

Penggunaan *Ultra High Frequency* RFID pada Sistem Inventarisasi Creatrix Organizer

Welly Tedja Kusuma
Program Studi Informatika
Fakultas Teknologi Industri
Universitas Kristen Petra
Jl. Siwalankerto 121 – 131
Surabaya 60236

Telp. (031) – 2983455, Fax. (031) –
8417658

w3llyliem@gmail.com

Yulia
Program Studi Informatika
Fakultas Teknologi Industri
Universitas Kristen Petra
Jl. Siwalankerto 121 – 131
Surabaya 60236

Telp. (031) – 2983455, Fax. (031) –
8417658

yulia@petra.ac.id

Resmana Lim
Program Studi T. Elektro
Fakultas Teknologi Industri
Universitas Kristen Petra
Jl. Siwalankerto 121 – 131
Surabaya 60236

Telp. (031) – 2983455, Fax. (031) –
8417658

resmana@petra.ac.id

ABSTRAK

Pada saat ini, *Event Organizer* Creatrix Organizer melakukan pendataan barang di Gudang secara manual. Creatrix Organizer belum menggunakan sistem yang terstruktur untuk melakukan pendataan. Creatrix Organizer sering kesulitan untuk mencari barang yang diperlukan untuk kebutuhan *event* pameran berskala kecil. Pada *event* berskala kecil tidak semua barang dibutuhkan dan memakan waktu untuk mencari barang yang diperlukan untuk kebutuhan *event*. Hal tersebut mengakibatkan keterlambatan penyelesaian suatu pekerjaan dan banyak proses yang tidak dapat diselesaikan secara maksimal di suatu *event* tersebut. Dampak dari hal itu membuat pelanggan menjadi tidak puas atas hasil yang kurang maksimal.

Melihat berbagai permasalahan yang dialami oleh *Event Organizer Creatrix Organizer* ini, maka dirancang sistem yang dapat melakukan pendataan barang secara otomatis menggunakan bantuan alat *Ultra High Frequency (UHF) Radio Frequency Identification (RFID)*. Tidak hanya menggunakan alat, aplikasi berbentuk *website* menggunakan *Framework CodeIgniter* dan aplikasi di *smartphone* menggunakan *Framework Ionic* dibuat untuk memudahkan para pekerja untuk mencari barang-barang yang diperlukan. *Database* dari kedua aplikasi tersebut menggunakan *MySQL*.

Hasil yang diperoleh dari aplikasi yang telah dibuat salah satunya adalah dapat melakukan pendataan barang secara otomatis menggunakan *UHF Radio Frequency Identification (RFID)*. Untuk melakukan pendataan dan melihat ketersediaan barang dapat dilakukan melalui *website*. Sedangkan aplikasi *smartphone* digunakan untuk melihat ketersediaan barang.

Kata Kunci: *Ultra High Frequency (UHF) Radio Frequency Identification (RFID), Framework CodeIgniter, Framework Ionic, Pendataan Barang Inventori.*

ABSTRACT

At this moment, Event Organizer Creatrix Organizer records its items manually. Creatrix Organizer have not used structural system to record the items yet. Creatrix Organizer always having trouble to find the item for small scale event. The small-scale events does not require all the items from the warehouse and take time to find the items needed for the event. This situation will result in a delayed process. The customer will not be satisfied because the work is not optimal. Based on the Creatrix Organizer

problem, the authors design a system that can record and count items automatically using Ultra High Frequency (UHF) Radio Frequency Identification (RFID). Not only tool, but also a website using CodeIgniter Framework and smartphone application using Ionic Framework for worker to make their job easier to find the needed items for the event. The database used for the applications is MySQL.

The results of the applications are, they can record and count the items automatically using UHF Radio Frequency Identification (RFID). Furthermore, they can record and search the items that available in the website. Smartphone application is used for searching the items that available to use.

Keywords: *Ultra High Frequency (UHF) Radio Frequency Identification, CodeIgniter Framework, Ionic Framework, Record Inventory Items.*

1. PENDAHULUAN

Acara-acara besar di mal seperti pameran tentunya memerlukan pihak ketiga yang sering juga disebut sebagai vendor. Vendor tersebut menyediakan berbagai perlengkapan yaitu meja, kursi, sketsel dan alat-alat lainnya untuk memenuhi kebutuhan yang dapat membuat acara berjalan dengan lancar. Tetapi seringkali vendor-vendor tersebut bingung dengan barang yang mereka punya, dimana letak gudang mereka menyimpan barang-barang yang diperlukan. Hal tersebut sangatlah wajar karena gudang para vendor biasanya tersebar, tidak hanya berada di satu lokasi saja.

Perkembangan zaman juga terjadi pada dunia vendor. Pencatatan dan penghitungan barang dapat dibantu dengan menggunakan sensor tertentu yang dapat menggantikan peran kerja manusia untuk mencatat dan menghitung barang-barang di gudang. Sistem pergudangan yang terkomputerisasi membuat kita menjadi lebih sedikit bekerja, hasil yang lebih efisien dan stabil [8]. Kita sebagai manusia yang hidup di dunia *modern* juga sangat terbantu jika ada sistem sensor yang dapat membantu kerja manusia menjadi lebih cepat dan otomatis atau dapat berjalan sendiri. Hal ini juga membuat para pekerja tidak bingung untuk menaruh barang pada gudang tanpa perlu repot-repot untuk mencatat.

Radio Frequency Identification atau sering disebut sebagai RFID merupakan suatu teknologi yang dapat digunakan secara otomatis dan tanpa memerlukan kabel [3]. RFID dapat menggantikan peran kerja manusia dalam melakukan pencatatan dan penghitungan barang pada suatu gudang secara otomatis. Unit dasar dalam penggunaan RFID adalah sebuah tag RFID [3]. Tag tersebut akan

diletakkan di barang yang akan di data, sehingga dapat langsung terdeteksi oleh RFID reader dan akan otomatis terinput ke dalam sistem. Sistem tersebut dapat menampilkan letak barang tersebut berada di gudang yang mana.

Pembuatan aplikasi ini bertujuan untuk mempermudah para pekerja vendor untuk mengecek barang yang mereka miliki berada di gudang mana. Dibandingkan dengan tradisional *barcode*, *Radio Frequency Identificaion* (RFID) dapat meningkatkan efisiensi dalam membaca sebuah data, jarak jangkauan deteksi yang jauh dan lebih aman [6]. Dengan bantuan UHF *Radio Frequency Identification* (RFID) yang akan membuat sistem menjadi *online*. Pekerja tidak perlu mencatat secara manual, semua otomatis akan terhitung oleh sistem melalui UHF *Radio Frequency Identification* (RFID). Tidak hanya itu, aplikasi ini juga dirancang agar dapat melihat apa saja kebutuhan pada acara tersebut dan dimana letak acara tersebut akan diadakan. Sehingga para pekerja akan lebih sedikit untuk melakukan kesalahan.

2. DASAR TEORI

2.1 Radio Frequency Identification (RFID)

RFID merupakan salah satu perkembangan dari teknologi *barcode* yang berbentuk *wireless*. Dibandingkan dengan teknologi *barcode*, RFID dapat meningkatkan efisiensi dari membaca sebuah data dengan jarak yang lebih jauh dan sekuriti yang lebih terjamin [6]. Tag RFID merupakan unit yang dibutuhkan selain RFID *reader*. Tag dibagi menjadi dua yaitu tag pasif dan aktif, dimana tag pasif tidak memiliki sumber daya sedangkan tag aktif mempunyai sumber daya. RFID sendiri memiliki beberapa golongan antara lain:

a. Low Frequency

Frekuensi antara 120 – 134.2 KHz, memiliki jarak baca yang dekat yaitu 10 cm. Lambat dalam membaca sebuah tag dan memiliki sensitifitas kecil terhadap gelombang radio

b. High Frequency

Memiliki frekuensi antara 13,56 MHz dan jarak baca antara 10cm hingga 1 meter.

c. Ultra High Frequency

Memiliki frekuensi dari 850 – 960 MHz dan jangkauan jarak baca hingga 3 meter. Memiliki tag termurah untuk diproduksi dibandingkan yang lainnya. UHF RFID biasanya dipakai di warehouse dalam skala besar dimana kecepatan dan efisiensi merupakan faktor utama.

Gambar 1. merupakan tabel perbandingan golongan *Radio Frequency Identificaion* (RFID). Pada masing-masing golongan mempunyai perbedaan. Perbedaan tersebut membuat penggunaan tiap golongan berbeda-beda satu dengan yang lainnya.

	LF	HF	UHF	Active
Frequency	125 – 134.2 KHz	13.56 MHz	850 – 960 MHz	100 KHz – 2.45GHz
Range	0.2 – 2m	Up to 1m	Up to 3m	Up to 100m
Cost	Typ. 3 GBP	(Typ. 0.50 GBP)	(Typ. 0.30 GBP)	(Typ. 20 GBP)
Memory	Typ. 64 bits	Typ. 2048 bits	Typ. 96 bits	Typ. 32 bits
Penetration of Materials	V. Good	Good	Poor	V. Good
Data Rate	Slow	Fast	Fast	Fast
Reader Cost	50 – 500 GBP	50 – 3000 GBP	1000- 3000 GBP	200-600 GBP
Read Multiple Tags	Poor	Good	Very Good	Good
Applications	Animal Tags, Vehicle Immobilisers, Industrial Applications	Item Tracking, Access Control, Smart Labels	Box and Pallet tracking, Some Item Tracking	Industrial Applications, Asset Tagging, Location Systems

Gambar 1. Perbandingan Golongan RFID [1]

2.2 Tag Radio Frequency Identification (RFID)

Radio Frequency Identification (RFID) dapat membaca ratusan tag RFID yang unik dari jarak yang jauh [2]. *Tag Radio Frequency Identification* (RFID) digunakan sebagai alat pendeteksi dan memiliki bentuk yang bermacam-macam, ada yang berbentuk seperti kartu, *chip* berwarna biru, *sticker* dan berbagai bentuk lainnya. Bentuk tag dapat dibuat sesukanya oleh penggunanya, tetapi hal yang pasti adalah setiap tag mempunyai frekuensi agar dapat di deteksi oleh sensor *Radio Frequency Identification* (RFID). Penggunaan bentuk tag disesuaikan dengan kebutuhan pada saat memakai *Radio Frequency Identification* (RFID). Tag berbentuk *sticker* merupakan bentuk yang paling fleksibel untuk digunakan dalam *inventory control*, karena dapat mendeteksi barang dengan hanya menempelkan tag *sticker* tersebut ke barang yang ingin di data. Gambar 2. adalah berbagai bentuk dari tag *Radio Frequency Identification* (RFID).



Gambar 2. Bentuk Tag RFID [4]

2.3 Framework CodeIgniter

CodeIgniter merupakan framework yang digunakan untuk membuat website. Codeigniter bersifat open source yang artinya dapat digunakan untuk membangun php php dinamis. Codeigniter mempunyai fungsi-fungsi / prosedur dan *class-class* yang dapat mempermudah dan mempercepat pekerjaan seorang *programmer*. CodeIgniter *Framework* menjadi sebuah *Framework* berbasis PHP dengan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis dengan menggunakan bahasa PHP yang dapat mempercepat pengembang untuk membangun sebuah *website* [5].

2.4 Framework Ionic

Ionic merupakan framework yang biasanya digunakan untuk pembuatan aplikasi pada ponsel. Ionic bersifat open source sehingga dapat digunakan secara bebas. Dengan menggunakan Ionic dapat mermpermudah dan mempercepat pekerjaan seorang *programmer*. Ionic *Framework* merupakan *Framework* yang bersifat *open source* yang menyediakan *UI toolkit* untuk membuat aplikasi *website* PWA, *desktop*, *mobile* (Android dan IOS) dengan menggunakan bahasa *website* seperti HTML, CSS, dan Javascript [7].

3. DESAIN SISTEM

3.1 Analisis Perusahaan

Event Organizer merupakan sebuah badan usaha yang bergerak di bidang penyedia jasa di Surabaya. Creatrix Organizer merupakan

salah satu penyedia jasa tersebut yang menawarkan beberapa jasa khusus untuk pameran mengenai *travel fair* dan penyewaan booth untuk bank ataupun pameran kecil lainnya. Beberapa jasa tersebut antara lain meminjamkan kebutuhan pameran seperti booth, meja, kursi, layar, dll. Tidak hanya penyedia barang, tetapi juga penyedia jasa untuk mengurus antrian *cashback*, pembayaran dan *redeem* hadiah.

Pada *event* besar tersebut, banyak barang keluar masuk gudang tanpa adanya control. Barang-barang tersebut tersebar di beberapa gudang yang berada di Surabaya. Pada saat *event* pameran kecil yang tidak membutuhkan semua barang, terjadi kendala yaitu kesulitan untuk mencari barang yang diperlukan ataupun lama mencari barang yang diperlukan tersebut. Proses pencarian barang tersebut memakan waktu yang cukup lama, karena jarak antar gudang juga tidak terlalu dekat atau menyebar di Surabaya.

Pada Creatrix Organizer sering mengalami kendala dalam mencari barang yang diperlukan pada saat *event* berskala kecil. Barang-barang yang diperlukan biasanya tersebar di beberapa gudang dan hanya dapat mengira-ngira dari letak terakhir kali barang tersebut di masukkan ke dalam gudang. Hal tersebut terjadi karena hanya beberapa orang yang berasal dari perusahaan sendiri untuk menangani proses bongkar muat dan pemasangan barang di dalam mall. Creatrix Organizer mengandalkan beberapa orang yang dipekerjakan perhari dan tidak terlibat kontrak dengan perusahaan, sehingga susah untuk menangani hal pengecekan atau pendataan barang.

3.1.1 Sistem Pendataan Barang

Untuk proses penyewaan jasa Creatrix Organizer sendiri memiliki alur. Hal pertama yang akan dilakukan yaitu menentukan kebutuhan barang apa saja yang akan mendukung berjalannya acara pameran. Setelah melakukan hal tersebut, tim dari Creatrix Organizer akan memberikan design atau layout booth-booth yang akan berada di venue tempat pameran. Tentu saja tidak semua peralatan dimiliki oleh Creatrix Organizer, hal tersebut akan dikoordinasi oleh tim untuk mencari vendor yang mau bekerja sama contohnya seperti layar *lcd* pada panggung. Setelah semua persiapan sudah dilakukan, proses pemasangan akan dilakukan. Untuk *event* dalam skala besar, biasanya pihak mal akan memberikan waktu pemasangan mulai dari hari senin dikarenakan harus memasang banyak booth besar, untuk *event* dalam skala kecil hanya membutuhkan 1-2 hari saja.

Pada saat *event* berlangsung biasanya juga terdapat beberapa perombakan atau pergantian barang dikarenakan kurang sesuai atau rusak. Setelah semua itu selesai, barang-barang tersebut harus di bongkar dalam waktu 1 hari yaitu pada hari minggu malam, karena hari senin sudah harus kosong sehingga penyelenggara *event* berikutnya di *venue* tersebut dapat mempersiapkan kebutuhan mereka. Dalam waktu 1 malam tersebut, tentu saja banyak barang yang akan di bongkar secara cepat tanpa bisa memilih atau mengelompokkan barang dan cepat untuk di masukkan ke dalam gudang.

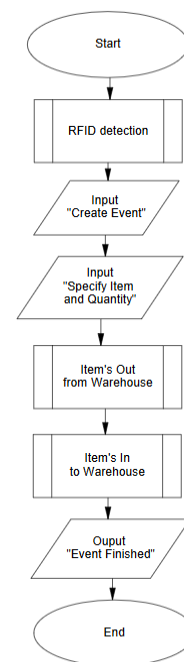
Proses-proses yang dikerjakan oleh manusia bisa digantikan dengan menggunakan teknologi RFID. Dengan menggunakan RFID maka akan mempermudah pekerjaan manusia dalam hal pendataan barang. Barang di gudang akan terdata secara rapi di sistem dan dapat dilihat melalui website maupun aplikasi yang ada di *smartphone*.

3.2 Alur Sistem

Pada alur sistem akan terdapat beberapa proses yang perlu dijalankan agar dapat melakukan pendataan barang dengan

menggunakan RFID. Sistem ini memiliki alur untuk melakukan sistem pencatatan, antara lain:

1. *Input "Create Event"*
Proses ini merupakan pembuatan event yang akan diadakan.
2. *Input "Specify Item and Quantity"*
Pada bagian ini akan melakukan input barang dan jumlah barang yang diperlukan untuk kebutuhan *event*.
3. *Item's Out from Warehouse*
Pada bagian ini merupakan pencatatan barang yang keluar dari gudang menggunakan bantuan RFID dan akan tercatat ke dalam sistem.
4. *Item's Into Warehouse*
Pada bagian ini merupakan pencatatan barang yang masuk ke dalam gudang menggunakan bantuan RFID dan akan tercatat ke dalam sistem.
5. *Output "Event Finished"*
Bagian ini menandakan bahwa *event* telah selesai dilaksanakan.



Gambar 3. Flowchart Garis Besar Sistem

3.2.1 Alur Pendaftaran Barang ke Sistem

Proses ini dijalankan melalui website untuk memasukkan barang yang belum terdaftar di dalam sistem. Pada saat mendaftarkan barang ke dalam sistem terdapat 2 proses yang perlu dikerjakan. Proses pertama yaitu mendaftarkan model barang yang akan didaftarkan ke dalam sistem. Pada saat mendaftarkan model barang ke dalam sistem akan ada data yang harus diisi, yaitu:

1. Enter Model Name

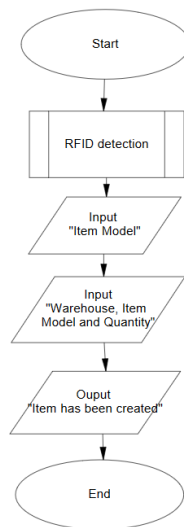
Pada bagian ini diisi dengan nama model barang yang akan dimasukkan ke dalam sistem. Model barang tersebut tidak bisa menggunakan nama yang sama dan harus dibedakan, misal meja plastik dan meja kayu.

Proses kedua untuk melakukan pendaftaran barang yaitu melakukan penambahan *quantity* pada model barang tersebut. Pada proses ini akan ada beberapa data yang harus diisi, antara lain:

1. *Warehouse*
Pada bagian ini merupakan daftar gudang yang dimiliki oleh Creatrix Organizer. Bagian ini dibuat dalam bentuk pilihan sehingga lebih mudah saat mendaftarkan barang dan tidak salah memasukkan model barang di gudang.
2. *Item Model*
Pada bagian ini akan ada daftar model barang yang sudah tercatat di sistem. Bentuk dari bagian ini dibuat dalam bentuk pilihan agar pada saat mendaftarkan barang menjadi lebih mudah dan model barang yang dimaksudkan lebih jelas sehingga mengurangi kesalahan memasukkan model barang.
3. *Item Tag*
Bagian ini merupakan bagian *tag* setiap barang akan di *input*. Setiap barang akan mendapatkan *tag* sendiri dan tidak akan ada *tag* yang kembar satu sama lain.

Dalam proses pendaftaran barang akan terdapat beberapa proses yang perlu dijalankan akan tercatat ke dalam sistem. Proses-proses tersebut antara lain:

1. *Input "Item Model"*
Pada proses ini merupakan proses untuk memasukkan nama model dari suatu barang
2. *Input "Warehouse, Item Model and Quantity"*
Pada proses ini akan memilih gudang, model barang dan memasukkan tag sehingga akan tercatat di dalam sistem, secara otomatis jumlah barang akan bertambah di dalam sistem.
3. *Output "Item has been created"*
Proses ini menandakan bahwa proses pendaftaran barang ke dalam sistem telah berhasil dan selesai dilakukan.



Gambar 4. Flowchart Pendaftaran Model Barang

3.2.2 Alur Pendaftaran Event ke Sistem

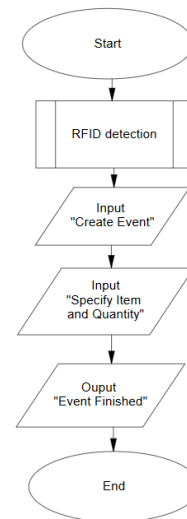
Proses ini akan dijalankan melalui website. Pada saat membuat event akan ada beberapa data yang harus diisi, antara lain:

1. *Enter Event Name*
Nama event diisi sesuai dengan *event* yang akan diadakan.
2. *Date Start*
Pada bagian ini akan diisi tanggal mulai diadakannya *event*.
3. *Date End*
Pada bagian ini akan diisi tanggal selesainya *event* yang diadakan.

4. *Location*
Bagian ini berisi lokasi diadakannya *event*, bagian ini bisa diisi berupa nama jalan maupun mal tempat diadakannya acara. Karena kebanyakan acara pameran diadakannya di mal-mal yang ramai pengunjung.
5. *Venue*
Bagian ini berisi tempat diadakannya *event*, pada bagian ini membantu untuk memperjelas *event* akan diadakan di gedung atau atrium berdasarkan lokasi. Contoh lokasi pada no. 4 akan diisi dengan Supermall – Surabaya, maka pada bagian ini akan berisi atrium tempat diadakannya *event*, yaitu *Grand Atrium*.
6. *Note*
Note merupakan catatan tambahan jika ada hal yang perlu ditambahkan untuk mendukung keberlangsungan *event*, detail-detail yang disampaikan oleh pelanggan dapat disimpan disini agar memudahkan pekerjaan. Pada bagian ini dapat dikosongi apabila tidak ada catatan tambahan.

Alur sistem pendaftaran *event* memiliki beberapa proses bagian agar proses pendaftaran tersebut terpenuhi, antara lain:

1. *Input "Create Event"*
Pada bagian ini kita akan mendaftarkan *event* yang akan diadakan.
2. *Input "Specify Item dan Quantity"*
Pada bagian ini akan diisi berupa barang dan jumlah barang yang akan diperlukan untuk kebutuhan *event*.
3. *Output "Success Create Event"*
Pada bagian ini merupakan penanda bahwa *event* yang telah dibuat telah masuk ke dalam sistem.

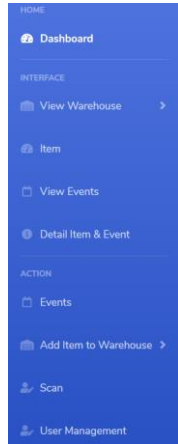


Gambar 5. Flowchart Pendaftaran Event

3.3 Desain Aplikasi

Pada tampilan utama *website* akan terbagi menjadi 3 menu, yaitu *home*, *interface*, *action*. Pada bagian *home* hanya menampilkan *dashboard*, sedangkan pada bagian *interface* akan menampilkan bagian yang digunakan untuk mencari barang atau hanya sekedar melihat barang. Pada bagian *action*, *user* akan melakukan *input* data atau pun melakukan aksi untuk membuat *event*,

mendaftarkan model dan jumlah barang, melakukan pendataan barang, dan melakukan pengaturan mengenai *user*.



Gambar 6. Tampilan Menu Website

4. IMPLEMENTASI SISTEM

Implementasi sistem dilakukan dengan mengkonfigurasi *Radio Frequency Identification* (RFID) reader. Aplikasi yang digunakan untuk mengkonfigurasi *Radio Frequency Identification* (RFID) di buat dengan bahasa pemrograman C Sharp. Aplikasi ini dibuat dan hanya bisa dijalankan melalui *Windows Executable*.

Setelah aplikasi tersebut dijalankan, hasil dari akan berbentuk file berformat txt dan txt tersebut akan dijalankan melalui *website* untuk memasukkan data *tag* ke dalam *database*.

Alat yang dibutuhkan *Radio Frequency Identification* (RFID) reader CT-I809, serta *tag Ultra High Frequency*

5. ANALISA DAN PENGUJIAN

5.1 Pengujian RFID Berdasarkan Jarak

Pada pengujian *Radio Frequency Identification* (RFID) yang berdasarkan jarak terdapat 5 titik sebagai acuan, setiap titik berjarak 1meter satu dengan yang lainnya. Pengujian ini dilakukan dengan menempelkan *tag* pada benda.

Pada pengujian jarak dilakukan 3 kali pengujian untuk mendapatkan hasil test yang akurat. Proses pengujian dilakukan dengan menggunakan 10 *tag* sebagai percobaan pertama, jika pada jarak tersebut dapat mendeteksi 10 *tag* secara bersamaan maka akan diuji untuk mendeteksi **sebanyak 15 tag**. 10 *tag* tersebut merupakan 5 *tag* silver dan 5 *tag* putih. Sedangkan jika uji coba tidak dapat mendeteksi sebanyak 10 *tag* secara bersamaan maka akan diuji coba menggunakan 5 *tag* dengan masing-masing warna diberikan pengujian sendiri. Pengujian ini dilakukan dengan cara melewati *Radio Frequency Identification* (RFID) reader selama 5 detik.

Table 1. Pengujian Sistem Berdasarkan Jarak dengan 10 Tag

Jarak	Akurasi tertinggi	Akurasi terendah	Rata-rata akurasi
1 Meter	90%	80%	81.6%
2 Meter	80%	60%	70%
3 Meter	70%	60%	68.3%
4 Meter	70%	50%	56.6%
5 Meter	60%	40%	46.6%

5.2 Pengujian RFID Berdasarkan Jumlah Deteksi

Pengujian ini dilakukan dengan tujuan untuk mengetahui *Radio Frequency Identification* (RFID) reader dengan tipe CT-I809 mampu untuk mendeteksi lebih dari 10 *tag* secara bersamaan. Berdasarkan Table1. tidak memungkinkan untuk mendeteksi 10 *tag* secara bersamaan. Dengan uji coba memasukkan 10 *tag* secara bersamaan akurasi tertinggi yang didapatkan adalah 90% pada jarak deteksi 1meter.

5.3 Pengujian Tag Radio Frequency Identification (RFID)

5.3.1 Pengujian Tag Terendam Air

Pada pengujian *tag* terendam oleh air, *tag* masih terdeteksi secara normal tidak ada kendala. *Tag* masih dapat terdeteksi oleh *Radio Frequency Identification* (RFID) reader seperti normal meskipun telah terendam oleh air. Hal ini membuktikan bahwa *tag* tersebut tahan terhadap air atau bisa disebut *waterproof*.



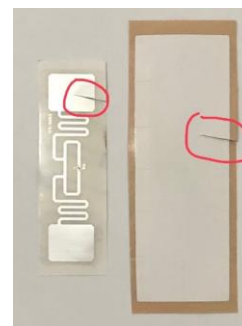
Gambar 7. Tag Terendam Air

No.	ID	EPCLength	Times
1	E200001D770F009017603CAD	12	8
No.	ID	EPCLength	Times
1	E200001D8911012625305D2F	12	11

Gambar 8. Tag Masih Terdeteksi Aplikasi Setelah Terendam

5.3.2 Pengujian Tag Terpotong Sebagian

Pada pengujian ini *tag* akan diuji coba dengan cara dipotong sebagian. *Tag* yang sudah terpotong sebagian masih dapat terdeteksi oleh *Radio Frequency Identification* (RFID) reader seperti normal. Hal ini membuktikan bahwa jika *tag* secara tidak sengaja terpotong ataupun tergores masih dapat dipakai. Jika *tag* terpotong secara keseluruhan maka *tag* tersebut tidak dapat dipakai lagi.



Gambar 9. Tag Terpotong Sebagian

No.	ID	EPCLength	Times
1	E200001D770F009017603CAD	12	5

No.	ID	EPCLength	Times
1	E200001D8911012625305D2F	12	5

Gambar 10. Tag Masih Dapat Terdeteksi Setelah Terpotong Sebagian

5.4 Pengujian Tata Letak Radio Frequency Identification (RFID) Reader dan Tag pada Benda

5.4.1 Pengujian Tata Letak Reader

Pengujian ini dilakukan terhadap letak reader ditempatkan. Radio Frequency Identification (RFID) reader tetap dapat mendeteksi tag selama dalam jangkauan reader tersebut. Jangkauan reader tersebut lebih mirip dengan lapangan baseball.



Gambar 11. Jangkauan Deteksi Reader

5.4.2 Pengujian Tata Letak Tag pada Benda

Pengujian ini dilakukan untuk mengetahui apakah tag dapat tetap terdeteksi apabila ditempelkan di berbagai tempat sebuah benda. Tag dapat diletakkan disegala posisi tidak harus diposisi tertentu. Tetapi tag tersebut tidak dapat diletakkan di satu tempat jika barang tersebut seperti tertumpuk dan tegak lurus dengan reader.

Hasil dari peletakkan tersebut reader tidak dapat mendeteksi secara keseluruhan jika hal tersebut dilakukan secara tegak lurus. Beberapa tag yang berada di posisi terdapat akan menghalangi penangkapan tag yang berada di belakang.

6. KESIMPULAN

Setelah dilakukan perancangan sistem, pengimplementasian, dan pengujian terhadap aplikasi yang telah dibuat, dapat ditarik kesimpulan sebagai berikut:

- Radio Frequency Identification (RFID) reader tidak dapat mendeteksi tag lebih dari 10 secara bersamaan. Posisi tag mempengaruhi pendeteksian. Jika posisi tag yang berada di belakang tag lainnya maka akan terhalang oleh tag yang berada di depan sehingga tidak dapat terdeteksi.

- Tingkat akurasi tertinggi untuk mendeteksi 10 tag adalah 90%, hanya dapat mendeteksi 9 barang jika tag yang berada di posisi belakang tidak terhalang oleh tag yang berada di posisi depan.
- Radio Frequency Identification (RFID) reader mampu mendeteksi tag yang berada di jarak 5 meter. Hal tersebut harus dilakukan dengan konfigurasi power 30 pada aplikasi.

Saran untuk pengembangan kedepannya adalah:

- Penyempurnaan program agar lebih mudah dilakukan oleh admin (user friendly)
- Pendeteksian tag dilakukan per 5 tag dikarenakan memiliki akurasi yang tinggi. Jika dilakukan dengan mendeteksi sedikit tag perlahan-lahan atau sedikit, aplikasi mampu untuk mendeteksi semuanya.
- Pendeteksian tag tidak dilakukan dengan cara menumpuk pada tempat yang sama atau tidak terhalang oleh tag lainnya agar dapat terdeteksi.
- Peletakkan Radio Frequency Identification (RFID) reader diletakkan di dekat pintu masuk setinggi 1.5 meter agar dapat menjangkau seluruh tag dengan posisi reader menghadap ke arah tag tersebut.

7. DAFTAR PUSTAKA

- [1] Advancedmobilegroup. 2015. *Why You Should Care About RFID Frequencies*. URI=<https://www.advancedmobilegroup.com/blog/rfid-which-frequency-is-the-right-choice>.
- [2] Bevilacqua, M., Ciarapica, F. E., Crosta, A., Mazzuto, G., & Paciarotti, C. 2015. Implementation of a RFID system in a furniture industry involved in the fashion sector: A case study. *International Journal of RF Technologies*, 6(2-3), 99-119.
- [3] D'Souza, P., Guo, M., Wang, D., & Lee, I. 2013. Real-time Inventory Management with RFIDash. Dept . of CIS-Senior Design 2012-2013.
- [4] Elangsakti. Pengertian RFID dan Aplikasinya : Case RFID Reader MLF8112WA. URI=<https://www.elangsakti.com/2015/09/pengertian-rfid-adalah.html>.
- [5] Idcloudhost.. 2017. Mengenal Apa itu Framework CodeIgniter. URI=<https://idcloudhost.com/panduan/mengenal-apa-itu-framework-codeigniter/>.
- [6] LI, W., & FAN, T. J. 2016. The Design of An RFID-based Warehouse Management System for Fresh Agricultural Products of Stores. *DEStech Transactions on Social Science, Education and Human Science*, (hsc).
- [7] Petanikode. 2019. Belajar Membuat Aplikasi Android Menggunakan Ionic Framework (Untuk Pemula). URI=<https://www.petanikode.com/ionic-untuk-pemula/>.
- [8] Tejesh, B. S. S., & Neeraja, S. 2018. Warehouse inventory management system using IoT and open source framework. *Alexandria engineering journal*, 57(4), 3817-3823.