APLIKASI *SMART WAREHOUSE* BERBASIS ANDROID

Eduardo Richard Eko[1)], Susmini Indriani Lestariningati[2)]
[1,2)]Jurusan Teknik Komputer, Fakultas Teknik dan Ilmu Komputer Universitas Komputer Indonesia
Jl. Dipatiukur No.112 Bandung 40132
[1)]erik_naxcl@yahoo.com, [2)]susmini.indriani@email.unikom.ac.id

Abstrak *– Gudang merupakan suatu area yang berfungsi untuk menyimpan barang untuk produksi. Barang produksi yang disimpan biasanya dalam jumlah yang banyak dan rentang waktu pendistribusian yang berbeda-beda baik secara internal maupun eksternal. Tidak jarang banyak masalah yang sering terjadi di pergudangan, masalah tersebut ialah pendataan barang yang lambat dan tidak akurat karena pendataan barang masih dilakukan secara manual yaitu menggunakan catatan yang ditulis dengan tangan. Maka oleh karena itu dibangunlah sistem Smart Warehouse berbasis android, di mana sistem ini menggunakan RFID reader untuk RFID reader membaca data yang ada pada RFID tag, lalu data tersebut akan terkirim melalui jalur komunikasi wifi ke sebuah Personal Computer. Data yang tersimpan di dalam PC dapat ditampilkan ke smartphone dengan menggunakan jalur komunikasi internet Pada aplikasi ini akan terhubung dengan jaringan internet, lalu akan mengambil data dari RFID, kemudian data akan ditampilkan pada aplikasi. Data yang ditampilkan ke user yaitu daftar barang masuk dan daftar barang keluar. Selain itu aplikasi yang dibangun dapat mengaktifkan saklar yang berfungsi untuk mengetahui keberadaan kategori barang dengan memanfaatkan bunyi buzzer.Dari hasil pengujian alpha dengan metode black box yang telah dilakukan dihasilkan aplikasi smart warehouse dapat menampilkan data barang masuk, keluar dan melakukan pencarian keberadaan kategori barang.*

Kata kunci *- Smart Warehouse, Android.*

1. PENDAHULUAN

Gudang merupakan fasilitas yang digunakan untuk lokasi penyaluran barang dari supplier (pemasok) sampai ke end user (pengguna). Salah satu tujuan dari gudang salah satunya ialah untuk menekan biaya transportasi dan produksi. Dewasa ini banyak sekali kasus-kasus atau masalah-masalah yang melibatkan gudang yang dimiliki oleh perusahaan-perusahaan. Salah satunya ialah durasi pendataan barang yang sangat lambat dan tidak akurat, hal ini disebabkan karena pendataan atau pengecekan data barang masih secara manual yaitu pembukuan dan tenaga kerja yang dibutuhkan sangat banyak.

Oleh karena itu dirancanglah sebuah aplikasi berbasis Android yang dapat menampilkan data barang yang sudah terintegrasi kedalam sistem Smart Warehouse dengan menggunakan RFID. RFID reader membaca data yang ada pada RFID tag, lalu data tersebut akan terkirim melalui jalur komunikasi wifi ke sebuah Personal Computer. Data yang tersimpan di dalam PC dapat ditampilkan ke *smartphone*. Selain itu, untuk mencari atau mengetahui keberadaan kategori barang dengan memanfaatkan bunyi dari buzzer. Sistem ini membantu para pekerja untuk pendaaan barang di pergudangan dan mengetahui letak kategori barang yang akan dicari di pergudangan.

2. TEORI PENUNJANG

Teori penunjang membahas tentang teori serta software yang akan digunakan pada sistem *Smart Warehouse* yang menggunakan jaringan lokal.

A. Smart Warehouse

Smart Warehouse adalah sistem gudang cerdas yang melakukan terjadinya keluar masuk barang disuatu pergudangan yang terintegrasi dengan Internet, sehingga memudahkan admin gudang dalam melakukan aktivitas pergudangan. *Smart Warehouse* dapat mengintegrasikan kegiatan pergudangan admin gudang secara efisien dan ekonomis [1].

B. Android

Android adalah sebuah sistem operasi untuk perangkat *mobile* berbasis linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan *platform* terbuka bagi pada pengembang untuk menciptakan aplikasi mereka. Awalnya, Google Inc. membeli Android Inc. yang merupakan pendatang baru yang membuat peranti lunak untuk ponsel/*smartphone*. Kemudian untuk mengembangkan Android, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia[2].

C. Pemrograman Berorientasi Objek

Secara spesifik, pengertian berorientasi objek berarti bahwa mengorganisasi perangkat lunak sebagai kumpulan objek tertentu yang memiliki struktur data dan perilakunya. Hal ini yang

membedakan dengan pemrograman konvensional dimana struktur data dan perilaku hanya berhubungan secara terpisah[3].

D. Unified Modeling Language (UML)

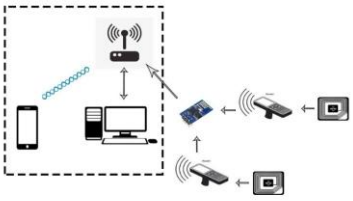
UML adalah bahasa grafis untuk mendokumentasi, menspesifikasikan, dan membangun sistem perangkat lunak. UML berorientasi objek, menerapkan banuak level abstraksi, tidak bergantung proses pengembangan, tidak bergantung bahasa dan teknologi, pemanduan beberapa notasi di beragam metodologi, usaha bersama dari banyak pihak, didukung oleh kakas-kakas yang diintegrasikan lewat XML[4].

E. Bahasa Pemrograman Java

Saat ini Java merupakan salah satu bahasa pemrograman paling populer untuk segala hal mulai dari permainan pemrograman untuk menciptakan aplikasi *mission-critical.* Tujuannya adalah untuk mempermudah dan menjadi platform independen alternatif menggunakan C++ . Java adalah program yang sangat populer di lingkungan pendidikan saat ini, hampir di seluruh kampus memiliki satuan ajar pemrograman ini. *Java Virtual Machine* (JVM) atau *Java Runtime Environment* (JRE) merupakan fitur Java yang membuatnya dapat berjalan pada semua *platform* sistem operasi[5].

3. PERANCANGAN SISTEM

Langkah awal dari perancangan sistem adalah analisis dan penentuan kebutuhan sistem. Pada langkah ini menentukan kebutuhan apa saja yang harus dipenuhi.



Gambar 1 Gambaran Umum Sistem

Dari gambar 1 dijelaskan bahwa sistem middleware akan menerima hasil pembacaan dua buah RFID reader yang sama pada RFID tag akan dikirimkan ke database melalui modul Wi-Fi dengan router. Kemudian setelah data diterima dan disimpan pada database, server akan mengirimkan data tersebut jika ada perangkat lain seperti *smartphone* yang ingin mengakses data RFID tag.

A. Analisis Kebutuhan Sistem

Aplikasi yang dibangun, diharapkan dapat menampilkan informasi data barang dari proses pengecekkan barang yang terintegrasi dengan sistem *smart warehouse* dan mencari keberadaan rak kategori barang.

B. Analisis Kebutuhan Non Fungsional

Analsisis kebutuhan non-fungsional dibagi menjadi dua, yaitu analisis perangkat keras dan analisis perangkat lunak. Analisis perangkat keras bertujuan untuk memudahkan proses perancangan dan implementasi dalam pembangunan sistem *smart warehouse*.

a. Analisis Kebutuhan Perangkat Keras

Analisis perangkat keras bertujuan untuk memudahkan proses perancangan dan implementasi dalam pembangunan sistem Smart Warehouse berbasis Android.

Adapun kebutuhan minimum perangkat keras mobile yang dibutuhkan oleh pengguna untuk menjalankan aplikasi dapat dilihat pada tabel dibawah ini.

Tabel 1 Kebutuhan Minimum Perangkat Keras Pengguna

No	Perangkat Keras	Spesifikasi Minimum
1	Prosesor *Smartphone*	1,2GHz
2	RAM	2GB
3	SD Card	16GB

b. Analisis Kebutuhan Perangkat Lunak

Sistem yang dibangun pada aplikasi *Smart Warehouse* membutuhkan perangkat lunak pendukung sehingga sistem tersebut dapat berjalan dengan baik. Adapun perangkat lunak minimum yang dibutuhkan di dalam sistem yang dibangun dapat dilihat pada Tabel 2 :

Tabel 2 Kebutuhan Minimum Perangkat Keras Pengguna

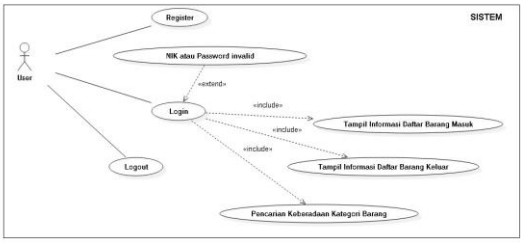
No	Perangkat Lunak	Spesifikasi Minimum
1	Sistem Operasi *Smartphone*	Android Jelly Bean 4.1
2	Sistem Operasi Laptop/PC	Windows 8.1 Pro 64 bit
3	Editor Aplikasi	Android Studio 2.2.3

C. Analisis Kebutuhan Fungsional

Dalam merancang sistem ini menggunakan UML (*Unified Modelling Language*). Pada dasarnya aplikasi menampilkan dua macam data barang, daftar barang masuk dan daftar barang keluar.

a. Use Case Diagram

Use case diagram merupakan suatu diagram yang menggambarkan fungsionalitas yang dimiliki oleh suatu sistem beserta aktor-aktor yang terlibat di dalamnya. Gambar di bawah ini merupakan *use case* diagram sistem *smart warehouse*.

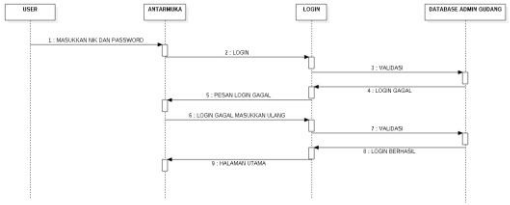


Gambar 2 Use Case Diagram Sistem Smart Warehouse

b. Sequence Diagram

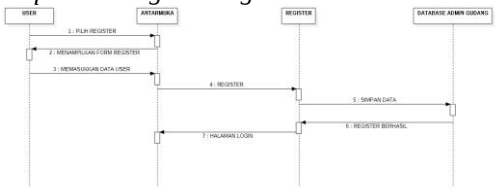
Sequence diagram menggambarkan *behavior* objek pada *use case* dengan mendeskripsikan waktu hidup dan pesan yang dikirimkan dan diterima antar objek. *Sequence diagram* pada sistem ini sebagai berikut.

1. *Sequence Diagram Login*



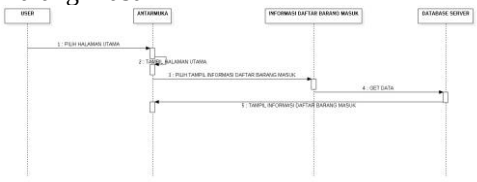
Gambar 3 Sequence Diagram Login

2. *Sequence Diagram Register*



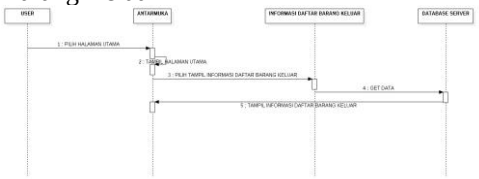
Gambar 4 Sequence Diagram Register

3. *Sequence Diagram* Tampil Informasi Daftar Barang Masuk



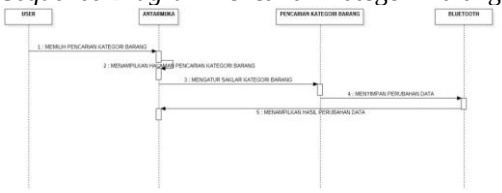
Gambar 5 Sequence Diagram Tampil Informasi Daftar Barang Masuk

4. *Sequence Diagram* Tampil Informasi Daftar Barang Keluar



Gambar 6 Sequence Diagram Tampil Informasi Daftar Barang Keluar

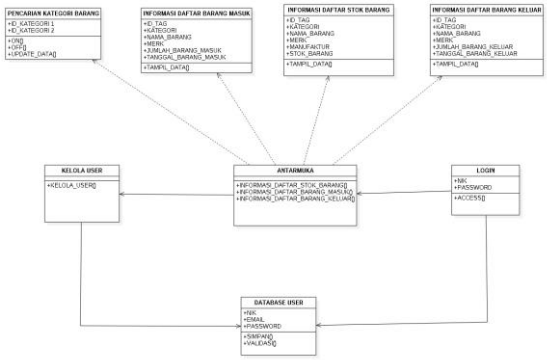
5. *Sequence Diagram* Pencarian Kategori Barang



Gambar 7 Sequence Diagram Pencarian Kategori Barang

c. Class Diagram

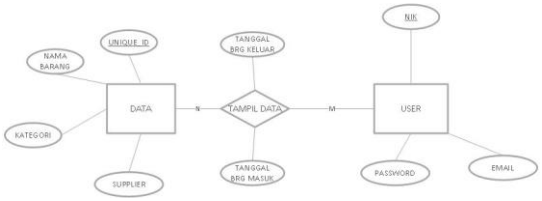
Class diagram adalah diagram yang digunakan untuk menampilkan beberapa kelas yang ada dalam perangkat lunak yang sedang dikembangkan. *Class diagram* menggambarkan struktur dan deskripsi kelas beserta hubungan kelas yang satu dengan yang lain.



Gambar 8 Class Diagram Sistem

d. Analisis Basis Data

Dalam memodelkan data dan menggambarakan hubungan antara data yang ada pada sistem digunakan alat bantu yaitu *Entry Relationship Diagram* (ERD). Sistem yang akan dibangun dapat dilihat hubungan antar entitas dan atributnya. Sistem yang akan dibangun memiliki ERD seperti pada gambar 15.



Gambar 9 Entity Relationship Diagram

e. Skema Relasi

Skema relasi merupakan rangkaian hubungan antara dua tabel atau lebih pada sistem basis data. Berikut adalah perancangan skema relasi untuk aplikasi Smart Warehouse dapat dilihat pada gambar 16.



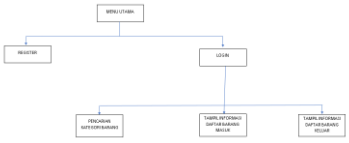
Gambar 10 Skema Relasi

D. Perancangan Sistem

Dalam membangun suatu sistem, salah satu hal yang harus dilakukan yaitu perancangan sistem. Perancangan sistem terdiri dari arsitektur menu dan perancangan antarmuka.

a. Arsitektur Menu

Dalam perancangan sebuah program, dibutuhkan suatu struktur menu yang memaparkan mengenai menu yang akan dibangun.



Gambar 11 Arsitektur Menu

b. Perancangan Antarmuka

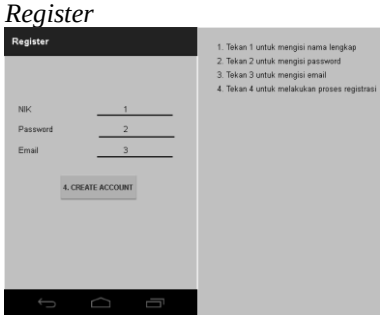
Antarmuka merupakan suatu media komunikasi antara pengguna dengan aplikasi yang dibangun. Oleh karena itu dalam perancangan sebuah aplikasi dibutuhkan suatu perancangan antarmuka guna memberikan gambaran umum bagi pengguna. Perancangan antarmuka aplikasi ini dibagi menjadi beberapa bagian, diantaranya

1. Perancangan Antarmuka Halaman *Login*



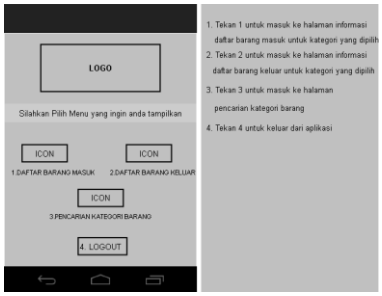
Gambar 12 Perancangan Antarmuka Halaman Login

2. Perancangan Antarmuka Halaman *Register*



Gambar 13 Perancangan Antarmuka Halaman Register

3. Perancangan Antarmuka Halaman Utama



Gambar 14 Perancangan Antarmuka Halaman Utama

4. Perancangan Antarmuka Halaman Informasi Daftar Barang Masuk



Gambar 15 Perancangan Antarmuka Halaman Informasi Daftar Barang Masuk

5. Perancangan Antarmuka Halaman Informasi Daftar Barang Keluar



Gambar 16 Perancangan Antarmuka Halaman Informasi Daftar Barang Keluar

6. Perancangan Antarmuka Halaman Pencarian Kategori Barang



Gambar 17 Perancangan Antarmuka Halaman Pencarian Kategori Barang

4. HASIL PENGUJIAN DAN ANALISA

Pengujian yang dilakukan pada aplikasi Smart Warehouse adalah dengan menggunakan pengujian *aplha blackbox*. Metoda pengujian *aplha blackbox* memfokuskan pada keperluan fungsional dari aplikasi yang telah dibangun.

A. Pengujian

Pengujian dilakukan menggunakan pengujian alpha dengan metode *black box*, yaitu dengan mencoba menu dan fungsi pada aplikasi.

1. Pengujian Splashscreen

Pengujian ini dilakukan setelah user pertama kali membuka aplikasi, Gambar 18 adalah antarmuka dari halaman splash screen, dan Tabel 3 adalah hasil pengujian splash screen



Gambar 18 Halaman Splashscreen

Tabel 3 Hasil Pengujian Splashscreen

Kelas Uji	Butir Uji	Hasil yang diharapkan	Hasil Pengujian
Splash screen	Tampilkan halaman *splash screen*	Menampilkan tampilan *splash screen*	[√]Berhasil [ ] Gagal

2. Pengujian Register

Untuk dapat menggunakan aplikasi ini, pengguna diminta untuk memasukkan alamat email dan *password*. Akses login dapat dilakukan setelah *user* melakukan register terlebih dahulu dengan menekan tombol register. Gambar 19 (a) merupakan tampilan halaman register dan Gambar 19 (b) merupakan tampilan halaman register yang telah diisi biodata user, dan Tabel 4 merupakan tabel dari hasil pengujian register.



(a) (b)

Gambar 19 Halaman Register

Tabel 4 Hasil Pengujian Register

Kelas Uji	Butir Uji	Hasil yang diharapkan	Hasil Pengujian
Register	Tampilkan halaman *register*	Menampilkan tampilan *register*	[√]Berhasil [ ] Gagal
	Mengisi form *register*	Mengisi form *register* yang telah tersedia	[√]Berhasil [ ] Gagal
	Menekan tombol *create account*	Membuat akun untuk login pada aplikasi	[√]Berhasil [ ] Gagal

3. Pengujian Login

Pengujian ini dilakukan setelah pengguna melakukan registrasi, pengguna dapat melakukan login dengan memasukkan NIK dan kata sandi (password). Gambar 20 (a) merupakan tampilan halaman login dan Gambar 20 (b) merupakan tampilan halaman login yang telah diisi nik dan password, dan Tabel 5 merupakan tabel dari hasil pengujian login.



(a) (b)

Gambar 20 Halaman Login

Tabel 5 Hasil Pengujian Login

Kelas Uji	Butir Uji	Hasil yang diharapkan	Hasil Pengujian
Login	Tampilkan halaman *login*	Menampilkan tampilan *login*	[√]Berhasil [ ] Gagal
	Menekan tombol *login*	Melakukan proses *login* dan masuk ke menu utama	[√]Berhasil [ ] Gagal

4. Pengujian Menu Utama

Pengujian ini dilakukan setelah login berhasil, maka akan tampil halaman utama dari aplikasi Smart Warehouse. Gambar 21 adalah antarmuka dari halaman utama, dan Tabel 6 adalah hasil pengujian menu utama.



Gambar 21 Halaman Menu Utama

Tabel 6 Hasil Pengujian Menu Utama

Kelas Uji	Butir Uji	Hasil yang diharapkan	Hasil Pengujian
Menu Utama	Tampilkan informasi daftar barang masuk	Menampilkan halaman informasi daftar barang masuk	[√]Berhasil [ ] Gagal
	Tampilkan informasi daftar barang keluar	Menampilkan halaman informasi daftar barang keluar	[√]Berhasil [ ] Gagal
	Tampilkan pencarian kategori barang	Menampilkan halaman pencarian kategori barang	[√]Berhasil [ ] Gagal
	Menekan tombol *logout*	Kembali ke halaman *login*	[√]Berhasil [ ] Gagal

5. Pengujian Menampilkan Informasi Daftar Barang Masuk

Pengujian ini dilakukan dengan mencocokkan data RFID yang disimpan dengan database oleh middleware, setelah data RFID dicocokkan dengan database, data akan ditampilkan pada android. Gambar 22 adalah antarmuka dari halaman informasi daftar barang masuk, dan Tabel 7 adalah hasil pengujian halaman informasi daftar barang masuk.



Gambar 22 Halaman Informasi Daftar Barang Masuk

Tabel 7 Hasil Pengujian Informasi Daftar Barang Masuk

Kelas Uji	Butir Uji	Hasil yang diharapkan	Hasil Pengujian
Informasi Daftar Barang Masuk	Tampilkan halaman informasi daftar barang masuk	Menampilkan halaman informasi daftar barang masuk.	[√]Berhasil [ ] Gagal

6. Pengujian Menampilkan Informasi Daftar Barang Keluar

Pengujian ini dilakukan dengan mencocokkan data RFID yang disimpan dengan database oleh middleware, setelah data RFID dicocokkan dengan database, data akan ditampilkan pada android. Gambar 23 adalah antarmuka dari halaman informasi daftar barang masuk, dan Tabel 8 adalah hasil pengujian halaman informasi daftar barang keluar.



Gambar 23 Halaman Informasi Daftar Barang Keluar

Tabel 8 Hasil Pengujian Informasi Daftar Barang Keluar

Kelas Uji	Butir Uji	Hasil yang diharapkan	Hasil Pengujian
Informasi Daftar Barang Keluar	Tampilkan halaman informasi daftar barang keluar	Menampilkan halaman informasi daftar barang keluar.	[√]Berhasil [ ] Gagal

7. **Pengujian Pencarian Keberadaan Kategori Barang**
   Pengujian ini dilakukan dengan mengaktifkan buzzer untuk pencarian kategori barang. Gambar 24 adalah antarmuka dari halaman pencarian kategori barang, dan Tabel 9 adalah hasil pengujian pencarian kategori barang.



Gambar 24 Halaman Pencarian Keberadaan Kategori Barang

Tabel 9 Pengujian Pencarian Keberadaan Kategori Barang

Kelas Uji	Butir Uji	Hasil yang diharapkan	Hasil Pengujian
Pencarian Kategori Barang	Menekan Deteksi Barang	Membuka alert dialog untuk menghubungkan koneksi Bluetooth pada aplikasi android	[√]Berhasil [ ] Gagal
	Memilih sambungkan untuk menghubungkan koneksi Bluetooth	Menyambungkan koneksi Bluetooth pada aplikasi android	[√]Berhasil [ ] Gagal
	Memilih Kategori 1	Menyalakan buzzer pada kategori barang 1	[√]Berhasil [ ] Gagal
	Memilih Kategori 2	Menyalakan buzzer pada kategori barang 2	[√]Berhasil [ ] Gagal
	Memilih Putuskan Koneksi	Memutuskan koneksi Bluetooth pada aplikasi android	[√]Berhasil [ ] Gagal

B. Analisa

Berdasarkan hasil pengujian pada aplikasi, aplikasi hanya dapat menampilkan data terakhir yang terdapat pada database, dengan analisa sebagai berikut :

1. Berdasarkan tabel 4.1 dan 4.2, aplikasi android hanya menampilkan 2 barang karena pada saat pengujian, data yang dicocokkan oleh middleware berjumlah 2 buah. Pada saat RFID membaca RFID tag ke 3, koneksi Modul Wi-Fi ESP terputus.
2. Berdasarkan tabel 4.4 aplikasi yang dibuat hanya dapat mendeteksi adanya koneksi Bluetooth hingga jarak maksimal lebih dari 25 meter.
3. Berdasarkan hasil pengujian, proses *pairing* dari aplikasi dengan Bluetooth sangat cepat, namun terkadang proses *pairing* gagal sehingga user dianjurkan untuk merestart aplikasi dan melakukan *pairing* kembali pada Bluetooth.

5. KESIMPULAN DAN SARAN

A. Kesimpulan

Dari hasil pengujian yang telah dilakukan, kesimpulan yang dapat diambil adalah sebagai berikut :

1. Berdasarkan hasil pengujian, secara fungsional sistem dapat berjalan sampai terkirimnya data ke basis data, sehingga aplikasi dapat menampilkan data yang terdapat pada basis data.
2. Aplikasi yang dibangun dapat mencari keberadaan barang., namun masih sering terjadi bug seperti proses *pairing* dari aplikasi dengan Bluetooth terkadang gagal sehingga user dianjurkan untuk merestart aplikasi dan melakukan *pairing* kembali pada Bluetooth.

B. Saran

Untuk saran yang dapat dijadikan pertimbangan untuk pengembangan aplikasi ini selanjutnya yaitu dengan membuat data yang ditampilkan secara online sehingga pengguna dapat melihat data barang di mana saja dan kapan saja, selain itu tampilan aplikasi dibuat menjadi lebih menarik dan praktis dalam penggunaannya serta memperbaiki fungsi yang tidak berjalan dengan semestinya.

REFERENSI

[1] J. B. W. Company, "Smart Warehouses, Smarter Productivy,"2008.[Online].Available:http://www.abelwomack.com/wpcontent/uploads/Smart_Warehouse_White_Paper.pdf. [Diakses 3 Desember 2016].

[2] H, Nazaruddin Safaat. (2015). Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android, Bandung: Informatika

[3] B, Hariyanto. (2004). *Rekayasa Sistem Berorientasi Objek*.Bandung: Informatika

[4] F, Martin. (2005). *UML Distilled 3th Ed., Panduan Singkat Bahasa Pemodelan Objek Standar*. Yogyakarta : ANDI

[5] B. Raharjo, Heryanto Imam and A. Haryono. (2010). Mudah Belajar JAVA, Bandung: Informatika