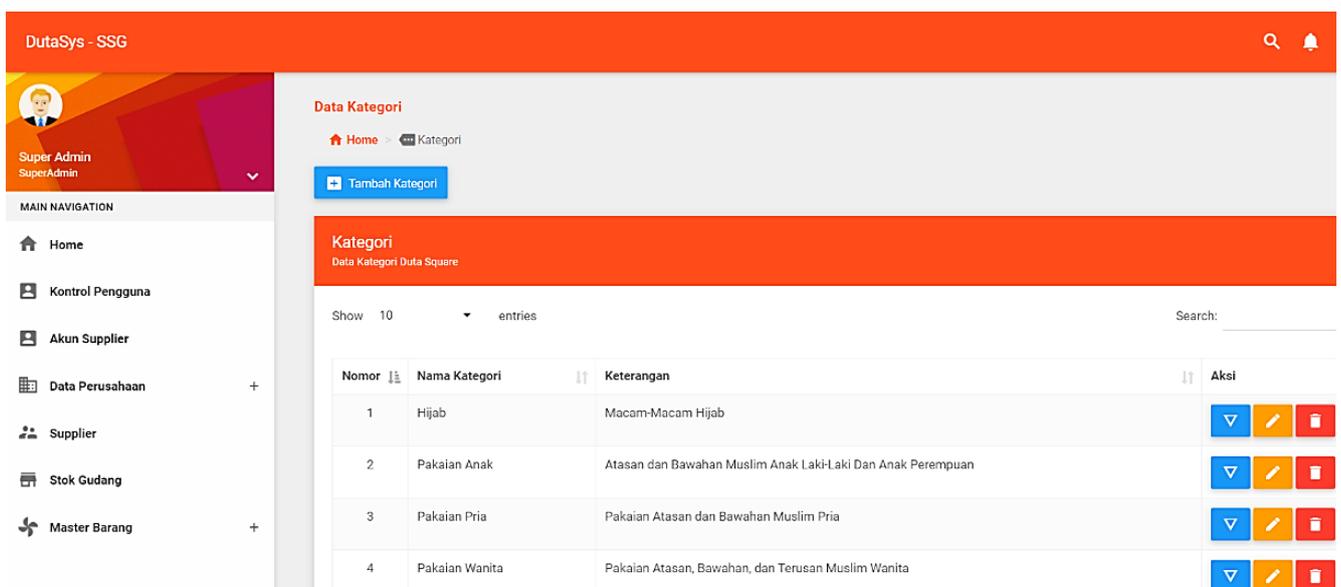


Gambar 4. Tampilan halaman kode jenis barang.

3. Tampilan Menu Kategori Barang

Menu kategori barang diperlihatkan pada Gambar 5. Menu kategori digunakan untuk membuat kategori barang misalnya kategori baju anak yang berisikan pakaian anak. Pada menu kategori terdapat data berupa nama kategori dan keterangan tentang kategori yang di-

inputkan. Pada halaman ini admin dapat melakukan tambah data kategori barang, mengubah, dan melihat *detail* kategori barang, dan menghapus kategori barang. Admin dapat melakukan pencarian data lewat *search* dengan memasukkan nama kategori.

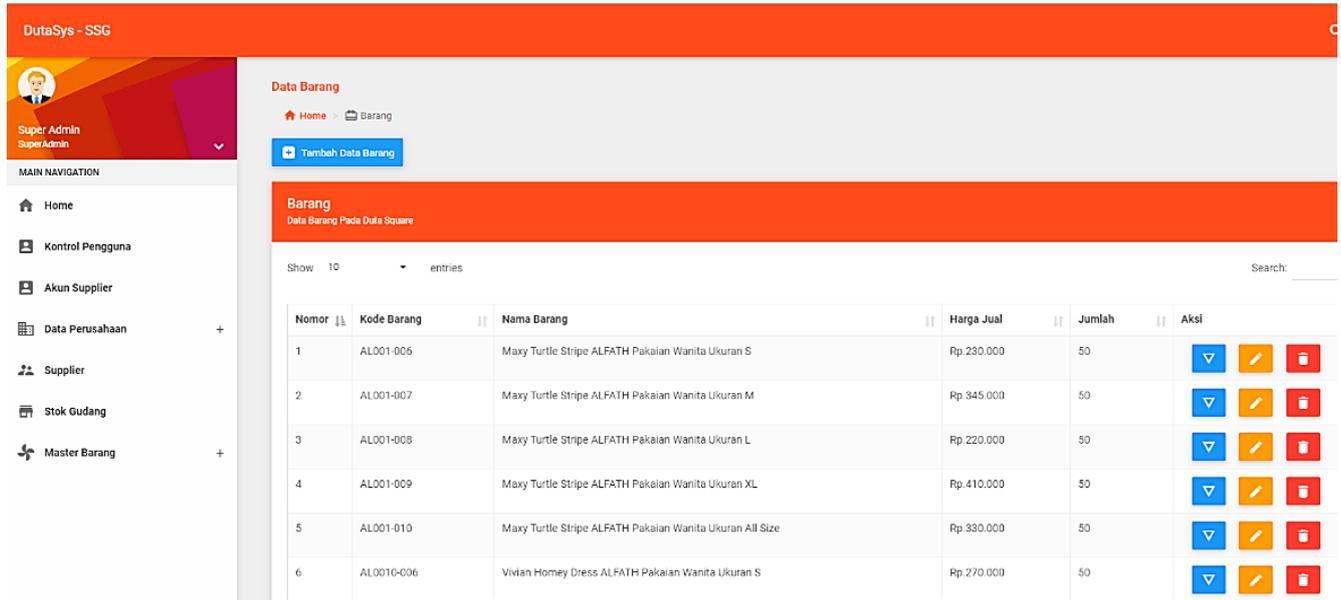


Gambar 5. Tampilan halaman kategori barang.

4. Tampilan Menu Barang

Menu barang diperlihatkan pada Gambar 6. Menu barang digunakan untuk mendata barang yang telah sampai di gudang dan sudah dilakukan pengecekan oleh pegawai perusahaan. Pada menu barang terdapat data berupa nomor, kode barang, nama barang, harga barang,

dan juga jumlah stok barang yang ada di gudang. Pada halaman ini juga admin dapat melakukan tambah data, mengubah, melihat *detail*, dan menghapus barang. Admin dapat melakukan pencarian data barang lewat *search* dengan memasukkan nama barang atau kode barang.

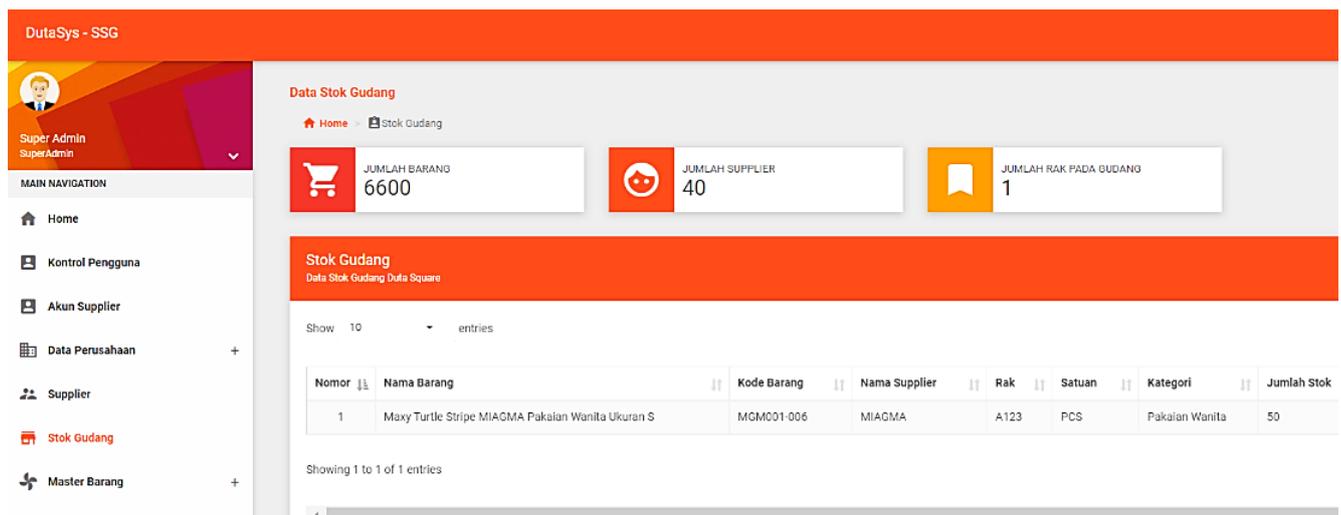


Gambar 6. Tampilan halaman barang.

5. Tampilan Menu Stok Gudang

Menu stok gudang diperlihatkan pada Gambar 7. Menu stok gudang digunakan untuk mengetahui keseluruhan stok barang yang ada di gudang. Pada menu stok gudang terdapat data berupa jumlah barang yang ada di gudang, jumlah *supplier*, jumlah rak barang yang ada di gudang, nama barang, kode barang, nama *supplier*,

nama satuan, nama kategori barang, dan jumlah stok barang. Pada halaman ini, admin hanya dapat melihat *detail* stok barang. Admin dapat melakukan pencarian data untuk melihat stok pada fitur *search* dengan memasukkan kata kunci berupa nama barang, kode barang, atau nama *supplier*.

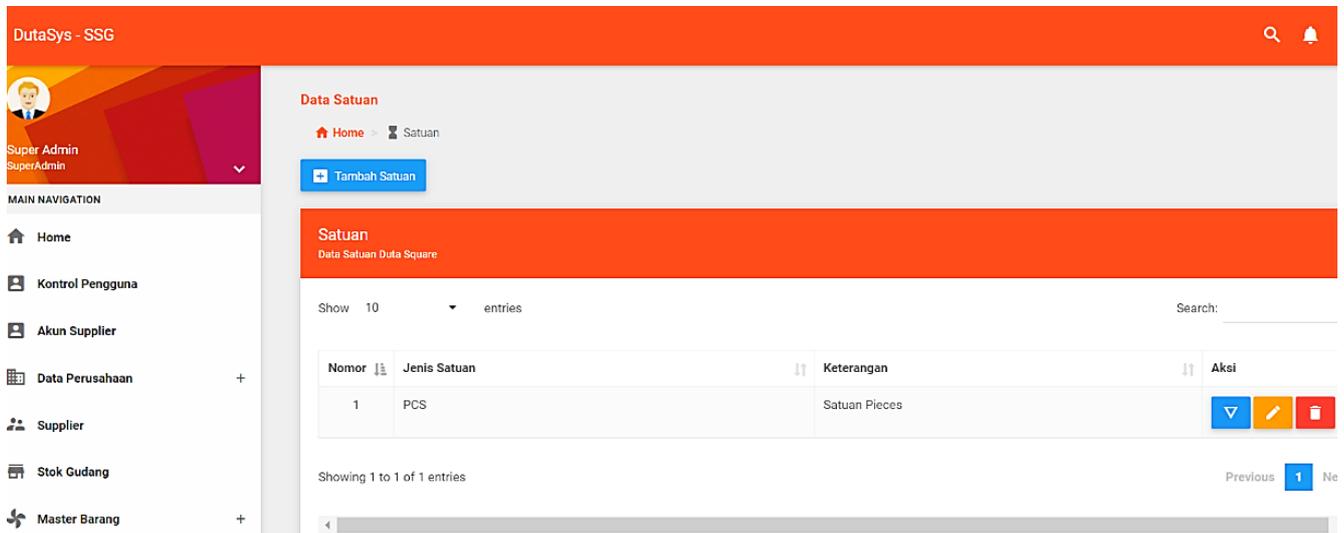


Gambar 7. Tampilan menu stok gudang.

6. Tampilan Menu Satuan

Tampilan menu satuan diperlihatkan pada Gambar 8. Menu satuan digunakan untuk membuat satuan barang misalnya satuan pcs, gr, dan ml. Pada menu satuan barang terdapat data berupa nomor, nama satuan, dan keterangan tentang satuan barang yang diinputkan.

Pada halaman ini, admin dapat melakukan tambah data satuan barang, mengubah satuan barang, melihat *detail* satuan barang, dan menghapus satuan barang. Admin dapat melakukan pencarian data satuan barang lewat fitur *search* dengan memasukkan kata kunci berupa nama satuan.

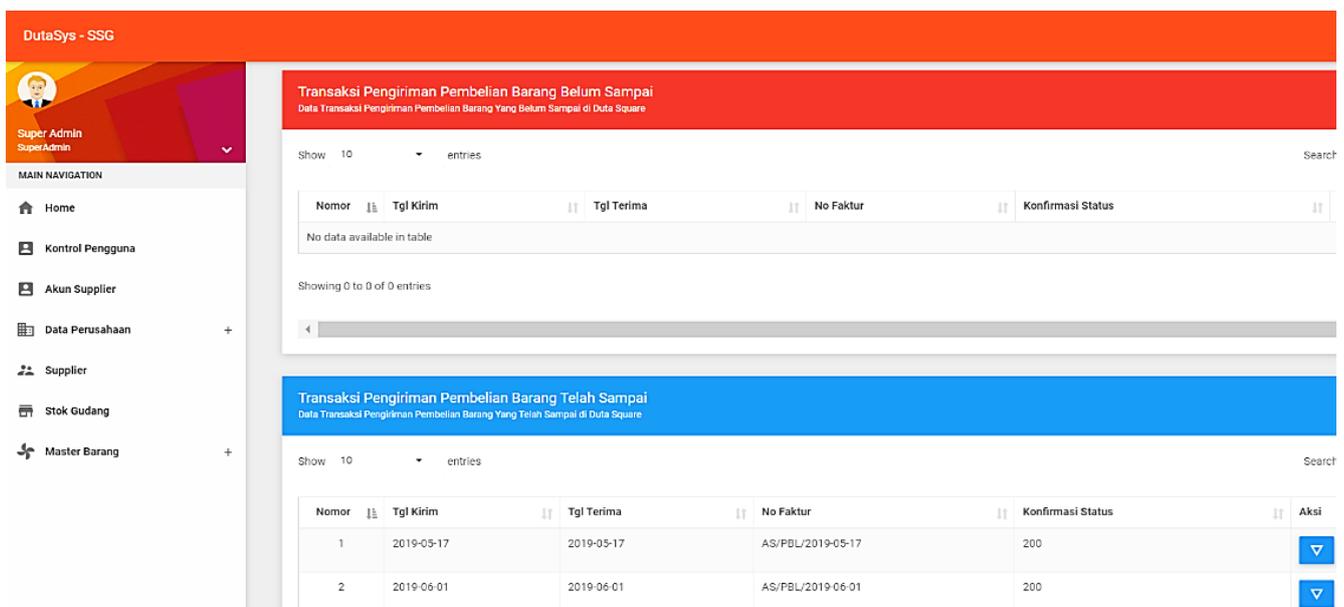


Gambar 8. Tampilan halaman menu satuan.

7. Tampilan Menu Transaksi Pengiriman Pembelian

Tampilan transaksi pengiriman diperlihatkan pada Gambar 10. Menu ini dapat dikatakan sebagai laporan pembelian barang dari *supplier* karena dilakukan pendataan pembelian barang dari *supplier*. Pada menu transaksi terdapat transaksi pembelian barang yang belum sampai di perusahaan dan pembelian barang yang telah sampai di perusahaan.

Barang yang sampai di perusahaan akan memiliki faktur sebagai data pembelian. Pada data transaksi tersebut terdapat data tanggal kirim, tanggal terima, nomor faktur, dan konfirmasi status. Pada halaman ini, admin dapat melakukan tambah data transaksi, mengubah, melihat *detail*, dan menghapus transaksi. Admin dapat melakukan pencarian data transaksi lewat *search* dengan memasukkan tanggal kirim atau nomor faktur.

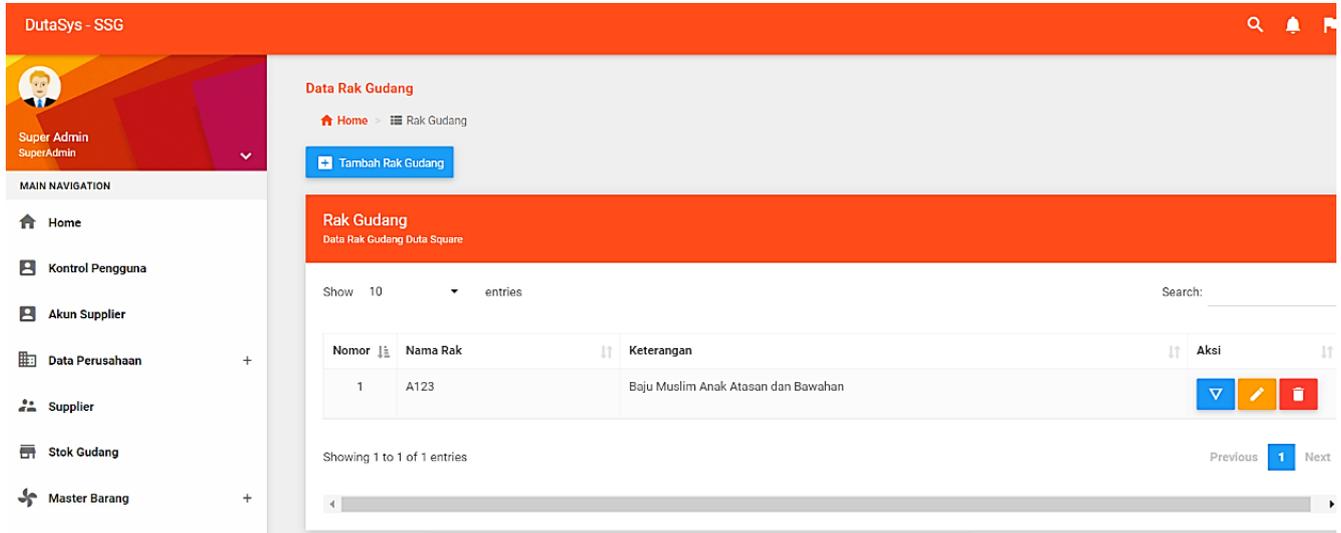


Gambar 9. Tampilan halaman menu transaksi pengiriman pembelian barang.

8. Tampilan Menu Rak Gudang

Tampilan rak gudang diperlihatkan pada Gambar 9. Menu rak gudang digunakan untuk membuat rak yang nantinya akan di isi daftar barang. Pada menu rak gudang terdapat data berupa nama rak gudang, dan keterangan tentang rak gudang contohnya yaitu kode A1

adalah kode rak untuk baju muslim anak. Pada halaman rak, admin dapat melakukan tambah data rak barang, mengubah rak barang, melihat *detail* rak barang, dan menghapus rak barang. Admin dapat melakukan pencarian data satuan barang lewat *search* dengan memasukkan kata kunci berupa nama rak.

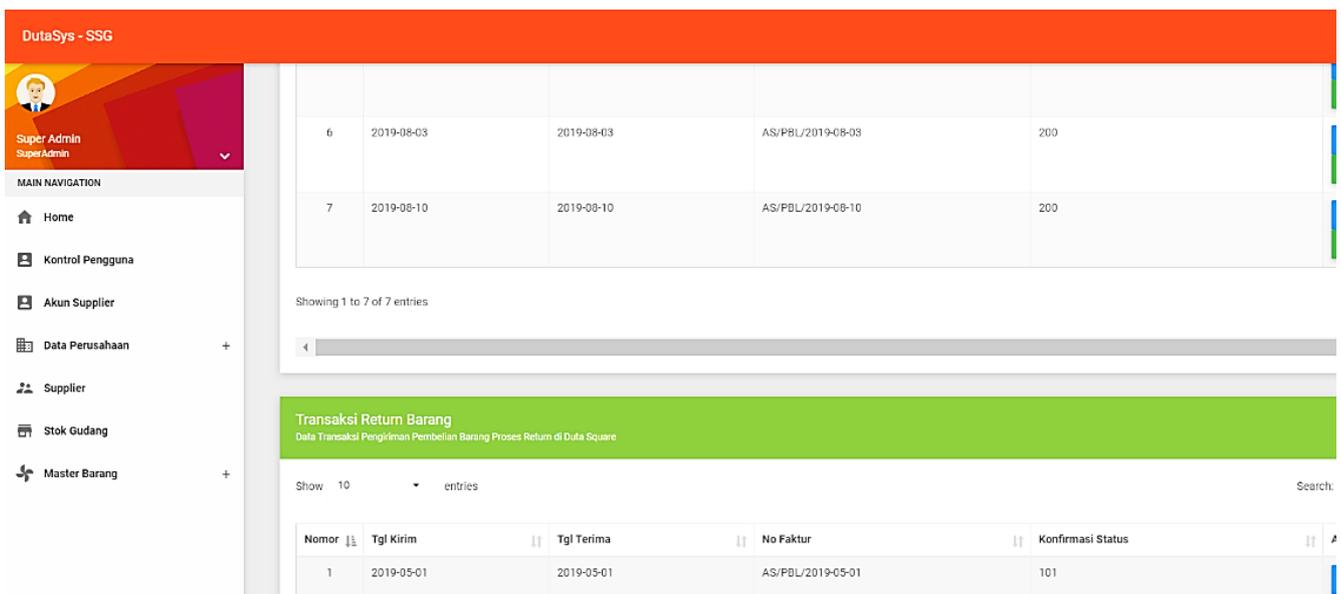


Gambar 100. Tampilan halaman menu *return* barang.

9. Tampilan Menu *Return* Barang

Tampilan *return* barang diperlihatkan pada Gambar 11. Menu *return* digunakan untuk mendata pengembalian barang (*return*) ke *supplier* ketika terdapat barang yang rusak/*reject*. Pada menu *return* barang terdapat data berupa tanggal kirim, tanggal terima, nomor faktur, dan konfirmasi status.

Pada halaman ini, admin dapat melakukan tambah data *return* barang dan melihat *detail return* barang. Admin dapat melakukan pencarian data *return* lewat *search* dengan memasukkan tanggal kirim atau nomor faktur. Ketika barang yang di-*return* sampai admin memilih tombol konfirmasi barang sampai.



Gambar 9. Tampilan halaman menu *return* barang.

10. Tampilan Menu Notifikasi

Tampilan notifikasi diperlihatkan pada Gambar 12. Menu notifikasi hanya dapat diakses oleh *role supplier*. Menu notifikasi digunakan untuk memberitahukan kepada *supplier* saat barang mereka hampir habis di gudang atau mencapai batas minimal. Fitur ini dibuat sebagai implementasi dari *supplier relationship management*, karena sebelumnya banyak *supplier* yang mengirimkan barang mereka ke perusahaan dengan inisiatif sendiri sehingga terjadi penumpukan stok di gudang.

Dengan adanya fitur ini diharapkan dapat meminimalisir *miss communication* antara *supplier* dan *owner* serta dapat membangun hubungan yang baik antara *supplier* dan perusahaan. Pada menu notifikasi terdapat pemberitahuan jumlah barang yang stoknya hampir habis. Pada halaman tersebut juga terdapat nama barang yang jumlahnya hampir habis di gudang dan dapat ditawarkan kembali kepada *owner* untuk dilakukan *restock* barang atau tidak. *Supplier* hanya dapat melihat tampilan halaman notifikasi tanpa bisa menambah, mengubah, atau menghapus data notifikasi.

The screenshot shows the 'Barang Yang Dapat Ditawarkan Supplier KARTINA' page. It features a navigation sidebar on the left with options like Home, Barang, and Notifikasi Barang. The main content area has a red header and a table listing items. A notification banner is present in the top right corner.

Nomor	Kode Barang	Nama Barang	Jumlah
1	KRT010013-018	Ananda Hijab KARTINA Hijab Khimar	45
2	KRT010026-013	Baju Anak Adelia KARTINA Pakaian Anak Ukuran L	45
3	KRT010019-001	Koko Rayon Bangkok KARTINA Pakaian Pria Ukuran S	44

Gambar 12. Tampilan halaman menu notifikasi.

11. Tampilan Menu Barang Tunda

Tampilan menu barang tunda diperlihatkan pada Gambar 13. Selain fitur notifikasi, fitur ini juga salah satu implementasi dari *supplier relationship management*. Tampilan menu barang tunda digunakan untuk memberitahukan kepada *supplier* barang yang tidak perlu dikirim ulang ke perusahaan. Data barang tersebut masuk kedalam barang tunda karena termasuk

barang yang tidak diminati pembeli. *Supplier* yang mengetahui daftar barang tunda tersebut, tidak perlu melakukan pengiriman barang yang ada di daftar barang tunda sehingga dapat menghemat biaya pengiriman atau biaya produksi yang dikeluarkan. *Supplier* hanya dapat melihat tampilan tanpa bisa menambah, mengubah, atau menghapus data.

The screenshot shows the 'Barang Supplier Ditunda' page. It features a navigation sidebar on the left with options like Home, Barang, and Barang Tunda. The main content area has a dark header and a table listing items. A notification banner is present in the top right corner.

Nomor	Kode Barang	Nama Barang
1	KRT01005-009	Luxine Maxy Dress KARTINA Pakaian Wanita Ukuran XL
2	KRT01007-008	Locca Set Dress KARTINA Pakaian Wanita Ukuran L

Gambar 13. Tampilan halaman menu barang tunda.

C. Pengujian Sistem

1. Black Box Testing

Black-box testing adalah metode pengujian perangkat lunak yang tes fungsionalitas dari aplikasi atau sistem. Metode *Black-box* berfokus pada keperluan fungsional yang dibutuhkan oleh *software*. *Black-Box testing* merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, *tester* dapat mendefinisikan kumpulan kondisi input dan melakukan pengujian pada spesifikasi fungsional program [17]. Pengujian ini menguji dua *role user* yang ada pada sistem yaitu *admin* dan *supplier*. Hasil dari pengujian yang telah dilakukan yaitu sistem telah berhasil memenuhi kebutuhan pengguna dan dapat mengidentifikasi kesalahan dengan menampilkan pesan *error*.

Pengujian untuk peran/*role admin* diperlihatkan pada Tabel I. Tabel I menunjukkan pengujian sistem pada aktor *admin* telah berfungsi sesuai keinginan pengguna dari *user stories*. Fitur-fitur yang diuji pada bagian aktor admin gudang yaitu pendataan *supplier*, pendataan barang, pendataan transaksi pembelian barang dari *supplier* dan pendataan *return* barang pembelian dari *supplier*.

TABEL I
PENGUJIAN SISTEM PERAN ADMIN

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang diharapkan	Hasil yang didapatkan	Validasi
1.	Halaman Menu Stok Gudang	Pengujian menampilkan halaman stok gudang.	Klik pada menu stok gudang.	Menampilkan informasi stok barang.	Sistem dapat menampilkan informasi stok.	VALID
2.	Halaman Menu Supplier	Pengujian menampilkan halaman menu <i>supplier</i>	Klik pada menu <i>supplier</i>	Menampilkan data semua <i>supplier</i> dan tombol aksi.	Sistem dapat menampilkan semua <i>supplier</i> dan aksi.	VALID
		Pengujian pada tombol "Tambah Data" yang ada di halaman menu <i>Supplier</i>	Klik pada tombol tambah data <i>Supplier</i>	Menampilkan form pengisian <i>Supplier</i> baru dan dapat menyimpan data.	Sistem dapat menampilkan form pengisian dan dapat menyimpan data	VALID

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang diharapkan	Hasil yang didapatkan	Validasi
		Pengujian pada Tombol aksi berupa ubah, detail, dan hapus	Klik tombol ubah/hapus/detail pada data <i>Supplier</i>	Menampilkan data <i>Supplier</i> yang akan diubah/dihapus/melihat detail	Sistem dapat menampilkan data <i>Supplier</i> yang diubah/dihapus/melihat detail	VALID
3.	Halaman Sub Menu Barang	Pengujian menampilkan halaman sub menu barang.	Klik pada menu Master Barang, lalu klik sub menu barang.	Menampilkan halaman yang berisi data barang.	Sistem dapat menampilkan halaman yang berisi data barang.	VALID
		Pengujian pada tombol Tambah Data yang menu barang.	Klik menu barang, dan klik tombol "Tambah Data" pada barang.	Menampilkan form pengisian data barang baru.	Sistem dapat menampilkan form pengisian data barang baru.	VALID
		Pengujian pada tombol aksi berupa ubah, detail, dan hapus	Klik tombol ubah/hapus/detail pada data barang	Menampilkan data barang yang akan diubah/dihapus/melihat detail.	Sistem dapat menampilkan data barang yang diubah/dihapus/melihat detail.	VALID
4.	Halaman Sub Menu Rak	Pengujian menampilkan halaman menu rak dan data barang yang ada di rak.	Klik pada menu rak, dan pilih rak yang akan dilihat.	Menampilkan halaman yang berisi data barang pada rak yang dipilih.	Sistem dapat menampilkan halaman data barang pada rak yang dipilih.	VALID

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang diharapkan	Hasil yang didapatkan	Validasi
		Pengujian pada tombol “Tambah Rak”	Klik menu rak, dan klik tombol tambah rak.	Menampilkan form pengisian tambah rak.	Sistem dapat menampilkan form pengisian tambah rak.	VALID
		Pengujian pada tombol aksi berupa ubah, detail, dan hapus	Klik tombol ubah/hapus/detail pada data rak.	Mempilkan data yang akan diubah / dihapus/ melihat detail	Sistem dapat menampilkan data rak yang akan diubah / dihapus/ lihat detail	VALID
5.	Halaman Sub Menu Satuan	Pengujian menampilkan halaman satuan	Klik pada menu satuan.	Mempilkan halaman data satuan.	Sistem dapat menampilkan halaman data satuan.	VALID
		Pengujian pada tombol “Tambah Satuan” yang ada di halaman sub menu satuan.	Klik menu satuan, dan klik tombol tambah satuan.	Mempilkan form pengisian untuk menambah satuan.	Sistem dapat menampilkan form pengisian untuk menambah satuan.	VALID
		Pengujian pada tombol ubah/hapus/detail pada data satuan.	Klik tombol ubah/hapus/detail pada data satuan.	Mempilkan data yang akan diubah / dihapus/ melihat detail	Sistem dapat menampilkan data yang akan diubah / dihapus/ lihat detail	VALID

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang diharapkan	Hasil yang didapatkan	Validasi
6.	Halaman Menu Kategori	Pengujian menampilkan halaman sub menu kategori.	Klik pada menu kategori	Mempilkan halaman data kategori.	Sistem dapat menampilkan halaman data kategori.	VALID
		Pengujian pada tombol “Tambah kategori” di halaman sub menu kategori.	Klik menu kategori, dan klik tombol tambah kategori.	Mempilkan form pengisianambah kategori.	Sistem dapat menampilkan form pengisian kategori	VALID
		Pengujian pada tombol Aksi berupa ubah, detail, dan hapus.	Klik tombol ubah/hapus/detail pada data kategori	Mempilkan data yang akan diubah / dihapus/ melihat detail.	Sistem dapat menampilkan data yang akan diubah / dihapus/ melihat detail.	VALID
7.	Halaman Menu Pengiriman Pembelian Barang	Pengujian menampilkan halaman transaksi pembelian barang	Klik pada halaman transaksi pembelian barang	Mempilkan halaman yang berisi data pengiriman pembelian barang	Sistem dapat menampilkan halaman yang berisi data pengiriman pembelian barang	VALID
		Pengujian pada tombol “Tambah Data” di halaman	Klik pada transaksi pengiriman dan	Mempilkan form pengisian data pengiriman	Sistem dapat menampilkan form pengisian	VALID

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang diharapkan	Hasil yang didapatkan	Validasi
		pengiriman pembelian barang	klik tombol "Tambah Data"	man baru	an data baru	
		Pengujian pada tombol Aksi berupa ubah, detail, dan hapus	Klik tombol ubah/hapus/detail pada data pengirim.	Mena mpilkan data yang akan diubah / dihapus/ melihat detail	Sistem dapat mena mpilkan data yang akan diubah / dihapus/ lihat detail	VALID
8.	Halaman Return	Pengujian menampilkan halaman Return.	Klik pada menu Transaksi dan klik Return	Mena mpilkan halaman berisi data Return	Sistem dapat mena mpilkan halaman Return	VALID
		Pengujian pada tombol "Tambah Data" Return.	Klik menu Return, dan klik tombol "Tambah Data".	Mena mpilkan form pengisian data Return baru.	Sistem dapat mena mpilkan form pengisian data baru	VALID
		Pengujian pada tombol Aksi berupa ubah, detail, dan hapus	Klik tombol ubah/hapus/detail pada data Return	Mena mpilkan data yang akan diubah / dihapus/ melihat detail	Sistem dapat mena mpilkan data yang akan diubah / dihapus/ melihat detail	VALID

Dari Tabel I, dapat dilihat fitur telah diuji dan didapatkan hasil yang sebenarnya sudah sesuai dengan analisis kebutuhan sistem dan *user stories owner*. Pengujian ini telah tervalidasi sesuai dengan hasil sebenarnya. Fitur-fitur yang diuji yaitu fitur stok gudang, barang, rak gudang, kategori, satuan, *supplier*, *return* barang, dan transaksi pengiriman pembelian. Setiap fitur diuji untuk menampilkan halaman saat admin melakukan perintah/ klik pada fitur tersebut. Pengujian juga dilakukan pada setiap tombol aksi dan hasil yang didapat tombol aksi dapat berjalan sesuai

perintah. Pada pengujian ini juga sistem dapat memunculkan pesan *error* saat muncul kesalahan.

Pengujian sistem untuk peran/*role Supplier* diperlihatkan pada Tabel II. Tabel II menunjukkan pengujian sistem pada aktor *Supplier* telah berfungsi sesuai keinginan pengguna. Fitur-fitur yang di uji pada bagian aktor *Supplier* yaitu fitur barang tunda dan notifikasi.

TABEL II
PENGUJIAN SISTEM PERAN SUPPLIER

No	Kelas Uji	Daftar Pengujian	Skenario Uji	Hasil yang diharapkan	Hasil yang didapatkan	Validasi
1.	Halaman Menu Barang	Pengujian menampilkan halaman barang	Klik pada menu barang.	Menampilkan halaman <i>supplier login</i> .	Menampilkan halaman <i>supplier login</i> .	VALID
2.	Halaman Notifikasi	Pengujian menampilkan halaman notifikasi	Klik pada notifikasi	Menampilkan informasi untuk <i>supplier</i>	Sistem menampilkan informasi untuk <i>supplier</i>	VALID
3.	Halaman Menu Transaksi dan Return	Pengujian menampilkan halaman transaksi dan <i>return</i>	Klik pada menu transaksi dan <i>return</i>	Menampilkan halaman berisi data transaksi dan <i>return supplier</i>	Sistem menampilkan halaman transaksi dan <i>return supplier</i>	VALID
4.	Halaman Menu Barang Tunda	Pengujian menampilkan halaman barang tunda	Klik pada menu barang tunda	Menampilkan informasi barang yang ditunda	Sistem dapat menampilkan informasi barang tunda	VALID

Dari Tabel II, dapat dilihat fitur yang dibuat telah diuji dan didapatkan hasil yang sebenarnya sudah sesuai dengan analisis kebutuhan sistem dan *user stories owner*. Pengujian ini telah tervalidasi fitur yang diuji sesuai dengan hasil sebenarnya. Fitur-fitur yang diuji diantaranya yaitu fitur menu barang, fitur notifikasi, dan fitur barang tunda. Masing-masing fitur diuji untuk menampilkan halaman saat admin melakukan perintah/ klik pada fitur tersebut. Pengujian juga dilakukan pada tombol aksi disetiap fitur dan hasil yang didapat tombol aksi dapat berjalan sesuai perintah.

2. Skala Likert

Pengujian selanjutnya dilakukan dengan menggunakan kuesioner untuk mengetahui penilaian terhadap sistem yang sudah dibuat. Hasil dari kuesioner tersebut, selanjutnya dilakukan perhitungan bobot poin menggunakan Skala Likert. Metode Skala Likert merupakan metode yang digunakan untuk mengukur tingkat kepuasan pengguna menggunakan skala

pengukuran yang dikembangkan oleh Likert pada Tahun 1932. Skala likert memiliki empat atau lebih butir-butir pertanyaan yang dikombinasikan sehingga membentuk sebuah skor atau nilai yang mempersentasikan sifat individu, misalkan pengetahuan, sikap dan perilaku. Skala likert dapat juga dikatan sebagai skala psikometrik yang umum digunakan dalam kuesioner dan merupakan skala yang paling banyak digunakan untuk penelitian. Biasanya disediakan lima pilihan skala dengan format seperti: Sangat setuju, Setuju, Cukup setuju, Kurang setuju, dan Tidak setuju [18].

Pengujian ini melibatkan 30 orang responden yang dibagi menjadi 2 bagian yaitu pegawai perusahaan dan *supplier* yang masing-masing berjumlah 15 orang. Pengujian ini dilakukan dengan memerintahkan pegawai dan *supplier* yang ada di perusahaan tersebut untuk menggunakan sistem yang sudah jadi dan mengisi kuesioner yang terdiri dari 8 butir pernyataan. Setiap pernyataan memiliki 5 penilaian. Pilihan penilaian terdiri dari Sangat Setuju (SS) dengan bobot 5, Setuju (S) dengan bobot 4, Cukup Setuju (CS) dengan bobot 3, Kurang Setuju (KS) dengan bobot 2, dan Tidak Setuju (TS) dengan bobot 1. Tujuan pembagian tersebut karena adanya perbedaan fitur di sistem pada *user admin* dan *user supplier*. Hasil analisis lembar kuesioner penilaian akan menggunakan interpretasi presentase menurut Diani dkk yaitu saat tingkat hasil rata-rata pencapaian $> 80\%$ memiliki kualifikasi Sangat Baik, $60\% < X \leq 80\%$ memiliki kualifikasi Baik, $40\% < X \leq 60\%$ memiliki kualifikasi Ragu-Ragu, $20\% < X \leq 40\%$ memiliki kualifikasi Kurang Baik, dan $0\% < X \leq 20\%$ memiliki kualifikasi Tidak Baik [19].

Tabel III merupakan hasil dari penilaian pegawai perusahaan terhadap kuesioner yang terdiri dari 8 butir pernyataan dengan 5 kategori penilaian. Hasil presentase yang didapatkan setelah dihitung rata-rata dengan skala *likert* yaitu 79,83% yang artinya pegawai di perusahaan tersebut menilai sistem pendataan barang dan *supplier* ini sudah baik.

TABEL III
HASIL PENILAIAN RESPONDEN PEGAWAI TERHADAP KUESIONER

No	Pernyataan	Total	Index %
1	Sistem dapat menyajikan informasi yang jelas dan mudah dimengerti	59	78,67%
2	Tata letak/ komposisi/ fitur pada sistem mudah untuk dipahami dan digunakan (<i>easy to use</i>)	61	81,33%
3	Tombol yang ada pada sistem dapat berfungsi dengan baik	63	84,00%
4	Sistem dapat memunculkan pesan konfirmasi saat data akan dihapus atau	59	78,67%

	jika terjadi <i>error</i>		
5	Sistem dapat menyimpan data yang telah diinputkan dan proses input data yang mudah digunakan	56	74,67%
6	Sistem mudah untuk diakses (tidak terjadi <i>crash/not responding</i>)	56	74,67%
7	Penilaian anda terhadap sistem dapat membantu anda mempercepat mendata <i>supplier</i> , barang, dan pembelian	66	88,00%
8	Penilaian anda terhadap sistem dapat membantu anda dalam pengambilan keputusan	59	78,67%
Total Rata-Rata (%)			79,83%

Tabel IV merupakan hasil dari penilaian *supplier* yang bekerja sama dengan perusahaan terhadap kuesioner yang terdiri dari 8 butir pernyataan dengan 5 kategori penilaian. Hasil presentase yang didapatkan setelah dihitung rata-rata dengan skala Likert yaitu 77,17% yang artinya pegawai perusahaan tersebut menilai sistem pendataan barang dan *supplier* ini sudah baik.

TABEL IV
HASIL PENILAIAN RESPONDEN *SUPPLIER* TERHADAP KUESIONER

No	Pernyataan	Total	Index %
1	Sistem dapat menyajikan informasi yang jelas dan mudah dimengerti	59	78,67%
2	Fitur notifikasi pada sistem dapat menyediakan informasi terbaru (<i>up to date</i>) secara <i>real time</i>	58	77,33%
3	Sistem (tata letak/ komposisi/ fitur) mudah untuk dipahami dan digunakan (<i>easy to use</i>)	55	73,33%
4	Tombol yang ada pada sistem dapat berfungsi dengan baik	61	81,33%
5	Sistem dapat memunculkan pesan konfirmasi saat terjadi <i>error</i>	57	76,00%
6	Sistem mudah untuk diakses (tidak terjadi <i>crash/not responding</i>)	59	78,67%
7	Penilaian anda terhadap sistem, memudahkan anda melakukan pengiriman barang ke perusahaan karena fitur barang tunda	61	81,33%
8	Penilaian anda terhadap sistem dapat membantu anda dalam mengetahui sisa stok barang yang ada di perusahaan lewat fitur notifikasi	53	70,67%
Total Rata-Rata (%)			77,17%

V. KESIMPULAN

Adapun kesimpulan yang didapat dari hasil penelitian ini yaitu menghasilkan sebuah sistem informasi untuk

pendataan *supplier* dan barang di salah satu perusahaan yang ada di Bandar Lampung berbasis *web* dengan menggunakan *framework* Laravel. Sistem berhasil melakukan pendataan *supplier*, barang, transaksi pengiriman, *return* barang, dan juga dapat memberikan notifikasi kepada *supplier*. Pendataan barang termasuk didalamnya yaitu kode barang, satuan barang, posisi rak barang, dan kategori barang.

Metode yang digunakan dalam membangun sistem informasi ini menggunakan metode *Extreme Programming* (XP) yang mana memiliki empat tahap yaitu *planning*, *design*, *coding*, dan *testing*. Pengujian yang dilakukan dalam penelitian ini menggunakan *Black Box Testing* untuk menguji fungsionalitas sistem. Hasil dari pengujian tersebut sistem dapat berfungsi dengan baik sesuai *request* dan tidak terdapat *error*.

Hasil kuesioner yang dilakukan pada penelitian ini yaitu, pada bagian pegawai perusahaan sebanyak 15 responden dengan masing-masing menjawab 8 butir pernyataan menghasilkan presentase rata-rata nilai yang dihitung dengan skala Likert yaitu 79,83% yang artinya pegawai perusahaan tersebut menilai sistem yang dibuat sudah baik. Sedangkan, pada bagian *supplier* yang bekerja sama dengan perusahaan diambil sebanyak 15 responden dengan masing-masing menjawab 8 butir pernyataan menghasilkan presentase rata-rata nilai yang dihitung dengan skala Likert yaitu 77,17% yang artinya *supplier* di perusahaan tersebut menilai sistem yang dibuat sudah baik.

Berdasarkan hasil dari penelitian ini, terdapat beberapa saran untuk pengembangan selanjutnya yaitu menambahkan fitur *life chat* antara *owner* dan *supplier* agar lebih mudah berdiskusi saat melakukan *restock* barang dan menambahkan fitur laporan untuk data *supplier*, data barang, dan data transaksi pembelian barang.

DAFTAR PUSTAKA

- [1] S. Kodu, "Harga, kualitas produk, dan kualitas pelayanan pengaruhnya terhadap keputusan pembelian mobil toyota avanza," *Jurnal Riset Ekonomi, Manajemen, Bisnis dan Akuntansi*, vol. 1, pp. 1252-1259, September 2013.
- [2] C. Schuh, M. F. Strohmer, S. Easton, M. D. Hales, & A. Triplat, *Supplier relationship Management: How to Maximize Vendor Value and Opportunity*, New York City: Apress 2014.
- [3] M. Soediro, "Analisis pengaruh *top management commitment* terhadap *supplier performance* dan *supplier relationship management* pada hotel Budget di Surabaya," *Jurnal Ilmiah Manajemen dan Akuntansi*, vol. 4, pp. 1-12, Juni 2017.
- [4] H. Agusvianto, "Sistem informasi inventori gudang untuk mengontrol persediaan barang pada gudang studi kasus : PT. Alaisys Sidoarjo," *Journal Information Engineering and Educational Technology*, vol. 1, pp. 40-46, Agustus 2017.
- [5] P. H. Sutanto, "Perancangan sistem stok barang di *warehouse* berbasis *web*," *Jurnal Sistem Komputer Musirawas*, vol. 4, pp. 9-18, Juni 2019.
- [6] R. Mantala, "Perancangan dan implementasi sistem informasi persediaan bahan baku dan suku cadang di PT. Triteguh Manunggal Sejati Banjarbaru," *Jurnal POSITIF*, vol. 2, pp. 16-20, November 2016.
- [7] B. Nugroho, *Program Toko dengan Visual Basic dan MySQL*. Yogyakarta, Indonesia: Gava Media 2014.
- [8] J. O'Brien, *Supplier Relationship Management: Unlocking the Hidden Value in Your Supply Base*. London: Kogan Page 2014.
- [9] Deloitte AG, "Supplier Relationship Management (SRM): Identifying and maximizing the value of strategic supplier partnering". London: The Creative Studio at Deloitte, 2015.
- [10] R. V. Hoek, *Supplier Relationship Management "How Key Suppliers Drive Your Company's Competitive Advantage"*. Netherlands: Pricewaterhouse Coopers Accountants, 2013.
- [11] U. Pauziah, "Perancangan sistem pendataan penduduk pada kelurahan Cililitan Jakarta Timur berbasis delphi," *Faktor Exacta*, vol. 6, pp. 189-199, Maret 2013.
- [12] Y. I. Chandra, "Perancangan aplikasi resep masakan tradisional indonesia menggunakan pendekatan *agile process* dengan model *extreme programming* berbasis android," *Proceeding Seminar Nasional APTIKOM*, 2016, vol. 1, pp.607-614.
- [13] R. V. Anand & M. Dinakaran, "Popular agile methods in software development: review and analysis," *Int. J. Appl. Eng. Res. ISSN*, vol. 11, no. 5, pp. 973-4562, 2016.
- [14] A. Fatoni & D. Dwi, "Rancang bangun sistem *extreme programming* sebagai metodologi pengembangan sistem," *Jurnal PROSISKO*, vol. 3, pp.17-20, Maret 2016.
- [15] M. A. Prayoga, Y. S. Siregar, S. Khairani, "Perancangan aplikasi pemilihan ketua organisasi di fakultas teknik dan komputer Universitas Harapan Medan menggunakan metode *extreme programming* berbasis android," *Proceeding SNASTIKOM*, 2019, pp.211-218.
- [16] I.G.N.Suryantara, *Merancang Aplikasi dengan Metodologi Extreme Programmings*. Jakarta, Indonesia: PT Elex Media Komputindo, 2017.
- [17] T. Hidayat & M. Muttaqin, "Pengujian sistem informasi pendaftaran dan pembayaran wisuda online menggunakan black box testing dengan metode equivalence partitioning dan boundary value analysis," *Jurnal Teknik Informatika UNIS*, vol. 6, pp.25-29, Oktober 2018.
- [18] S.Syofian, T. Setyaningsih, N. Syamsiah, "Otomatisasi metode penelitian skala likert berbasis web," *Proceeding Seminar Nasional Sains dan Teknologi*, 2015, vol. 2, pp.1-8.
- [19] R. Diani, Yuberti, & S. Syafitri, "Uji *effect size* model pembelajaran *scramble* dengan media video terhadap hasil belajar fisika peserta didik kelas x MAN 1 Pesisir Barat," *Jurnal Ilmiah Pendidikan Fisika Al-BiRuNi*, vol. 5, pp.265-275, 2016.