



## Rancang Bangun Aplikasi Monitoring Pembuangan dan Pengumpulan Scrap Produksi Berbasis Web

Dwi Amalia Purnamasari<sup>1</sup>, Mutiara Samsida<sup>2</sup>, Yeni Rokhayati<sup>3,\*</sup>  
<sup>1,2,3</sup>Jurusan Teknik Informatika, Politeknik Negeri Batam, Batam  
<sup>1,2,3</sup>Jln. Ahmad Yani Batam Center, Kota Batam, 29464, Indonesia  
\*Corresponding Author: [yeni@polibatam.ac.id](mailto:yeni@polibatam.ac.id)

**Abstract**— Companies, especially in the manufacturing sector, are changing the way of manual production towards digitalization in accordance with the times in the era of globalization 4.0. PT. XYZ is one of the manufacturing companies in Batam that want to manage their scrap disposal to monitor data or waste disposal and collection, and also optimize the benefit. Waste disposal is carried out by the operators, while the collection is executed by the storemen. The problem is the historical data could not be archived or stored for review in order to become a report for management. So, a monitoring application is needed. This application is decided to utilize IoT technology to get real-time data. This application is web-based using PHP programming and MySQL database by utilizing the Codeigniter 4 framework. The development follows the Waterfall method that consists of requirement analysis, system design, implementation, system testing, and operation and maintenance. Testing was done by using the black box method which is used to determine the functionality of the system. As the result, this application can be implemented in PT XYZ and simplify the managing and monitoring the scrap disposal.

**Keywords**— Digitalization, IoT, Monitoring Application, Real-Time Data, *Scrap Disposal*

**Intisari**— Perusahaan khususnya di sektor manufaktur mengubah cara produksi manual menuju digitalisasi sesuai dengan perkembangan zaman di era globalisasi 4.0. PT. XYZ adalah salah satu perusahaan manufaktur di Batam yang ingin mengelola pembuangan sampah mereka untuk memantau data atau pembuangan dan pengumpulan sampah, dan juga mengoptimalkan manfaatnya. Pembuangan sampah dilakukan oleh operator, sedangkan pengumpulan dilakukan oleh petugas gudang. Masalahnya adalah data historis tidak dapat diarsipkan atau disimpan untuk ditinjau agar menjadi laporan bagi manajemen. Oleh karena itu, diperlukan sebuah aplikasi monitoring. Aplikasi ini diputuskan untuk memanfaatkan teknologi IoT untuk mendapatkan data secara real-time. Aplikasi ini berbasis web dengan menggunakan pemrograman PHP dan database MySQL dengan memanfaatkan framework Codeigniter 4. Pengembangan mengikuti metode Waterfall yang terdiri dari analisis kebutuhan, perancangan sistem, implementasi, pengujian sistem, serta operasi dan pemeliharaan. Pengujian dilakukan dengan menggunakan metode black box yang digunakan untuk mengetahui fungsionalitas sistem. Hasilnya, aplikasi ini dapat diimplementasikan di PT XYZ dan mempermudah pengelolaan dan pemantauan pembuangan scrap.

**Kata kunci**— Digitalisasi, IoT, Aplikasi Monitoring, Data Real-Time, Pembuangan Scrap

## I. PENDAHULUAN

Perusahaan khususnya di bidang manufacturing berlomba untuk mengubah cara produksi manual menuju digitalisasi sesuai dengan perkembangan zaman di era globalisasi 4.0 [1]. Secara umum, Industri 4.0 merepresentasikan tren peningkatan teknologi dan otomatisasi proses serta pertukaran data di industri manufaktur. Tren tersebut antara lain *Internet of Things* (IoT), *Industrial IoT*, *Cyber Physical Systems* (CPS), *Artificial Intelligence* (AI), *Smart Factory*, dan *Cloud Computing Systems* [2], [3].

Penerapan Industri 4.0 menjadi peluang untuk merevitalisasi sektor manufaktur di tanah air dan menghasilkan produk berkualitas secara lebih efisien. Oleh karena itu, pemerintah sebagai pengambil kebijakan akan terus mendorong dan mendukung kebutuhan riil sektor-sektor prioritas dalam adopsi teknologi Industri 4.0 secara optimal.[4]

Salah satu perusahaan manufacturing yang ada di kota Batam dijadikan sebagai tempat studi kasus dalam rancang bangun aplikasi, dimana perusahaan ini telah menerapkan *smart facturing* dalam revolusi industri 4.0. Hal ini menjadi acuan karena masalah kompleks yang terjadi di internal perusahaan *manufacturing*. Dengan adanya *section smart factory* [5] sebagai spesialis *upgrade system* yang ada di perusahaan [6].

Perusahaan ini mempunyai 4 departemen produksi yang setiap harinya memproduksi dan menghasilkan barang *reject* atau *scrap*. Produk *scrap* adalah produk yang dalam keadaan rusak atau tidak memenuhi standar kualitas yang ditetapkan [7], tidak dapat diperbaiki, namun secara ekonomis masih bisa menjadi produk yang bagus [8]. Jika produk ditolak, maka akan mengakibatkan kerugian bisnis, tetapi perusahaan dapat memasukkan *scrap* tersebut ke dalam pendapatan perusahaan melalui penjualan oleh vendor.

Namun, masalahnya perusahaan ini masih menggunakan metode manual untuk proses pencatatan, pengumpulan dan penjualan *scrap* yang tidak dapat dimonitoring secara detail dan rinci setiap waktunya. Tempat pengumpulan

barang *reject* disebut sebagai *scrap bin* dimana *scrap bin* tersebut tidak selalu dimonitoring setiap waktunya oleh *operator* dan *storeman*. Oleh karena itu, diperlukan sebuah aplikasi yang dapat menyelesaikan masalah manual tersebut menggunakan teknologi 4.0 secara digitalisasi dengan tujuan datanya dapat terekam oleh sistem secara real-time, sehingga tidak ada lagi pencatatan secara manual dan dapat dimonitoring melalui satu aplikasi web.

Dengan adanya ini diharapkan dapat menyelesaikan permasalahan tersebut, didukung dengan adanya teknologi IoT yang terhubung dengan jaringan perusahaan yang memiliki efisiensi waktu yang singkat sehingga dapat merekam data secara real-time dan dapat dimonitoring di aplikasi web ini.

## II. TINJAUAN PUSTAKA

Penelitian [9] membuat sistem monitoring keluhan sampah kota Makassar yang mana menggunakan model pengembangan *Progressive Web Application* (PWA) dengan API sebagai sumber data dan menghasilkan sebuah sistem yang memungkinkan untuk menyimpan file dalam konten situs web. Sedangkan penelitian [10] membangun sistem monitoring tempat sampah berbasis IoT menggunakan protokol MQTT dengan metode penelitian yaitu menggunakan protokol MQTT yang berperan sebagai pertukaran data dan sebuah sensor pengukur jarak jauh menggunakan sensor ultrasonic Hy-SRF05 dan menghasilkan tempat sampah pintar yang mampu mengontrol dan mengirimkan data saat tempat sampah sudah mengalami kepenuhan. Lalu, penelitian [11] membuat sistem pemantauan *volume* dan gas buang menggunakan metode RTOS, yaitu sistem operasi *real-time*.

Mengadopsi beberapa hasil penelitian sebelumnya yang telah menerapkannya pada pengelolaan sampah, maka penelitian ini memutuskan untuk menerapkan *real-time* kondisi yang bertujuan mendapatkan data histori secara realtime dalam hal pembuangan dan pengumpulan *scrap*. Menggunakan Raspberry Pi 4, *load cell*, dan sensor *lock door*, dan *traffic lamp*, data secara real-time dikirim ke aplikasi monitoring yang dibangun pada penelitian ini.

### III. METODE PENELITIAN

Paragraf harus teratur. Semua paragraf harus rata, yaitu sama-sama rata kiri dan dan rata kanan.

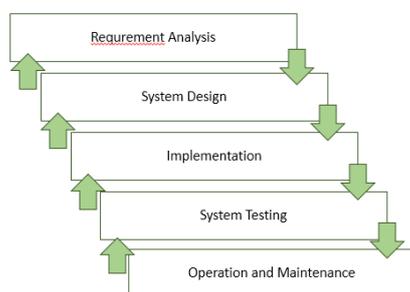
#### A. Metode Pengumpulan Data

Metode pengumpulan data pada penelitian ini dilakukan dengan observasi. Observasi dilakukan di lingkungan produksi penghasil *scrap* produksi. Observasi ini untuk memahami operasional pembuangan *scrap* yang terjadi dan yang akan di *upgrade* dengan cara IoT dan data ditampilkan dalam bentuk digitalisasi.

Pada *scrap* binnya dipasang Raspberry Pi 4, *load cell*, dan sensor *lock door* (untuk indikasi membuka bin), dan *traffic lamp* (untuk indikasi keadaan *scrap* sudah penuh atau belum). Data yang direkam adalah berat *scrap bin*, yang mana secara visual berat bin secara *real-time* dapat dilihat pada aplikasi web yang dibuat ini, tepatnya di halaman layoutnya. Selain itu, jika bin sudah penuh, maka juga bisa memberikan *alert* yang tampil di halaman layout aplikasi web monitoringnya.

#### B. Metode Pengembangan Sistem

Metode pengembangan aplikasi yang akan dipakai untuk pembuatan Aplikasi web tersebut ialah model Waterfall. Dapat diketahui model *Waterfall* alur model bergantung pada pekerjaan sebelumnya jadi jalur kerja pada sistem model ini berupa linear [12], [13]. Selengkapnya dapat dilihat pada Gambar 1 di bawah ini.



Gambar 1. Metode Waterfall

Requirement Analysis and Definition adalah tahapan Analisa dalam pembangunan aplikasi monitoring yang dapat dilihat pada Tabel 1, Tabel 2 dan Tabel 3, melakukan kick off meeting (memberikan konsep rancangan

berdasarkan observasi ke atasan), design concept memberikan konsep atau proposal final ke atasan, serta concept approval (final konsep dari revisi meeting sebelumnya).

Pada Tahap System and Software Design ini akan dibuat suatu arsitektur sistem dari persyaratan yg sudah ditetapkan. Seperti melakukan design flow pembuangan *scrap* dari step 1 ke step 4, desain cabinet bin yang ada di *station*, desain architecture system, rancangan Use case, *Class diagram*, flowchart sistem keseluruhan, desain User Interface aplikasi dengan figma, skema relasi aplikasi, dan mendata keperluan software dan hardware yang dibutuhkan.

Selama tahap implementasi, hasil desain perangkat lunak diwujudkan sebagai rangkaian program atau unit program. Dalam hal ini aplikasi dibuat menggunakan Bahasa Pemrograman PHP, Framework Codeigniter 4 dan menggunakan MySQL sebagai basis datanya.

Dalam tahap *Integration and System Testing* ini, setiap unit program akan diintegrasikan satu sama lain dan hasilnya diuji dengan *blackbox testing*.

Dalam tahap ini dilakukan tahapan instalasi aplikasi monitoring, pembuangan dan pengumpulan *scrap* produksi. Selanjutnya pihak perusahaan mencoba aplikasi tersebut. Bila nantinya saat proses pengujian terjadi error atau bug maka akan segera diperbaiki.

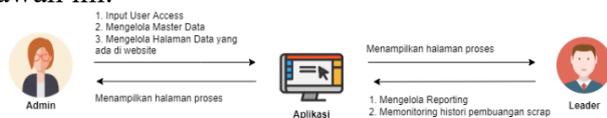
### IV. HASIL DAN PEMBAHASAN

#### A. Deskripsi Umum Sistem

Aplikasi Monitoring Pembuangan dan Pengumpulan *Scrap* adalah aplikasi berbasis web yang bertujuan untuk memonitoring data atau histori pembuangan dan pengumpulan *scrap*, dimana pembuangan *scrap* yang dilakukan oleh operator dan pengumpulan yang dilakukan oleh *storeman* sebelum adanya aplikasi ini, tidak ada pengontrolan secara digitalisasi sehingga data histori tidak dapat di arsip atau disimpan untuk dilakukan review guna menjadi laporan untuk manajemen.

Pengguna Aplikasi Monitoring ini adalah *Company Admin* dan *leader*. *Company Admin* bertugas menjadi *Company Administrator* dalam hal pengelolaan data *scrap* yang berkaitan dengan jenis *scrap*, lokasi *station*, dan penamaan data *scrap* (nomor mesin, nomor jenis *scrap* dan nomor posisi *station*) serta mendaftarkan pengguna guna dapat mengakses kabinet *scrap* yang dimana kabinet hanya dapat dibuka sesuai dengan identitas yang telah didaftarkan oleh *Company Admin* untuk dilakukannya pembuangan dan pengumpulan *scrap*.

*Leader* merupakan pengguna kedua dari Aplikasi monitoring ini dimana *leader* dapat melakukan login untuk masuk ke halaman web, melihat alert di halaman layout, dimana layout tersebut berisikan informasi berat, posisi *scrap* secara realtime dan *leader* dapat mengelola atau mengunduh laporan histori pembuangan dan pengumpulan *scrap* yang terekam oleh web secara realtime, dapat dilihat pada Gambar 2 di bawah ini.



Gambar 2. Gambaran umum sistem

### B. Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan proses untuk memecahkan suatu masalah melalui sistem yang dibuat [14]. Kebutuhan fungsional untuk pembangunan aplikasi ini dapat dilihat pada Tabel 1.

Tabel 1. Kebutuhan Fungsional

Kode	Kebutuhan Fungsional
F001	Pengguna dapat login menggunakan akun yang telah didaftarkan.
F002	<i>Company Admin</i> dapat mengelola data <i>Scrap</i> pada setiap <i>station</i> seperti jenis, urutan, berat bin, dan berat maksimal yang dapat ditampung.
F003	<i>Company Admin</i> dapat mengelola penempatan <i>layout station</i> kabinet yang ada di departemen kapasitor.
F004	<i>Company Admin</i> dapat mengelola menu <i>Company Administrator</i> .
F005	<i>Company Admin</i> dapat mengelola data pengguna seperti nama, nomor barcode id karyawan dan departemen bagian.
F006	<i>Company Admin</i> dapat mengelola hak akses pengguna untuk setiap section.
F007	<i>Company Admin</i> dan <i>Leader</i> dapat mengelola

	data reporting.
F008	<i>Company Admin</i> dapat memasukkan jadwal perencanaan penjualan <i>scrap</i> yang ditampilkan di halaman Dashboard.
F009	<i>Company Admin</i> dapat mendaftarkan QR Code mesin untuk proses authentication membuka kabinet.
F010	<i>Leader</i> dapat melihat <i>alert</i> yang ada di halaman <i>layout</i> .
F011	Setiap pengguna dapat melakukan logout untuk keluar dari aplikasi.

### C. Kebutuhan Non-Fungsional

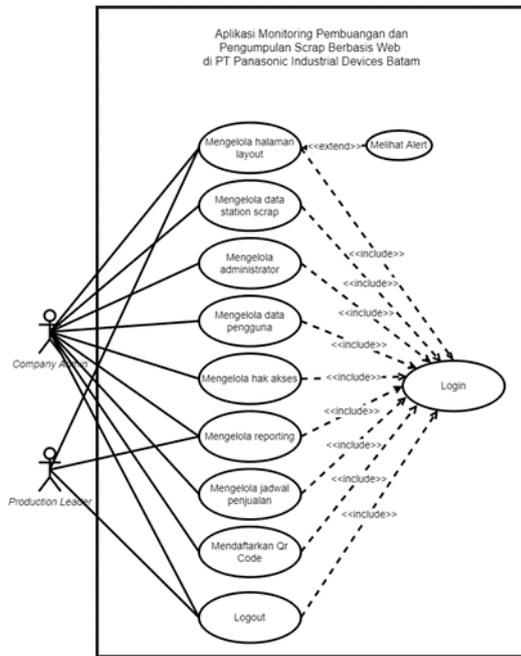
Kebutuhan Non Fungsional adalah kebutuhan tambahan yang berisi batasan yang disediakan system [15], [16]. Kebutuhan non fungsional pada aplikasi ini dapat dilihat pada Tabel 2.

Tabel 2. Kebutuhan Non Fungsional

Kode	Kebutuhan Non Fungsional
NF001	Sistem tersedia dalam Bahasa Indonesia.
NF002	Sistem hanya dapat digunakan oleh pengguna yang telah didaftarkan atau memiliki hak akses.
NF003	Sistem hanya dapat digunakan menggunakan jaringan pribadi Perusahaan Industrial Devices Batam.
NF004	Pengajuan pengguna baru diajukan melalui akun <i>teams Leader</i> .

### D. Diagram Use Case

Diagram Use Case adalah bentuk pertama dari persyaratan pembuatan sistem untuk program perangkat lunak yang baru dan belum dikembangkan [17].

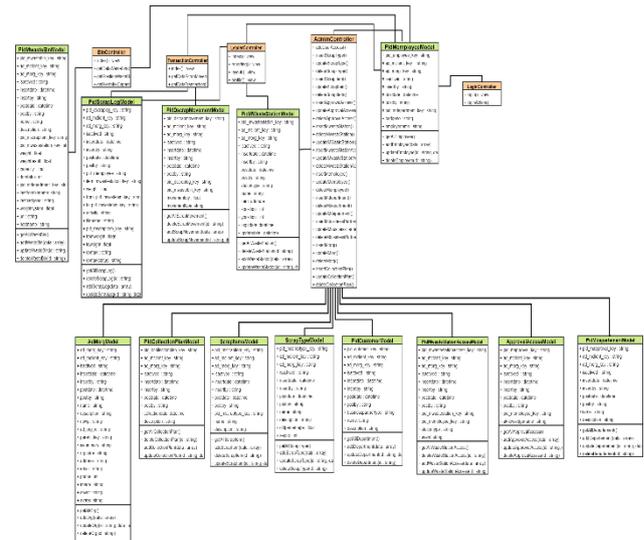


Gambar 3. Diagram Use Case

Dapat dilihat dari Gambar 3 terdiri dari 3 komponen yaitu sistem, aktor, dan use case [18]. Untuk sistem digambarkan bentuk persegi berisi use case. Pada komponen aktor digambarkan berbentuk orang pada luar sistem menunjukkan interaksi aktor dengan use case pada sistem. Gambar terakhir use case digambarkan dengan lingkaran lonjong menunjukkan fungsional pada sistem.

### E. Diagram Class

Class diagram adalah diagram yang menggambarkan secara jelas struktur sistem beserta hubungan setiap objeknya. Dikarenakan sistem yang akan dibangun menggunakan konsep MVC [19], [20] dengan tujuan dapat dikerjakan oleh beberapa developer. Merancang struktur sistem dalam hal mendefinisikan kelas yang dibuat untuk membuat sistem atau class diagram ditunjukkan pada Gambar 4.



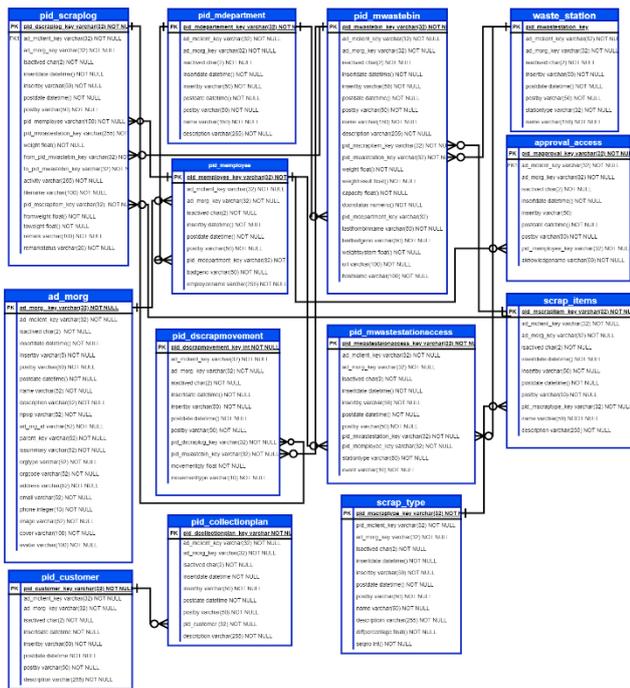
Gambar 4. Class diagram

Aplikasi yang akan dirancang menerapkan arsitektur MVC (*Model View Controller*) maka class diagram pada pembangunan aplikasi ini digambarkan sebagai model dan controller. Di mana model digambarkan dengan warna kuning sedangkan controller diberi warna hijau. Berdasarkan Gambar diatas diketahui bahwa terdapat 5 controller dan 13 model.

### F. Entity Relationship Diagram

ER Diagram adalah diagram yang digunakan untuk perancangan suatu basis data dan menunjukkan relasi antar entitas beserta atribut-atributnya [21]. ER Diagram yang dihasilkan untuk pembangunan aplikasi ini memiliki 13 Entitas, yaitu diantaranya *pid\_scraplog*, *ad\_morg*, *pid\_customer*, *pid\_mdepartment*, *pid\_employeed*, *pid\_dscrapmovement*, *pid\_collectionplan*, *pid\_mwastebin*, *pid\_mwastationaccess*, *scrap\_type*, *waste\_station*, *approval\_access*, dan *scrap\_items*.

Rancangan Struktur dalam pemodelan basis data, akan menggunakan relasi Tabel yang akan ditunjuk pada Gambar 5.



Gambar 5. Entity Relationship Diagram

G. Antarmuka Sistem

Ada banyak antarmuka yang dihasilkan dari pengembangan aplikasi ini. Hanya saja yang disajikan di sini hanya beberapa di antaranya saja, yaitu halaman *dashboard*, *layout*, *station maker*, *waste station*, dan *define waste bin*.

Halaman *dashboard* adalah halaman dimana pengguna berhasil *login* ke dalam aplikasi web. Pada halaman dashboard informasi mengenai *Scrap* perharinya seperti *Receivable*, *Saleable*, Grafik perbandingan dan jumlah *scrap* perharinya secara detail perjenis *scrapnya*. Tampilan dapat dilihat pada Gambar 6 berikut.



Gambar 6. Halaman Dashboard

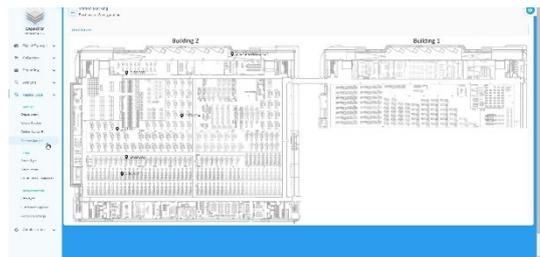
Halaman *Layout* merupakan halaman dimana pengguna seperti *Company Admin*,

*Storeman* dan *Leader* bisa melihat kondisi bin di *station* secara *real-time* dimulai dari *current weight* dan *alert* ketika bin sudah penuh. Tampilan dapat dilihat pada Gambar 7 berikut.



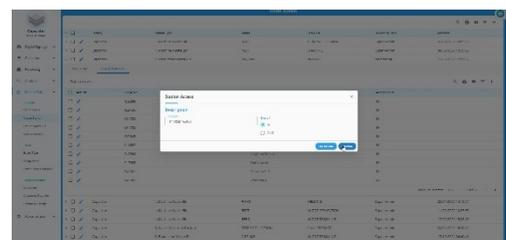
Gambar 7. Halaman Layout

Halaman *Station Marker* merupakan halaman yang berfungsi untuk merubah atau memperbaharui letak *station* terbaru ketika terjadi perubahan *layout* di area produksi dan halaman ini hanya bisa diakses oleh *Company Admin*. Tampilan dapat dilihat pada Gambar 8.



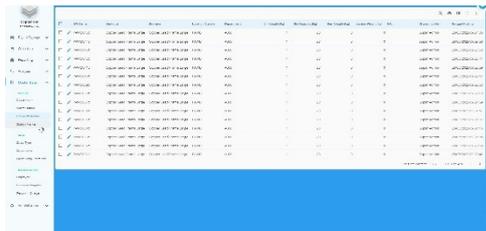
Gambar 8. Halaman Station Marker

Halaman *Station* merupakan halaman yang hanya bisa dikelola oleh *Company Admin*, di halaman ini *Company Admin* bisa menambah atau merubah data *station* yang ada di area produksi berdasarkan inisial proses, contohnya: BFBD -> Anode Trimming. Tampilan dapat dilihat di Gambar 9 di bawah ini.



Gambar 9. Halaman Waste Station

Halaman *Define Waste Bin* merupakan halaman yang berisikan data QR Code yang berada di *station* yang menerangkan QR Code tersebut bisa meliputi data: nomor mesin, jenis tipe *scrap*, berat bin dan nama departemen dan halaman ini hanya bisa dikelola oleh *Company Admin*. Ini dapat dilihat pada Gambar 10 di bawah ini.



Gambar 10. Halaman *Define Waste Bin*

#### H. Pengujian Sistem

Pengujian aplikasi dilakukan dengan cara menguji aplikasi yang telah dikembangkan. Pengujian melibatkan *end user* untuk melakukan pengujian yang akan disebut sebagai penguji. Penguji akan melakukan pengujian, lalu mengisi kuesioner mengenai kegunaan yang dimiliki pada aplikasi. Pada pengujian aplikasi ini digunakan *black-box testing*.

Tujuan dari *blackbox testing* adalah untuk melakukan pengujian terhadap kebutuhan fungsional aplikasi monitoring pembuangan dan pengumpulan *scrap*[22][22]. Pengujian ini dilakukan terhadap 10 testcase pada halaman *Company Admin*. Setelah melakukan pengujian, didapatkan hasil bahwa 100% seluruh *testcase* milik *Company Admin* berhasil berfungsi sesuai dengan yang diharapkan. Hasil pengujian dapat dilihat pada Tabel 3.

Tabel 3. Pengujian

ID pengujian	Aktifitas Pengujian	Hasil	Bukti
TC 01	Form Login	Proses autentikasi pengguna berhasil dan diarahkan ke	

		halaman Dashboard	
TC 02	Halaman Layout	Pengguna dapat melihat berat, posisi dan kondisi <i>station</i> secara realtime	
TC 03	Halaman Reporting	Pengguna dapat melihat dan mengunduh laporan reporting sesuai yang diinginkan pengguna	
TC 04	Halaman Collection Option	<i>Company Admin</i> dapat memperbarui data collection option dan data dapat tampil dihalaman dashboard	
TC 05	Halaman Employee	<i>Company Admin</i> dapat menambahkan pengguna baru	
TC 06	Halaman Waste Station	<i>Company Admin</i> dapat mengisi hak akses sesuai dengan proses produksi employee	
TC 007	Form Define Waste Bin	<i>Company Admin</i> dapat mengisi data barcode mesin	
TC 008	Halaman Station Marker	<i>Company Admin</i> dapat memilih filter nomor mesin dan data layout dapat terbaharui	
TC 009	Halaman Company Administrator	<i>Company Admin</i> dapat menginput data terbaru	

TC 010	Logout	Pengguna dapat keluar dari halaman website	
-----------	--------	--	---

## V. KESIMPULAN

Berdasarkan kesimpulan dari penelitian yang dilakukan adalah Aplikasi monitoring pembuangan dan pengumpulan scrap produksi dengan menggunakan metode waterfall dapat diimplementasikan pada perusahaan XYZ. Pengujian yang dilakukan pada Aplikasi monitoring pembuangan dan pengumpulan scrap dengan menggunakan blackbox testing digunakan untuk mengetahui fungsionalitas system, yang mana hasilnya semua fungsionalitas telah berjalan dengan semestinya.

## REFERENSI

- [1] V. E. Satya, "Pancasila Dalam Menghadapi Era Revolusi Industri 4.0," *Pus. Penelit. Badan Keahlian DPR RI, X*, vol. 9, p. 19, 2018.
- [2] A. Sasmitadharjo, "DIPLOMASI EKONOMI INDONESIA TERHADAP SINGAPURA DALAM MENGHADAPI PERSAINGAN DIGITAL DUNIA TAHUN 2018-2020," Universitas Muhammadiyah Malang, 2021.
- [3] L. Banjanović-Mehmedović and F. Mehmedović, "Intelligent manufacturing systems driven by artificial intelligence in industry 4.0," in *Handbook of Research on Integrating Industry 4.0 in Business and Manufacturing*, IGI global, 2020, pp. 31–52.
- [4] Q. Qi and F. Tao, "Digital twin and big data towards smart manufacturing and industry 4.0: 360 degree comparison," *Ieee Access*, vol. 6, pp. 3585–3593, 2018.
- [5] H. Nguyen, K. Tran, X. Zeng, L. Koehl, P. Castagliola, and P. Bruniaux, "Industrial Internet of Things, big data, and artificial intelligence in the smart factory: A survey and perspective," in

*ISSAT International Conference on Data Science in Business, Finance and Industry*, 2019, pp. 72–76.

- [6] R. R. Tjandrawinata, "Industri 4.0: Revolusi industri abad ini dan pengaruhnya pada bidang kesehatan dan bioteknologi," *J. Med.*, vol. 29, no. 1, pp. 31–39, 2016.
- [7] A. Wuryanto-UBSI, N. Hidayatun-UBSI, and S.-S. N. Mandiri, "Rancang Bangun Aplikasi Persediaan Scrap (Studi Kasus PT Metal Castindo Industriatama)," *SPEED-Sentra Penelit. Eng. dan Edukasi*, vol. 12, no. 2, 2020.
- [8] A. Puspasari, D. Mustomi, E. Anggraeni, C. Sitasi, and A. Puspasari, "Proses Pengendalian Kualitas Produk Reject dalam Kualitas Kontrol Pada PT. Yasufuku Indonesia Bekasi," *J. Sekr. Dan Manaj.*, vol. 3, no. 1, pp. 71–78, 2019.
- [9] M. Wegmuller, J. P. von der Weid, P. Oberson, and N. Gislin, "High Resolution Fiber Distributed Measurements With Coherent {OFDR}," *Proc. {ECOC}'00*, no. April, p. 109, 2000.
- [10] E. Susanti and J. Triyono, "Prototype Alat IoT (Internet Of Things) untuk Pengendali dan Pemantau Kendaraan Secara Realtime," 2016.
- [11] D. Iryaning, "Perancangan sistem informasi inventori gudang berbasis intranet," *J. Tek. Ind.*, vol. 9, no. 2, pp. 147–154, 2008.
- [12] G. W. Sasmito, "Penerapan metode Waterfall pada desain sistem informasi geografis industri kabupaten Tegal," *J. Inform. J. Pengemb. IT*, vol. 2, no. 1, pp. 6–12, 2017.
- [13] E. M. Khalifa, D. Jawawi, and H. A. Jamil, "An efficient method to generate test cases from UML-use case diagram," *Int. J. Eng. Res. Technol.*, vol. 12, no. 7, pp. 1138–1145, 2019.
- [14] F. Zailani, "Literature Review: Penggunaan Teknik dan Analisis Requirement Engineering Dalam Perancangan Aplikasi Website."
- [15] N. F. Najwa, M. A. Furqon, and F. Mahananto, "Literature Review on Extended Use case in Modeling Non-

- functional Requirement,” *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, vol. 3, no. 3, pp. 1832–1844, 2018.
- [16] M. Binkhonain and L. Zhao, “A review of machine learning algorithms for identification and classification of non-functional requirements,” *Expert Syst. with Appl. X*, vol. 1, p. 100001, 2019.
- [17] H. Koç, A. M. Erdoğlan, Y. Barjakly, and S. Peker, “UML diagrams in software engineering research: a systematic literature review,” *Multidiscip. Digit. Publ. Inst. Proc.*, vol. 74, no. 1, p. 13, 2021.
- [18] T. Ahmad, J. Iqbal, A. Ashraf, D. Truscan, and I. Porres, “Model-based testing using UML activity diagrams: A systematic mapping study,” *Comput. Sci. Rev.*, vol. 33, pp. 98–112, 2019.
- [19] M. Massago and T. E. Colanzi, “An assessment of tools for UML class diagram modeling: support to adaptation and integration with other tools,” in *Proceedings of the XVIII Brazilian Symposium on Software Quality*, 2019, pp. 10–19.
- [20] M. Elaasar and Y. Labiche, “Diagram definition: a case study with the UML class diagram,” in *International Conference on Model Driven Engineering Languages and Systems*, 2011, pp. 364–378.
- [21] F. Huber and G. Hagel, “Tool-supported teaching of UML diagrams in software engineering education-A systematic literature review,” in *2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO)*, 2022, pp. 1404–1409.
- [22] Tri Snadhika Jaya, “Testing IT An Off The Shelf Software Testing Process,” *J. Inform. Pengemb. IT*, vol. 3, no. 2, pp. 45–46, 2018.