

Implementasi Scrum Model Development Pada Monitoring Inventory Control Cleaning Equipment dan Chemical PT. Explore Global Solution

Meta Amalya Dewi¹, Rani Andriani²

¹Program Studi Sistem Informasi, School of Engineering Tanri Abeng University Jakarta

²Jurusan Sistem Informasi STMIK Raharja Tangerang

meta.amalya@tau.ac.id

Diterima 31 Oktober 2017

Disetujui 20 Desember 2017

Abstract— The rapid technological advances have driven many service companies to take advantage of technology to improve their services. PT. Explore Global Solution, a cleaning service company with a manual hygiene control monitoring system through record keeping using ledgers and forms provided by the company, so that the company's activities have not run optimally, it is necessary to design the information system with the availability of time for 10 weeks. This research is done by using the method of exploration and application development with scrum model. The end result of this research is web-based applications that can improve operational performance, especially logistics division, with completion time of 373 hours or equivalent to 5 working days 8 hours per day is 9 weeks 3 days.

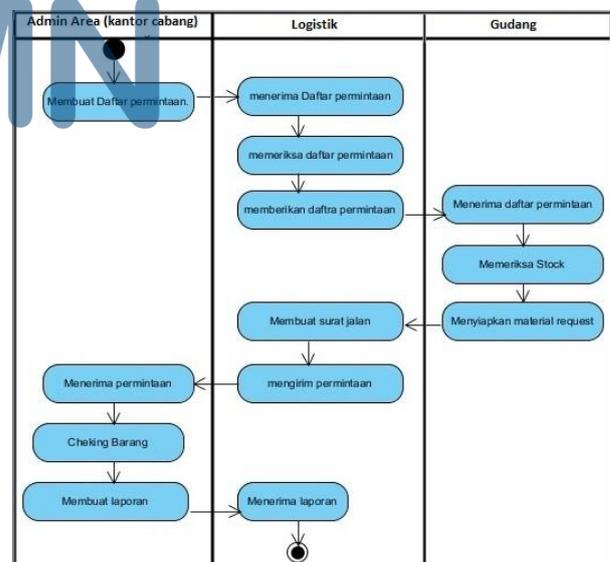
Index Terms—stock, request, backlog, sprint

I. PENDAHULUAN

Instansi pemerintah dan organisasi bisnis membutuhkan jasa *cleaning service* untuk menjaga kebersihan demi kesehatan dan kenyamanan ruangan, baik ruang kerja maupun ruang pelayanan publik. PT. Explore Global Solution merupakan perusahaan penyedia jasa *cleaning service* yang senantiasa terus menjaga kepercayaan konsumen, untuk mendapatkan loyalitas pelanggan dengan memberikan pelayanan terbaiknya [1].

Dalam menjalankan proses bisnisnya, perusahaan membutuhkan *monitoring stock control cleaning equipment* dan *chemical* untuk memudahkan proses permintaan dan pendataan barang sebagai bagian dari upaya sistemik dalam menetapkan standar kinerja perencanaan [2], apalagi pada saat ini pelanggan perusahaan sudah semakin banyak, namun sistem informasi yang berjalan saat ini masih jauh dari ideal, prosedur sistemnya *relative* membutuhkan waktu lama, selain panjangnya proses yang harus dijalani (dapat dilihat pada gambar 1 di bawah ini) juga banyaknya isian dokumen yang harus dilengkapi.

Karena belum adanya sistem kendali inventory, maka proses *stock control cleaning equipment* dan *chemical* masih menjadi kendala utama, yang menyebabkan terjadinya keterlambatan permintaan, keterlambatan pengiriman, serta penumpukan *cleaning equipment* dan *chemical* yang tidak terpakai. Dengan sistem pencatatan yang manual menggunakan berbagai macam formulir, kertas, buku dan proses input rekapitulasi data menggunakan software *spread sheet* menyebabkan sulitnya mencari data-data yang diperlukan, kehilangan data karena tercecernya formulir, serta staf yang tidak memiliki kemampuan mengolah data dengan berbagai rumus dari aplikasi *spread sheet*.



Gambar 1. Activity Diagram *Monitoring Inventory Control*

Oleh sebab itu, perlu dilakukan pengembangan sistem *monitoring stock control* sebagai kendali persediaan barang yang dapat mengatasi berbagai

kendala dan permasalahan yang terjadi dengan memanfaatkan kemajuan teknologi yang ada, yaitu membangun aplikasi berbasis web yang dapat memberikan informasi yang dibutuhkan.

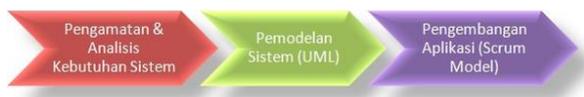
Permasalahannya adalah ketersediaan waktu yang sangat terbatas, perusahaan hanya memberikan waktu selama kurang lebih 10 minggu mulai dari perencanaan awal hingga penyelesaian aplikasi dengan potensi banyak perubahan seiring proses pengembangan sistem, maka pemilihan model pengembangan sistem menjadi penentu berhasil tidaknya perancangan aplikasi sistem informasi ini.

Jika menggunakan model waterfall yang prosesnya dilakukan secara terurut mulai analisis, desain, code dan test, metode ini memiliki sifat kaku, sulit melakukan perubahan di tengah proses sehingga ketika terdapat kekurangan proses atau prosedur dari tahap sebelumnya, maka tahapan pengembangan harus dilakukan mulai dari awal lagi, maka metode ini akan memakan waktu yang lama [3]. Alternatif kedua menggunakan model prototype yang mampu melibatkan user secara baik dalam prosesnya serta dapat mempersingkat waktu pengembangan namun masih memiliki kekurangan dalam hal dokumentasi, kurang fleksible dalam menghadapi perubahan, dan prototype yang dihasilkan tidak selamanya mudah dirubah [4].

Berdasarkan kelemahan pada model waterfall dan prototype, maka penulis memilih pengembangan sistem menggunakan *scrum model* agar aplikasi yang dihasilkan dapat disesuaikan dengan lingkungan, dengan ukuran tim yang kecil dan banyak perubahan seiring proses pengembangan sistem [4] serta dapat meningkatkan kecepatan waktu pengembangan software [5].

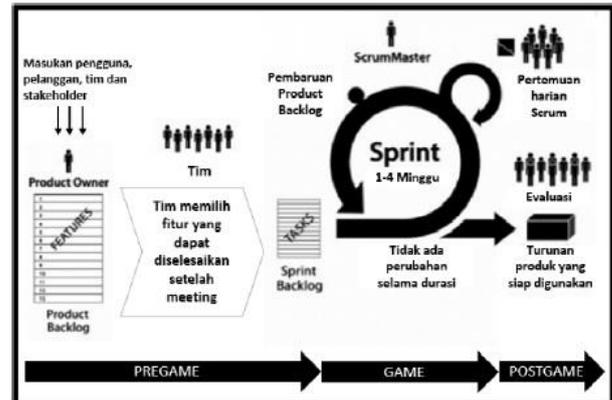
II. METODE

Alur dari pelaksanaan penelitian ini dapat dilihat pada gambar 2 berikut :



Gambar 2. Alur Penelitian

Penelitian ini dilaksanakan dengan menggunakan desain riset eksploratori melalui pengamatan dan analisis terhadap kebutuhan sistem dan perancangan berorientasi pada objek dengan tools UML, yaitu sebuah bahasa pemodelan untuk perangkat lunak yang berparadigma 'berorientasi objek', sementara pemodelan digunakan untuk penyederhanaan permasalahan kompleks sedemikian rupa sehingga lebih mudah dipahami [6], pemodelan ditulis dengan menggunakan diagram activity dan diagram usecase, kemudian pembangunan aplikasinya menggunakan *scrum model* melalui tahapan sebagai berikut :



Gambar 3. Tahapan dan Pihak yang terlibat dalam Scrum [7]

1. Backlog

Backlog adalah daftar kebutuhan atau fitur yang memberikan nilai bisnis klien, dan produk *backlog* yang dibuat dapat bertambah.

2. Sprints

Sprints adalah unit pekerjaan yang diperlukan untuk memenuhi kebutuhan yang ditetapkan dalam *backlog* sesuai dengan waktu yang ditetapkan dalam time-box. Selama proses ini berlangsung *backlog* tidak ada penambahan.

3. Scrum Meetings

Aktivitas *scrum meetings* merupakan pertemuan rutin yang dilakukan untuk mengevaluasi apa yang telah dikerjakan, hambatan yang ada, dan target penyelesaian untuk bahan meeting selanjutnya.

4. Demo

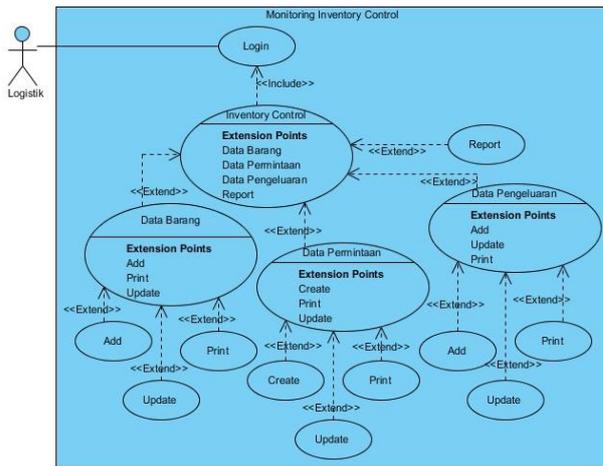
Penulisan coding aplikasi menggunakan bahasa pemrograman Java, Netbeans 7 untuk desain web, MySQL sebagai database dan Xampp. Demo progress aplikasi dilakukan untuk memberikan peningkatan perangkat lunak kepada klien yang telah diimplementasikan sehingga dapat ditunjukkan dan dievaluasi oleh klien.

III. HASIL

A. Perancangan Usecase Diagram

Dalam pengembangan aplikasi sistem informasi ini, rancangan proses digambarkan dengan menggunakan usecase diagram untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut [8]

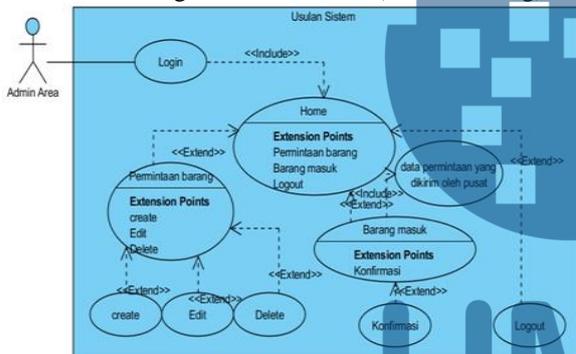
1. Usecase Diagram Divisi Logistik



Gambar 4. Usecase Diagram Divisi Logistik

Berdasarkan gambar di atas dapat diketahui bahwa divisi logistik berinteraksi dengan sistem melalui login, jika pengisian login benar maka bisa berinteraksi dengan menu data barang, data permintaan dan data pengeluaran, serta menu report.

2. Usecase Diagram Admin Area (kantor cabang)



Gambar 5. Usecase Diagram Admin Area (kantor cabang)

Pada gambar di atas merupakan interaksi yang dapat dilakukan oleh admin Area (kantor cabang) dengan sistem melalui login, dan jika pengisian login benar admin Area dapat mengelola permintaan barang dan mengelola barang masuk yang dikirimkan oleh divisi logistik kantor pusat.

B. Tahapan Scrum Model

Dari permasalahan yang ada, selanjutnya dirancang aplikasi untuk *monitoring inventori control cleaning equipment* dan *chemical* yang memiliki kebutuhan fungsional diantaranya:

1. Sistem harus dapat mengelola data barang *cleaning equipment* dan *chemical*
2. Sistem harus dapat mengelola data permintaan barang dari Area (kantor cabang)
3. Sistem harus dapat mengelola data pengeluaran barang

4. Sistem harus dapat mengontrol stok persediaan barang yang tersedia di logistik

1. Menentukan *Product Backlog*

Pada tahapan ini, ditentukan semua *user requirement* (kebutuhan sistem) baik yang bersifat fungsional maupun nonfungsional beserta skala prioritasnya, berikut ini merupakan daftar fitur yang akan dikembangkan :

Tabel 1. *Product Backlog*

| No | Deskripsi Fitur | Prioritas | Estimasi Waktu (Jam) |
|----|---|---------------|----------------------|
| 1. | Mengelola data barang <i>cleaning equipment</i> dan <i>chemical</i> | Tinggi | 49 |
| 2. | Mengelola data permintaan barang dari Area (kantor cabang) | Tinggi | 87 |
| 3. | Mengelola data pengeluaran barang | Tinggi | 56 |
| 4. | Monitoring permintaan barang dari Area (kantor cabang) | Sangat Tinggi | 63 |
| 5. | Monitoring pengeluaran barang | Sangat Tinggi | 51 |
| 6. | Monitoring Stok persediaan barang yang tersedia di logistik | Sangat Tinggi | 67 |

2. Menentukan Sprint

Di tahap ini dilakukan *breakdown* dari *product backlog* pada tabel 1 di atas menjadi periode waktu penyelesaian proyek. Terdapat 6 *sprints* yang harus dilakukan, diantaranya :

Tabel 2. *Sprint* 1 Pengelolaan Data Barang

| Fitur Backlog | Task | Estimasi Waktu (Jam) |
|---|---|----------------------|
| Mengelola data barang <i>cleaning equipment</i> dan <i>chemical</i> | Divisi Logistik dapat menambah data item barang | 15 |
| | Divisi Logistik dapat menambah kategori barang | 10 |
| | Divisi logistik dapat merubah data item barang (edit dan hapus) | 12 |
| | Divisi logistik dapat merubah data kategori barang (edit dan hapus) | 12 |
| | TOTAL | 49 |

Sprint 1 terdiri dari 4 *task* dan memiliki total estimasi waktu sebanyak 49 jam. Setelah *sprint* 1 selesai dilaksanakan dilanjutkan dengan *sprint* 2 yang dapat dilihat pada tabel berikut di bawah ini.

Tabel 3. *Sprint* 2 Pengelolaan Permintaan Barang

| Fitur Backlog | Task | Estimasi Waktu (Jam) |
|--|--|----------------------|
| Mengelola data permintaan barang dari Area (kantor cabang) | Admin Area (kantor cabang) dapat melakukan permintaan barang | 14 |
| | Admin Area (kantor cabang) dapat menambah permintaan barang | 12 |
| | Divisi Logistik dapat melihat data permintaan barang dari Area (kantor cabang) | 13 |
| | Divisi Logistik dapat melihat detail permintaan barang dari Area (kantor cabang) | 13 |
| | Divisi logistik dapat memberi approve terhadap permintaan barang yang masuk | 11 |
| | Divisi logistik dapat memberikan catatan kondisional terhadap permintaan barang | 13 |
| | Divisi logistik dapat melakukan aksi melakukan perubahan terhadap permintaan barang (edit dan hapus) | 11 |
| | TOTAL | 87 |

Sprint 2 ini terdiri dari 7 *task* dan memiliki total estimasi waktu sebanyak 87 jam dan *product backlog* pada print ini fokus pada kebutuhan Area (kantor cabang) untuk dapat melakukan permintaan barang, setelah selesai dilanjut mengerjakan *sprint 3* pada tabel di bawah ini.

Tabel 4. *Sprint 3* Pengelolaan Pengeluaran Barang

| Fitur Backlog | Task | Estimasi Waktu (Jam) |
|-----------------------------------|--|----------------------|
| Mengelola data pengeluaran barang | Divisi Logistik dapat menginput data pengiriman barang | 13 |
| | Divisi Logistik dapat menambah data pengiriman barang | 14 |
| | Divisi Logistik mengubah data pengiriman barang | 13 |
| | Admin Area (kantor cabang) dapat melihat penerimaan barang masuk | 16 |
| TOTAL | | 56 |

Sprint 3 terdiri dari 4 *task* dan memiliki total estimasi waktu sebanyak 56 jam dan *product backlog* pada *sprint* ini difokuskan pada pengiriman barang berdasarkan permintaan yang telah di *approve*, berikutnya adalah *sprint 4* yang dapat dilihat pada tabel 5 di bawah ini.

Tabel 5. *Sprint 4* Monitoring Permintaan Barang

| Fitur Backlog | Task | Estimasi Waktu (Jam) |
|--|---|----------------------|
| Monitoring permintaan barang dari Area (kantor cabang) | Pimpinan dapat melihat laporan permintaan per-periode | 15 |
| | Pimpinan dapat melihat grafik permintaan barang | 18 |
| | Pimpinan dapat mengetahui barang apa yang paling sering diminta | 14 |
| | Pimpinan dapat mengetahui Area (kantor cabang) yang memiliki kebutuhan lebih banyak | 16 |
| | TOTAL | 63 |

Sprint 4 memiliki 4 *task* dengan total estimasi waktu sebanyak 63 jam. *Product backlog* pada *sprint 4* difokuskan pada kebutuhan monitoring permintaan barang dari Area (kantor cabang)

Tabel 6. *Sprint 5* Monitoring Pengeluaran Barang

| Fitur Backlog | Task | Estimasi Waktu (Jam) |
|-------------------------------|---|----------------------|
| Monitoring pengeluaran barang | Pimpinan dapat melihat laporan pengeluaran barang per-periode | 18 |
| | Pimpinan dapat melihat grafik pengeluaran barang | 18 |
| | Pimpinan dapat mengetahui persentase efektifitas pengeluaran barang | 15 |
| | TOTAL | 51 |

Sprint 5 memiliki 3 *task* dengan total estimasi waktu sebanyak 51 jam dan berikutnya adalah *sprint 6* yang memiliki 4 *task* dengan total estimasi waktu sebanyak 67 jam pada tabel 7, *product backlog* yang dikerjakan antara lain :

Tabel 7. *Sprint 6* Monitoring Stok Persediaan Barang

| Fitur Backlog | Task | Estimasi Waktu (Jam) |
|---|---|----------------------|
| Monitoring Stok persediaan barang yang tersedia di logistik | Pimpinan dapat mengecek tingkat persediaan (stock level) | 18 |
| | Pimpinan dapat mengontrol akurasi data persediaan | 15 |
| | Pimpinan dapat menganalisa jumlah dan jenis persediaan barang | 17 |
| | Pimpinan dapat menganalisa tingkat persediaan yang sehat | 17 |
| | TOTAL | 67 |

3. Melakukan Scrum Meeting

Aktifitas berikutnya yang menjadi bagian dari tahapan *scrum model* adalah melaksanakan *scrum meetings* yang dijadwalkan rutin perhari pada pukul 16.00 hingga pukul 17.30. Agenda meeting harian ini dilakukan dalam rangka mengevaluasi pekerjaan yang telah diselesaikan, kendala yang dihadapi serta solusi dan target penyelesaian sebagai bahan pertemuan berikutnya.

4. Demo

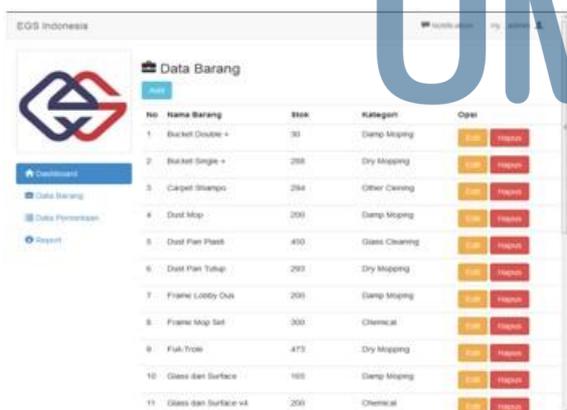
Tahapan ke 4 adalah demo yang dilaksanakan untuk memberi informasi kepada klien peningkatan dari perangkat lunak yang telah dihasilkan dan dapat langsung dievaluasi oleh klien.

Hasil tampilan dari pembangunan sistem informasi *monitoring inventory control*, dapat dijelaskan dimulai dari halaman utama aplikasi seperti gambar berikut :



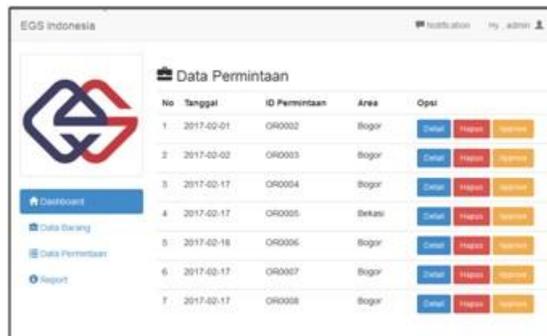
Gambar 6. Halaman Utama Aplikasi

Gambar 6 di atas merupakan tampilan utama dari aplikasi setelah user berhasil memasukkan sandi dengan benar pada halaman login, dan selanjutnya pengelolaan data barang dapat dilihat pada gambar di bawah ini :



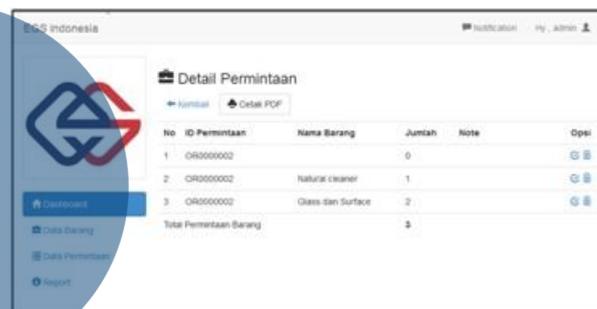
Gambar 7. Menu Master Data Barang

Pada menu pengelolaan data barang terdiri dari nama barang, kategori barang, informasi ketersediaan barang dan terdapat opsi atau pilihan aksi edit dan hapus. Untuk pengelolaan permintaan barang dapat dilihat pada gambar 7.



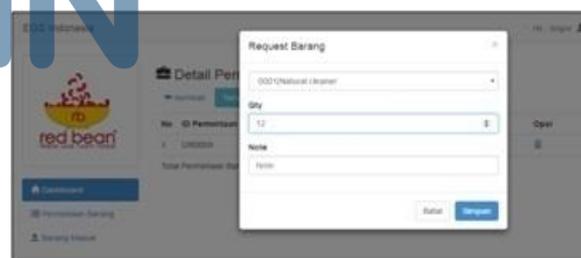
Gambar 8. Menu Data Permintaan Barang

Di menu data permintaan dapat dilihat tanggal permintaan, id permintaan, Area (kantor cabang) yang melakukan permintaan serta opsi untuk melihat detail permintaan, menghapus dan melakukan persetujuan (*approve*) terhadap permintaan. Halaman detail permintaan dapat dilihat pada gambar 8.



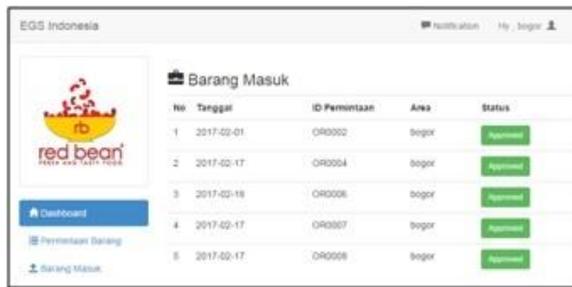
Gambar 9. Halaman Detail Permintaan

Area (kantor cabang) dapat melakukan permintaan barang melalui halaman permintaan barang seperti pada gambar 9 berikut :



Gambar 10. Halaman Permintaan Barang Area (kantor cabang)

Jika permintaan barang disetujui (*approve*) dan telah dikirimkan, maka Area (kantor cabang) dapat menginput data penerimaan barang di halaman barang masuk seperti gambar di bawah ini :



| No | Tanggal | ID Permintaan | Area | Status |
|----|------------|---------------|-------|----------|
| 1 | 2017-02-01 | ORD002 | logor | Approved |
| 2 | 2017-02-17 | ORD004 | logor | Approved |
| 3 | 2017-02-18 | ORD006 | logor | Approved |
| 4 | 2017-02-17 | ORD007 | logor | Approved |
| 5 | 2017-02-17 | ORD008 | logor | Approved |

Gambar 11. Halaman Penerimaan Barang Area (kantor cabang)

Berikutnya adalah tampilan laporan (*Report*) sebagai informasi untuk memudahkan proses monitoring persediaan barang yang dapat dilihat pada gambar 10 dan 11 di bawah ini.



Gambar 12. Tampilan Report Stok Periodik



Gambar 13. Tampilan Stok Level Report

Gambar 12 merupakan tampilan grafik persediaan barang periodik yang dapat dipilih harian atau mingguan dan bulanan di setiap Area (kantor cabang). Gambar 13 adalah tampilan laporan stok level di setiap Area (kantor cabang).

IV. SIMPULAN

Adapun kesimpulan dari penelitian ini adalah sebagai berikut :

1. Penerapan scrum model dapat menghasilkan aplikasi *monitoring inventory control cleaning equipment* dan *chemical* yang terdiri dari 6 *backlog*, 6 *sprint*, *scrum meeting* harian dan demo aplikasi yang sangat menunjang dalam perancangan aplikasi dengan waktu penyelesaian

selama 373 jam atau disamakan dengan 5 hari kerja 8 jam perhari adalah 9 minggu 3 hari.

2. Hasil rancangan aplikasi *monitoring inventory control cleaning* dan *chemical* ini dapat memudahkan Area (kantor cabang) dalam melakukan proses permintaan barang, membantu Divisi Logistik dalam mengelola dan mengontrol persediaan barang serta menunjang aktivitas pimpinan dalam monitoring persediaan barang.

DAFTAR PUSTAKA

- [1] R. Chumpitaz dan N. Paparoidamis, "Service quality, relationship satisfaction, trust, commitment and business to business loyalty," *Journal of Commerce*, 2005, hal. 30-35.
- [2] N.I. Widiastuti dan R. Susanto, "Kajian sistem monitoring dokumen akreditasi teknik informatika Unikom", *Majalah Ilmiah Unikom*, Vol.12 No. 2, 2014
- [3] R.S. Pressman, "Rekayasa Perangkat Lunak: Pendekatan Praktisi Buku I". Andi Offset: Yogyakarta, 2015
- [4] F. Mubarak, Harliana, dan I. Hadijah, "Perbandingan Antara Metode RUP dan Prototype Dalam Aplikasi Penerimaan Siswa Baru Berbasis Web", *Citec Journal*, Vol. 2 No. 2, 2015, hal. 114-127.
- [5] K. Made, "Implementasi Metodologi Scrum dalam Pembangunan Situs Harga Komoditas", *Jurnal Sistem Informasi*, Vol. 9 No. 2, hal. 149 – 160, 2014.
- [6] R.A. Azdy dan S.N. Azhari, "Implementasi Scrum pada pengembangan Software Terdistribusi, Seminar Nasional Informatika 2012 (semnasIF 2012) ISSN: 1979-2328 UPN "Veteran" Yogyakarta, 2012.
- [7] A. Nugroho, "Rekayasa Perangkat Lunak Menggunakan UML dan Java," Andi Offset: Yogyakarta, 2010, hal. 6.
- [8] K. Schwaber, "Agile Project Management with Scrum", Microsoft Press, 2004.
- [9] H. Ade, "Pemodelan UML Sistem Informasi Monitoring Penjualan dan Stok Barang (Studi Kasus: Distro Zhezha Pontianak), *Jurnal Khatulistiwa Informatika*, Vol. IV, No. 2, 2016.