

APLIKASI MOBILE MENGGUNAKAN FRAMEWORK PHONEGAP UNTUK MONITORING PERSEDIAAN BARANG PADA PERUSAHAAN DISTRIBUSI SECARA REAL-TIME

¹Jordy Saragih, ²Atje Setiawan Abdullah, ³Juli Rejito

^{1,2,3}Program Studi Teknik Informatika Departemen Matematika FMIPA Universitas Padjadjaran

¹jordysaragih@gmail.com, ²atjesetiawan@gmail.com, ³rejito@unpad.ac.id

Abstrak

Pada era teknologi informasi seperti sekarang ini, efisiensi waktu dan tenaga serta kemudahan dalam mendapatkan suatu informasi sangatlah dibutuhkan. Dalam sebuah perusahaan yang bergerak dalam bidang penjualan dan distribusi barang, aplikasi untuk monitoring kondisi, ketersediaan, dan akurasi jumlah barang secara real time menjadi suatu kebutuhan yang tidak dapat dipisahkan.

Rancangan aplikasi mobile yang dibangun memiliki antarmuka berbasis web memanfaatkan pemrograman berorientasi objek yang didukung framework phonegap dan pemrograman script PHP sehingga aplikasi ini dapat dipergunakan secara mobile kapanpun dan dimanapun secara real time. Implementasi aplikasi mobile ini menggunakan arsitektur client-server tersambung pada jaringan internet, dengan antarmuka aplikasi pada posisi client dapat menggunakan smartphone dan tablet, sedangkan server databasenya menggunakan Mysql. Disamping aplikasi mobile yang dimanfaatkan untuk monitoring, disini yang lain dibangun juga aplikasi untuk update transaksi barang.

Dari hasil pengujian aplikasi dihasilkan kesimpulan bahwa update data transaksi barang masuk dan keluar menjadi bagian yang terpenting sehingga aplikasi mobile dapat menyajikan informasi tentang persediaan barang. Aplikasi mobile ini memberikan kemudahan karena informasi yang dihasilkan secara cepat dan akurat serta real time kepada pimpinan perusahaan dapat dijadikan sebagai bahan dalam mengambil keputusan bisnis.

Kata Kunci: aplikasi mobile, framework, PhoneGap, real time.

1 Pendahuluan

Dewasa ini, perkembangan teknologi informasi begitu cepat, sehingga sangat membantu manusia dalam menyelesaikan pekerjaan yang ada. Dahulu manusia menyimpan semua dokumen dan informasi menggunakan kertas secara manual, kemudian disimpan dalam suatu tempat, sekarang metode penyimpanan tersebut berubah menjadi penyimpanan *file* yang berisikan data perusahaan menggunakan komputer (*computerized*). Diharapkan penyimpanan data dapat dilakukan dengan baik, dan terhindar dari kesalahan yang

terjadi apabila penyimpanan dilakukan secara manual.

Setiap perusahaan yang bergerak dalam bidang penjualan dan distribusi barang memiliki kebutuhan informasi yang berbeda-beda untuk memenuhi kebutuhan serta meningkatkan produktivitas perusahaan tersebut. Banyak perusahaan melakukan penyimpanan data secara manual, berpindah menggunakan *software* pendukung yang umum digunakan untuk menyimpan data-data barang yang masuk dan keluar, serta ketersediaan barang yang ada di gudang. Namun kendala dalam menampilkan data tetap ditemukan ketika semua informasi tersebut dibutuhkan dengan cepat dan akurat, sehingga dibutuhkan *software* aplikasi yang lebih sesuai untuk dapat memenuhi kebutuhan tersebut.

Smartphone adalah sebuah ponsel yang menawarkan kemampuan lebih serta komputasi canggih dan konektivitas dari telepon dengan fitur terbaru.^[1] Mereka jauh lebih efisien dalam faktor bentuk, jenis chip, kapasitas penyimpanan internal, masa pakai baterai dan sistem operasi. *Platform mobile modern* yang banyak dikenal adalah Symbian, iPhone, Android, Windows Phone 7, Palm, Blackberry, dll.^[2] Aplikasi *hybrid* adalah kombinasi dari aplikasi asli dan aplikasi Web. Kerangka *smartphone* dipengaruhi oleh cepatnya teknik pengembangan aplikasi.^[3]

PhoneGap adalah sebuah *framework open-source* yang dapat membangun aplikasi *mobile cross-platform* seperti pada iOS dan Android. PhoneGap menjadi trend sejak tahun 2009 dan versi stabil dari PhoneGap dirilis pada April 2015. Dalam pengembangannya, PhoneGap hanya menggunakan bahasa Html, javascript dan CSS tanpa perlu menggunakan java(android), objective C(ios), c# untuk windows phone, dsb. Sampai saat ini telah ada lebih dari tiga ribu enam ratus aplikasi yang dibuat menggunakan PhoneGap, seperti Snowbuddy, Sworkit, dan lainnya. Aplikasi PhoneGap dikembangkan menggunakan HTML, CSS, dan JavaScript. Namun produk akhir dari aplikasi PhoneGap adalah aplikasi arsip biner yang

dapat didistribusikan melalui ekosistem aplikasi standar.^[4]

Tujuan dari penelitian ini adalah (1) Mampu menampilkan data persediaan barang secara *online* dan *realtime*, serta tepat dan akurat, dengan sistem monitoring pada perangkat mobile serta (2) Membangun fitur-fitur yang terdapat pada sistem monitoring pada perangkat *mobile* sehingga mampu meningkatkan efisiensi operasional perusahaan distribusi, serta mampu menyelesaikan masalah terkait dengan persediaan barang pada perusahaan distribusi yang tidak dapat dimonitor secara *online* dan *realtime*.

2 Landasan Teori

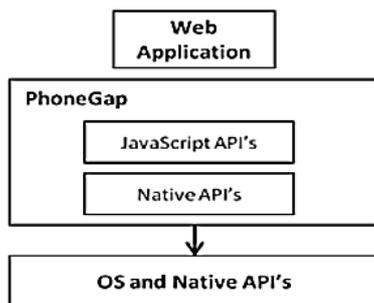
Pada penelitian digunakan beberapa landasan teori sebagai acuan pada pembuatan aplikasi *mobile*.

PhoneGap

PhoneGap adalah sebuah kerangka kerja atau *framework open source* yang dipakai untuk membuat aplikasi *cross-platform mobile* dengan HTML5, CSS, dan JavaScript. PhoneGap menjadi suatu solusi yang ideal untuk seorang pengembang web yang tertarik dalam pembuatan aplikasi di *smartphone*. dengan *framework* phonegap kita hanya melakukan satu kali *coding* langsung bisa di *compile* ke semua *platform* sekaligus.^[5]

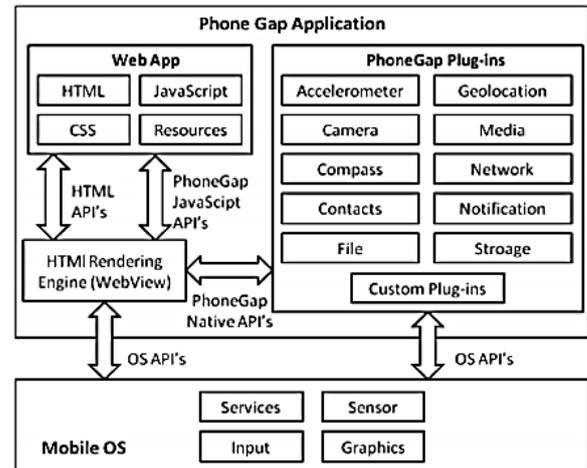
PhoneGap memberikan JavaScript API untuk pengembang yang memungkinkan akses ke fungsi perangkat canggih, seperti Accelerometer, Barcode, Bluetooth, Kalender, Kamera, Kompas, Connection, Kontak, File, GPS, Menu, NFC, dll^[6]

Arsitektur utama PhoneGap terdiri dari tiga lapisan: Aplikasi Web, PhoneGap, dan OS dan API asli. Pada Gambar 1 dijelaskan bahwa lapisan atas merupakan kode sumber aplikasi. Lapisan tengah terdiri oleh JavaScript dan API asli. Utamanya, lapisan ini bertanggung jawab untuk interfacing antara aplikasi web dan lapisan PhoneGap.^[7]



Gambar 1 Lapisan Interface Dari Arsitektur PhoneGap

Pada Gambar 2 ditunjukkan arsitektur skema yang lebih rinci yang disediakan oleh IBM. Ini mewakili semua komponen tentang aplikasi web, mesin rendering HTML, PhoneGap API dan lapisan OS. Selain itu, beberapa antarmuka yang berbeda ditunjukkan dalam detail, seperti *interfacing* antara API PhoneGap dan lapisan API asli.^[8]



Gambar 2 Skema Lengkap Arsitektur PhoneGap Dan Interfacing Antar Komponen

HTML

HTML adalah singkatan dari *Hypertext Markup Language*. Disebut *hypertext* karena di dalam HTML sebuah text biasa dapat berfungsi lain, kita dapat membuatnya menjadi *link* yang dapat berpindah dari satu halaman ke halaman lainnya dengan hanya meng-klik text tersebut. Kemampuan text inilah yang dinamakan *hypertext*, walaupun pada implementasinya nanti tidak hanya text yang dapat dijadikan *link*.

Disebut *Markup Language* karena bahasa HTML menggunakan tanda (*mark*), untuk menandai bagian-bagian dari teks. Misalnya, teks yang berada di antara tanda tertentu akan menjadi tebal, dan di antara tanda lainnya akan tampak besar. Tanda ini akan kita kenal di HTML sebagai tag.

HTML merupakan bahasa dasar pembuatan web. Disebut dasar karena dalam membuat web, jika hanya menggunakan HTML, tampilan web kita akan terasa hambar. Terdapat banyak bahasa pemrograman web yang ditujukan untuk memanipulasi kode HTML, seperti JavaScript dan PHP. Namun sebelum belajar JavaScript maupun PHP, memahami HTML merupakan hal yang paling awal.

HTML bukan bahasa pemrograman (*programming language*), tetapi bahasa *markup* (*markup language*), hal ini terdengar sedikit aneh, tapi jika telah mengenal bahasa pemrograman lain, dalam

HTML tidak akan ditemukan struktur yang biasa di temukan dalam bahasa pemrograman seperti IF, LOOP, maupun variabel. HTML hanya sebuah bahasa struktur yang fungsinya untuk menandai bagian-bagian dari sebuah halaman.^[9]

CSS

CSS adalah kumpulan kode yang digunakan untuk mendefinisikan desain dari bahasa *markup*, dimana bahasa markup ini salah satunya adalah HTML. Dengan kata lain, CSS adalah kumpulan kode program yang digunakan untuk mendesain atau mempercantik tampilan halaman HTML. Dengan CSS kita bisa mengubah desain dari teks, warna, gambar dan latar belakang dari hampir semua kode *tag* HTML.

CSS biasanya selalu dikaitkan dengan HTML, karena keduanya memang saling melengkapi. HTML ditujukan untuk membuat struktur, atau konten dari halaman web. Sedangkan CSS digunakan untuk tampilan dari halaman web tersebut. Istilahnya, "*HTML for content, CSS for Presentation*".^[9]

JavaScript

JavaScript adalah bahasa pemrograman web yang bersifat *Client Side Programming Language*. *Client Side Programming Language* adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*. Aplikasi *client* yang dimaksud merujuk kepada *web browser* seperti Google Chrome dan Mozilla Firefox. Jenis bahasa pemrograman *Client Side* berbeda dengan bahasa pemrograman *Server Side* seperti PHP, dimana untuk *server side* seluruh kode program dijalankan di sisi *server*.

Untuk menjalankan JavaScript, dibutuhkan aplikasi *text editor*, dan *web browser*. JavaScript memiliki fitur: *high-level programming language, client-side, loosely typed*, dan berorientasi objek.

JavaScript pada awal perkembangannya berfungsi untuk membuat interaksi antara pengguna dengan situs web menjadi lebih cepat tanpa harus menunggu pemrosesan di web server. Sebelum javascript, setiap interaksi dari user harus diproses oleh *web server*.

Sebelum dikembangkannya javascript, mengisi *form* registrasi untuk pendaftaran sebuah situs web, lalu men-klik tombol submit, menunggu sekitar 20 detik untuk website memproses isian form tersebut, dan mendapati halaman yang menyatakan bahwa terdapat kolom form yang masih belum diisi. Untuk keperluan seperti inilah JavaScript dikembangkan. Pemrosesan untuk mengecek apakah seluruh form telah terisi atau tidak, bisa dipindahkan dari web server ke dalam *web browser*.

Dalam perkembangan selanjutnya, JavaScript tidak hanya berguna untuk validasi *form*, namun untuk berbagai keperluan yang lebih *modern*. Berbagai animasi untuk mempercantik halaman web, fitur *chatting*, efek-efek *modern, games*, semuanya bisa dibuat menggunakan JavaScript. Akan tetapi karena sifatnya yang dijalankan di sisi *client* yakni di dalam *web browser* yang digunakan oleh pengunjung situs, pengguna sepenuhnya dapat mengontrol eksekusi JavaScript. Hampir semua *web browser* menyediakan fasilitas untuk mematikan JavaScript, atau bahkan mengubah kode JavaScript yang ada. Sehingga kita tidak bisa bergantung sepenuhnya kepada JavaScript.^[10]

Ionic

Ionic adalah *platform* yang menargetkan programmer web agar bisa membuat aplikasi mobile dengan teknologi web. Programmer web yang ingin menjadi programmer *mobile* tidak perlu belajar Java atau Objective C atau C# untuk membuat versi Aplikasi dari layanan webnya. Inilah pandangan dari pendiri Ionic. Sebelumnya, pendiri ionic ingin membuat gebrakan revolusioner dengan konsep mengubah konsep *Write one Run Anywhere* dengan satu *base code* berbasis javascript.

Ketika ide ini disampaikan ke para programmer Android atau iOS, ide ini tidak disambut hangat karena tidak semua bisa dilakukan dengan teknologi web. Salah satunya adalah pemrograman permainan di *smartphone* yang memanfaatkan OpenGL ES atau aplikasi yang sangat tergantung sensor. Dari sinilah pengembang Ionic akhirnya mempivot targetnya adalah untuk para programmer web yang ingin membuat aplikasi mobile (biasanya aplikasi bisnis) tanpa harus belajar bahasa pemrograman lain.

Ionic platform menggunakan lisensi *opensource*, programmer dapat membuat aplikasi *free* ataupun komersil dengan Ionic. Target ionic hanya untuk Android 4 dan ios 7 Keatas. Ionic hanya menyediakan *framework*-nya, untuk membungkusnya menjadi aplikasi Android atau iOS tetap menggunakan PhoneGap.^[11]

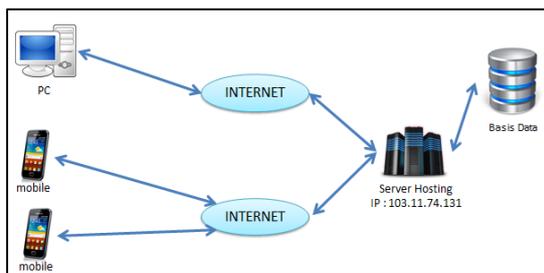
3 Analisis

Sistem perlu untuk dianalisis agar dapat memberikan gambaran umum kepada pengguna tentang sistem yang akan dibuat. Keunggulan dan kelemahan sistem dapat diketahui dan dapat dijadikan acuan tahapan evaluasi. Penelitian ini disusun berdasarkan analisis tentang tahapan dan cara membangun suatu aplikasi, untuk mengetahui informasi-informasi mengenai data yang ada pada aplikasi.

Aplikasi mobile ini diintegrasikan dengan *database* atau basis data yang menyimpan data-data mengenai informasi yang dibutuhkan dalam membangun aplikasi. Aplikasi ini dibangun berdasarkan kebutuhan-kebutuhan sebagai berikut:

1. Mampu memberikan informasi yang dibutuhkan oleh pengguna. Informasi yang dimaksud adalah informasi mengenai daftar item, daftar supplier dan persediaan barang.
2. Dapat melihat persediaan barang secara *online* dan *realtime* dari manapun.

Arsitektur sistem adalah desain sistem komputer secara keseluruhan untuk memenuhi kebutuhan-kebutuhan organisasi yang spesifik. Arsitektur sistem informasi berguna sebagai penuntun bagi operasi sekarang atau menjadi cetak-biru (*blueprint*) untuk arahan di masa mendatang. Gambar 3 menjelaskan tentang arsitektur sistem aplikasi secara umum.

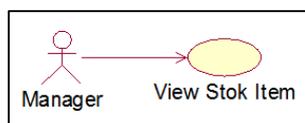


Gambar 3 Arsitektur Sistem Aplikasi

Perancangan UML

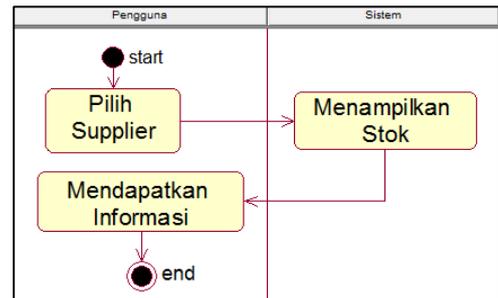
Proses pengembangan perangkat lunak membutuhkan analisis dan rancangan. Salah satu model untuk merancang pengembangan perangkat lunak yang berbasis *object oriented* adalah UML (*Unified Modelling Language*).

Diagram use case mendeskripsikan interaksi antara pengguna sistem dengan sistem itu sendiri dengan menggambarkan kebutuhan sistem dari sudut pandang di luar sistem. Diagram use case menentukan nilai yang diberikan oleh sistem kepada pemakai. Gambar 4 menjelaskan bahwa pengguna sistem monitoring pada perangkat mobile adalah manager, dan pada sistem monitoring ini pengguna dapat melihat stok item secara *realtime*.



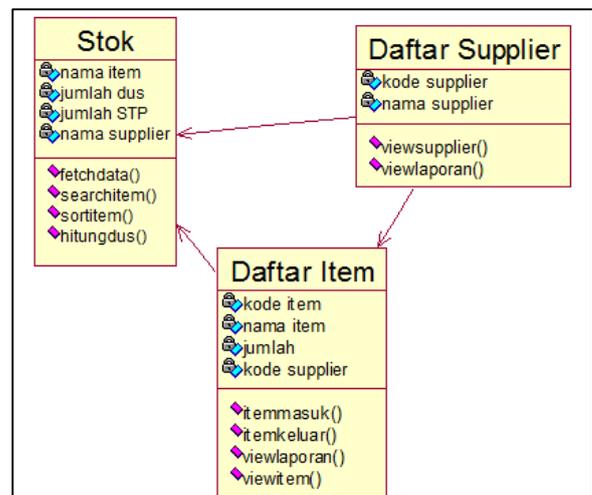
Gambar 4 Diagram Use Case

Diagram aktivitas menggambarkan berbagai aliran aktivitas dalam sistem yang sedang dirancang. Gambar 5 menjelaskan alur ketika pengguna menggunakan sistem monitoring pada perangkat *mobile*. Pengguna dapat memilih supplier dan sistem akan menampilkan persediaan barang sesuai dengan supplier yang telah dipilih, atau pengguna dapat menampilkan persediaan barang secara keseluruhan tanpa memilih supplier terlebih dahulu.



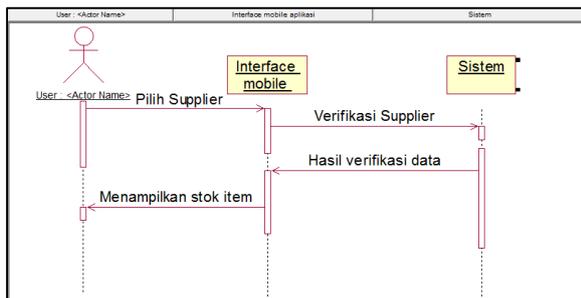
Gambar 5 Diagram Aktivitas

Diagram kelas menggambarkan kelas-kelas yang terdapat pada sistem monitoring. Gambar 6 menjelaskan tentang kelas-kelas yang ada pada sistem monitoring. Pada supplier terdapat kelas-kelas *public* yaitu kode supplier dan nama supplier serta terdapat operasi *viewsupplier* dan *viewlaporan*. Pada item terdapat kelas-kelas *public* yaitu kode item, nama item, jumlah dan kode supplier serta terdapat operasi *itemmasuk*, *itemkeluar*, *viewlaporan*, *viewitem*. Pada stok terdapat kelas-kelas *public* yaitu nama item, jumlah dus, jumlah STP, dan nama supplier serta terdapat operasi *fetchdata*, *searchitem*, *sortitem*, dan *hitungdus*.



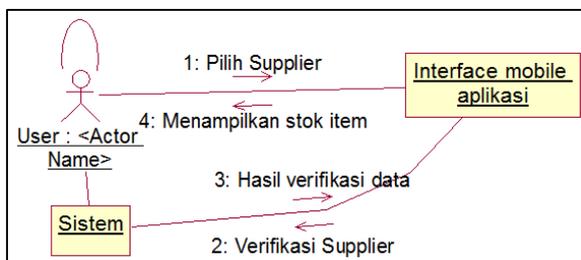
Gambar 6 Diagram Kelas

Diagram sekuensial merupakan diagram interaksi yang menekankan pada pengiriman pesan dan menggambarkan alur kerja dari fungsi-fungsi dalam sistem tersebut secara terurut. Gambar 7 menjelaskan tentang hubungan antara pengguna ke sistem monitoring apabila pengguna menjalankan sistem monitoring. Pengguna dapat memilih supplier untuk dapat menampilkan stok item dari supplier tertentu. Sistem akan melakukan verifikasi lalu data stok item ditampilkan oleh sistem.



Gambar 7 Diagram Sekuensial

Diagram kolaborasi adalah diagram yang menekankan kepada struktur aliran dari objek – objek yang mengirim dan menerima pesan. Gambar 8 menjelaskan tentang hubungan antara pengguna ke sistem monitoring pada perangkat *mobile* dan sebaliknya. Interface akan menampilkan opsi filter untuk memilih satu supplier tertentu. Setelah pengguna memilih satu buah supplier maka interface akan mengirimkan data kepada sistem, lalu sistem akan menampilkan stok item sesuai dengan supplier yang dipilih user melalui interface. Apabila pengguna tidak memilih supplier, maka sistem akan menampilkan stok item dari seluruh supplier tanpa terkecuali.



Gambar 8 Diagram Kolaborasi

Perancangan Antarmuka

Dalam pembuatan sebuah aplikasi, rancangan antarmuka merupakan bagian yang sangat penting untuk menggambarkan komunikasi antara sistem dengan pengguna. Tujuan dari merancang

antarmuka adalah untuk mempermudah pengguna dalam menggunakan sistem tersebut, serta mempermudah dalam pembuatan aplikasi itu sendiri, sehingga proses pembuatan aplikasi menjadi terstruktur dan sistematis. Dengan rancangan antarmuka yang baik dan mudah dimengerti, maka aplikasi dapat dibuat dengan lebih baik dan lebih cepat, serta informasi yang akan ditampilkan oleh sistem lebih mudah diterima oleh pengguna sehingga penggunaan aplikasi menjadi lebih efektif.



Gambar 9 Rancangan Interface Halaman Utama

Gambar 9 menjelaskan rancangan halaman utama aplikasi mobile. Halaman ini akan memberikan informasi kepada pengguna tentang persediaan barang. Pengguna dapat melakukan *filter* dengan memanfaatkan *dropdown menu* untuk memilih supplier dan kolom *search* untuk kata kunci tertentu. Di pojok kiri atas terdapat tombol *refresh* untuk melakukan penarikan data ulang dari *database* sehingga data yang ditampilkan adalah data terbaru. Di pojok kanan atas terdapat tombol *about* yang akan mengantarkan pengguna ke halaman *about*.

Selain halaman utama, akan dibangun pula *splash screen* dari aplikasi mobile yang akan muncul selama beberapa detik ketika pengguna memulai aplikasi monitoring tersebut, dan halaman *about* guna memberikan informasi kepada pengguna tentang hal yang perlu pengguna ketahui tentang aplikasi ini dan tentang perusahaan. Di bagian bawah terdapat tombol *back* yang akan mengantarkan pengguna kembali ke halaman sebelumnya.

4 Implementasi

Implementasi perancangan antarmuka pada sistem merupakan peranan yang penting, karena

pada implementasi inilah bagaimana cara pengguna menggunakan sistem aplikasi yang telah dibuat dijelaskan. Berdasarkan perancangan sistem yang dibuat, halaman sistem aplikasi *mobile* yang dibangun adalah *splash screen*, halaman monitoring, halaman *about*, dan ada penambahan halaman diagram.



Nama Item	Jumlah (DUS)	Jumlah (STP)
Senna Krupukku Uidang	61	24
Senna Krupukku Ikan	20	24
Senna Krupukku Bawang	99	18
PMS Bolu Lapis Pandan	610	1
PMS Bolu Lapis Strawberry	467	7
PMS Bolu Lapis Coklat	584	7
PMS Pia Kacang Hitam	250	0
Cleo Cup 250ml	139	0
Cleo Cup 550ml	57	0
Cleo Cup 1500ml	44	0
UBM Puffy Coklat	55	11
UBM Puffy Peanut	0	0

Gambar 10 Tampilan Halaman Utama Aplikasi Mobile

Gambar 10 menjelaskan tampilan halaman utama sistem monitoring. Sistem akan mengantarkan pengguna kepada halaman ini sesaat setelah halaman *splash screen* dimunculkan. Pada halaman ini pengguna dapat melihat persediaan barang secara *real time*, dengan rincian nama item, jumlah (DUS), dan jumlah (STP). Pengguna dapat memilih supplier tertentu atau melakukan pencarian kata kunci tertentu untuk memudahkan dalam melihat data.

Kode Program Halaman Utama Aplikasi Mobile

```
$http.get("http://stok.rjs.co.id/api/index.php")
.success(function(Data) {$scope.Data=Data.users;
console.log($scope.Data)});

$http.get("http://stok.rjs.co.id/api/sup.php")
.success(function(Supplier) {
$scope.Supplier=Supplier.users;
console.log($scope.Supplier)

$scope.search= function(query) {
console.log("searching");
$scope.searchitem = query;

$http.get("http://stok.rjs.co.id/api/search.php?search="+$scope.searchitem+"&order="+$scope.sorttoggle)
.success(function(Data) {
$scope.Data = {}; $scope.Data=Data.users;
$scope.State = "search";
$scope.StateSearch = $scope.searchitem;
```

```
$scope.sort= function() {
console.log("sorting");
if ($scope.sorttoggle=="asc") {
$scope.sorttoggle = "desc"; } else {
$scope.sorttoggle = "asc"; }
if($scope.State == ""){
$http.get("http://stok.rjs.co.id/api/sort.php?order="+$scope.sorttoggle)
.success(function(Data) {$scope.Data=Data.users;
console.log($scope.Data)}); }
else if($scope.State == "search") {
$scope.search($scope.StateSearch); }
else if($scope.State == "filter"){
$scope.filter($scope.StateFilter); }
}
$scope.filter = function(filterquery) {
console.log("searching");
$scope.filterquery = filterquery;
if($scope.filterquery == "-"){
$http.get("http://stok.rjs.co.id/api/index.php")
.success(function(Data) {$scope.Data=Data.users;
console.log($scope.Data)});
$scope.State = '';
} else {
$http.get("http://stok.rjs.co.id/api/supitem.php?filter="+$scope.filterquery+"&order="+$scope.sorttoggle)
.success(function(Data) {
$scope.Data = {};
$scope.Data=Data.users;
$scope.State = 'filter';
$scope.StateFilter =
$scope.filterquery;
console.log($scope.Data)});
$scope.refresh = function() {
console.log($scope.State);
if ($scope.State == "")
($http.get("http://stok.rjs.co.id/api/index.php")
.success(function(Data) {
$scope.Data=Data.users;
}))
else if ($scope.State == 'search')
{
$scope.search($scope.StateSearch); }
else if ($scope.State == "filter") {
$scope.filter($scope.StateFilter);
}};
```

Penggalan kode program diatas adalah kode program utama pada halaman utama sistem monitoring untuk menampilkan persediaan barang dengan perincian nama item, jumlah DUS, dan jumlah STP. Kode program juga berfungsi untuk menampilkan tombol-tombol seperti diagram, urutkan, *about*, *refresh*, dan juga menampilkan filter pilih supplier dan *search*.



Gambar 11 Tampilan Halaman Diagram Aplikasi Mobile

Gambar 11 menjelaskan tampilan halaman diagram aplikasi mobile. Halaman ini akan muncul ketika pengguna menekan tombol diagram terdapat pada halaman aplikasi mobile.

Kode Program Halaman Diagram Sistem Monitoring

```

$scope.showchart = function() {
    $scope.labels = [];
    $scope.datas = [];
    $scope.y = [];
    for(var i = 0 ; i <
    $scope.Data.length; i++){
    $scope.labels.push($scope.Data[i].nama);
    $scope.y.push(Math.floor($scope.Data[i].jumlah/
    $scope.Data[i].isi));
    }
    $scope.datas.push($scope.y);
    $scope.series = ['Series A'];
    $scope.closechart = function() {
    $scope.chart.hide();
    }
    }
    
```

Penggalan kode program diatas adalah kode program utama pada halaman diagram sistem monitoring untuk menampilkan persediaan barang dalam bentuk diagram.

Pengujian Dengan Data Sebenarnya

Pengujian sistem dilakukan untuk mengetahui seberapa baik dan sesuai sistem yang dibuat. Pengujian sistem dilakukan dengan memasukkan data sebenarnya dari perusahaan distribusi. Data diinputkan pada sistem berbasis web yang mendukung aplikasi mobile untuk menampilkan data.

Tabel 1 Data Supplier dan Item
Supplier : SUP002 - PT. Stanli Tri Jaya

Kode Item	Nama Item	STP
80201	PMS Bolu Lapis Pandan	8
80202	PMS Bolu Lapis Strawberry	8
80203	PMS Bolu Lapis Coklat	8
80204	PMS Pia Kacang Hitam	10

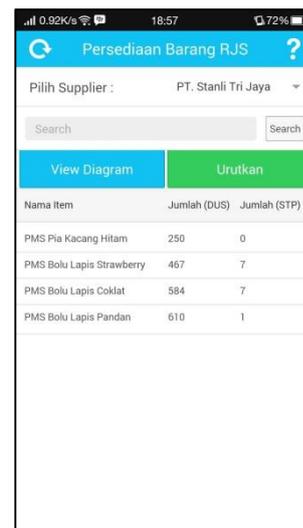
Tabel 2 Data Persediaan Barang
Supplier : SUP002 - PT. Stanli Tri Jaya

Kode Item	Jumlah DUS	Jumlah STP
80201	300	1
80202	321	3
80203	306	5
80204	44	6

Tabel 3 Transaksi Item Keluar
Supplier : SUP002 - PT. Stanli Tri Jaya

Kode Item	Jumlah DUS	Jumlah STP
80201	39	6
80202	31	7
80203	66	3
80204	2	5

Tabel 1, 2, dan 3 menjelaskan data yang diinputkan pada sistem berbasis web yang akan dimunculkan oleh sistem aplikasi mobile. Setelah data diinputkan, maka sistem aplikasi mobile akan menampilkan data persediaan barang, dalam bentuk list dan grafik.



Gambar 12 Sistem Aplikasi Menampilkan Data

Gambar 12 menjelaskan tampilan sistem aplikasi setelah data diinputkan. Dapat dilihat bahwa sistem aplikasi dapat menampilkan data sesuai dengan yang diinputkan.



Gambar 13 Sistem Aplikasi Menampilkan Diagram

Gambar 13 menjelaskan tampilan diagram sistem aplikasi setelah data diinputkan. Dapat dilihat bahwa sistem aplikasi dapat menampilkan data persediaan barang sesuai dengan data yang ada.

5 Kesimpulan

Dari hasil analisis terhadap masalah dan aplikasi yang telah dikembangkan, dapat ditarik kesimpulan sebagai berikut:

1. Aplikasi *mobile* secara umum dapat memenuhi kebutuhan pada perusahaan distribusi.
2. Aplikasi *mobile* dapat menampilkan persediaan barang secara *online* dan *realtime* dengan tepat.
3. Fitur yang terdapat pada aplikasi *mobile* secara umum dapat digunakan dengan baik sehingga mampu meningkatkan efisiensi operasional.
4. Cara penggunaan aplikasi yang mudah dipahami dan tampilan aplikasi cukup menarik membuat pengguna awam mudah mengoperasikannya.
5. Proses transaksi barang masuk dan barang keluar menjadi bagian terpenting dari aplikasi *mobile* karena data yang ditampilkan oleh aplikasi adalah hasil dari kegiatan transaksi barang masuk dan keluar.

6 Daftar Pustaka

- [1] Avinash S, Anandkumar P, "To Study and Design a Cross-Platform Mobile Application for Student Information System using PhoneGap

Framework", www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 3, Issue 9, September 2013).

- [2] Weifeng S, Maofa W, "A Study on Educational Administration Inquire System based on Mobile Device", CSAE, 2012 IEEE International Conference, Volume 3: Page(s): 392- 395.
- [3] P. Douangboupha, "SmartPhone Platform Comparison", Whitepaper Series, R2integrated, (2009).
- [4] Wen-Hann Wang, CV Vick, Christos Georgiopoulos, Jon Bork, "Cross-Platform Multi-Network Mobile Application Architecture"
- [5] Myer, Thomas, 2012. Beginning PhoneGap. Indianapolis : John Wiley & Sons, Inc.
- [6] <http://phonegap.com/about/feature/>
- [7] Manuel P, Inderjeet Singh, Antonio Cicchetti, "Comparison of Cross-Platform Mobile Development Tools" 2012 16th International Conference on Intelligence in Next Generation Networks.
- [8] <http://www.slideshare.net/drbaac/phonegap-day-ibm-phonegap-andthe-enterprise>.
- [9] Ariona, Rian, 2013. Belajar HTML & CSS "Tutorial Fundamental dalam Mempelajari HTML & CSS". diakses tanggal 13 Oktober 2015, <http://www.ariona.net/ebook-belajar-html-dan-css/>
- [10] Haverbeke, Marijn, 2014. Eloquent JavaScript, A Modern Introduction to Programming. Diakses tanggal 25 Oktober 2015, http://eloquentjavascript.net/Eloquent_JavaScript.p
- [11] Ravulavaru, Arvind, 2015. Learning Ionic. Birmingham, UK : Packt Publishing Ltd.