

Aplikasi *Monitoring* Produktivitas Proses *Injection Moulding* Menggunakan Algoritma *Selection Sort* Berbasis *Website* (Studi Kasus : PT Mada Wikri Tunggal)

Rudi Budi Agung.S.Si.,M.M.S.I

Dosen STMIK Bani Saleh, Email: rudibudiagung73@gmail.com

Tunik Dwi Lestari.S.Kom

Jurusan Teknik Informatika, Sekolah Tinggi Manajemen Informatika & Komputer Bani Saleh, Bekasi
Jln. Mayor M.Hasibuan No.68, Margahayu, Bekasi Timur, Kota Bekasi, 17113, Indonesia
email: dwitunik@gmail.com

Abstract—*Injection moulding process productivity monitoring system is a system to oversee the achievement results of production process real-time. Based on research and analysis, productivity monitoring activity injection moulding process that runs on PT Mada Wikri Tunggal still delayed due to sectoral data, data processing errors and lack of production activity information. This analysis shown production results, Not Good product, downtime, and control of material usage, with website-based injection moulding monitoring application process using codenigniter framework, MySQL, with selection sort algorithm with descending methods to sort 3 of biggest ratings of Not Good product, downtime and material usage. This is intended to provide effectiveness in getting information about the productivity of injection moulding process to be more informative, easy, fast, and accurate.*

Abstrak—Sistem monitoring produktivitas proses *injection moulding* merupakan sistem yang dibangun untuk mengawasi pencapaian hasil proses produksi secara *realtime*. Berdasarkan penelitian dan analisis, aktivitas *monitoring* produktivitas proses *injection moulding* yang berjalan pada PT Mada Wikri Tunggal masih terjadi keterlambatan informasi karena data bersifat sektoral, adanya kesalahan pengolahan data dan kurangnya informasi aktivitas produksi. Analisis ini membahas mengenai pencapaian hasil produksi, produk cacat/*Not Good*, *downtime*, dan pengendalian material yang digunakan, dengan pembuatan aplikasi *monitoring* proses *injectionmoulding* berbasis *website* dengan menggunakan *framework codenigniter*, *MySQL*, dan penggunaan algoritma *selection sort* dengan metode *descending* yang difungsikan untuk mengurutkan 3 peringkat terbesar produk cacat/*Not Good*, *downtime* dan penggunaan material. Hal ini ditujukan untuk memberikan efektifitas dalam mendapatkan informasi mengenai produktivitas proses *injectionmoulding* menjadi lebih informatif, mudah, cepat, dan akurat.

Kata Kunci—*Monitoring Produktivitas InjectionMoulding, SelectionSort, FrameworkCodeigniter, PHP dan MySQL*

I. PENDAHULUAN

PT Mada Wikri Tunggal adalah sebuah perusahaan nasional yang bergerak dalam bidang industri manufaktur. Ruang lingkup PT Mada Wikri Tunggal meliputi produktivitas dari komponen-komponen plastik. Selama 35 Tahun, PT Mada Wikri Tunggal mengelola komponen-komponen tersebut untuk menghasilkan produk-produk teknis dengan presisi yang membutuhkan teknologi tinggi, serta

hasil produk yang dihasilkan banyak digunakan oleh pelanggan khususnya dalam bidang otomotif, seperti interior dan eksterior kendaraan, serta *spare parts* kendaraan.

Proses *injectionmoulding* adalah metode atau proses kerja material berbahan dasar biji plastik yang meleleh karena proses pemanasan yang di injeksikan oleh *plunger*, ke dalam cetakan/*moulding* yang telah didinginkan oleh air. Setelah proses tersebut dilalui, produk yang dihasilkan akan berubah menjadi dingin dan menghasilkan perubahan tekstur menjadi keras sehingga produk yang dihasilkan dapat dikeluarkan dari cetakan/*moulding*. Setelah proses tersebut selesai dilakukan, proses selanjutnya dilakukan dokumentasi hasil proses *injectionmoulding*.

Hasil proses *injectionmoulding* haruslah selalu terdokumentasi dengan baik, hal ini ditujukan untuk mengetahui berapa banyak mesin yang beroperasi sesuai dengan jadwal, berapa banyak *downtime* terjadi dan produk cacat/*NotGood* yang dihasilkan, maupun produk OK yang dihasilkan sehingga menjadi masukan untuk manajemen dalam membuat keputusan tentang hasil proses *injection* yang terjadi. *Monitoring* hasil proses *injectionmoulding* dimanfaatkan oleh manajemen untuk melakukan pemantauan data produk cacat/*NotGood* maupun *downtime* secara *realtime*, mengetahui kondisi aktual dari proses *injectionmoulding* yang berjalan, mengetahui efektifitas, serta jika *monitoringinjectionmoulding* dilakukan secara efektif maka akan meminimalisir *losstime* yang terjadi dan dapat dijadikan tindakan yang *preventive* bagi manajemen dalam meningkatkan mutu dan kualitas produk yang dihasilkan.

Untuk itu diperlukan sebuah aplikasi yang dapat mengintegrasikan mengenai data-data yang mempengaruhi produktivitas khususnya pada proses *injectionmoulding* seperti: pencapaian hasil proses produksi, produk cacat/*NotGood*, *downtime*, dan pengendalian material proses yang digunakan untuk *monitoring* data proses dengan kondisi aktual produk yang dihasilkan secara *realtime*. Hal ini bertujuan untuk memperbaiki dan meningkatkan sistem produktivitas proses *injectionmoulding* secara cepat, tepat dan akurat.

II. PENELITIAN YANG TERKAIT

Dalam penelitian ini juga menjelaskan beberapa hasil penelitian dari penulis lainnya mengenai penerapan algoritma *selectionsort* di beberapa contoh kasus yang adabeserta landasan teori sebagai berikut:

A. Tinjauan Pustaka

a) Penelitian yang dilakukan oleh Zaima Faiza Hakim pada *Journal Information Engineering and Educational Technology*, di Surabaya, pada tanggal 20 Januari 2017 terkait “Implementasi Metode *SelectionSort* Untuk Barang Yang Harus Di Stok Ulang Dalam Sistem Informasi Penjualan”, diperoleh kesimpulan bahwa penelitian yang dilakukan ditujukan untuk menghasilkan sebuah laporan data barang, data *supplier*, data *customer*, data penjualan, serta data pembelian yang telah terurut dari stok paling sedikit dan stok paling banyak sehingga dapat memberikan sebuah informasi kepada *user* apabila stok barang tersisa sedikit sehingga *user* dapat mempersiapkan pengajuan barang apa yang akan dipesan kembali untuk memenuhi kebutuhannya. Penerapan *selectionsort* pada penelitian yang dilakukan menggunakan *ascending* atau mengurutkan dari nilai terkecil ke terbesar dari stok barang dengan menggunakan *PPH* dan *databasePostgrSQL*. [1]

b) Penelitian yang dilakukan oleh Benardo, Mesterjon, dan Leni Natalia Zulita terkait “Implementasi Metode *SelectionSort* Untuk Menentukan Nilai Prestasi Siswa Kelas 3 Dan Kelas 4 SD Negeri 107 Seluma”, diperoleh kesimpulan hasil dari metode *selectionsort* yaitu *descending*, dengan mengurutkan nilai terbesar ke nilai terkecil dari hasil penilaian siswa. Didapati *rank* berdasarkan nilai yang didapati siswa tiap mata pelajaran. Pada sistem ini, penilaian berdasarkan kelas dan semester sehingga mata pelajaran yang terdapat dikelas dan semester tersebut akan muncul berserta nilai yang didapatinya. Penerapan yang dilakukan menggunakan bahasa pemrograman *VisualBasicNet 2010* dengan *database SQL Server 2008r2*. Berdasarkan hasil pengujian dan kuisisioner yang dilakukan, dengan adanya penerapan metode *selectionsort* pada sistem penilaian siswa, dapat membantu dalam membuat peringkat/ranking kelas pada SD Negeri 107 Seluma dengan hasil persentase 92% menjawab ya, 4% menjawab tidak, dan 4% menjawab tidak tahu. [2]

c) Penelitian yang dilakukan oleh Panny Agustia Rahayuningsih terkait “Analisis Perbandingan Kompleksitas Algoritma Pengurutan Nilai (*Sorting*)”, Diperoleh kesimpulan bahwa peran algoritma dalam perangkat lunak sangatlah penting, algoritma pengurutan data sendiri sangatlah banyak dan memiliki kompleksitas masing-masing, namun secara sederhana dari perbandingan algoritma yang dilakukan oleh peneliti ditemukan bahwa algoritma *selectionsort* lebih cepat dalam mengurutkan data dibandingkan dengan algoritma *bubblesort*, hal ini ditunjukkan dengan kecilnya nilai yang didapat oleh algoritma *bubblesort*. Algoritma *quicksort* lebih cepat dalam melakukan pengurutan data dibandingkan dengan *shell sort*, *insertion sort*, *selection sort*, dan *bubble sort*. Algoritma *shell sort* yang paling efisien dibandingkan algoritma lainnya dengan kompleksitas yang sama. Algoritma *insertion sort* dua kali lebih cepat dan efisien dibandingkan *bubble sort* namun algoritma ini masih kurang efisien untuk tabel yang berukuran besar (menyimpan banyak nilai). Sedangkan algoritma *bubble sort* merupakan algoritma yang sederhana sehingga lebih mudah dipahami dengan jumlah data sedikit, *selection sort* dapat diunggulkan bila dibandingkan dengan *bubble sort*. [3]

B. Landasan Teori

a) Monitoring

Monitoring dapat di definisikan sebagai langkah apakah kegiatan yang dilaksanakan telah sesuai dengan rencana, mengidentifikasi masalah yang timbul agar langsung dapat diatasi, melakukan penilaian apakah pola kerja dan manajemen yang digunakan sudah tepat untuk mencapai tujuan, mengetahui kaitan antara kegiatan dengan tujuan untuk memperoleh ukuran kemajuan. [4]

b) Produktivitas

Produktivitas adalah suatu pendekatan interdisipliner untuk menentukan tujuan yang efektif, pembuatan rencana, aplikasi penggunaan cara yang produktif untuk menggunakan sumber secara efisien dan tetap menjaga adanya kualitas yang tinggi. [5]

c) Proses *Injection Moulding*

Proses *injection moulding* adalah proses pembentukan material polimer biasanya dilakukan pada suhu tinggi dan dengan aplikasi tekanan. Tekanan diberikan untuk mempertahankan bentuk atau pola cetakan ketika produk mengalami pendinginan. Namun untuk material *thermoset* biasanya dapat dikeluarkan pada saat masih panas, karena bentuknya sudah stabil atau tetap bertahan. [6]

d) Downtime

Waktu kerusakan mesin biasa disebut dengan *downtime*, dimana *downtime* adalah jumlah waktu dimana suatu *equipment* tidak dapat berfungsi disebabkan adanya kerusakan (*failure*). [7]

e) Produk Cacat (*Not Good*)

Produk cacat adalah produk yang dihasilkan dalam proses produksi, dimana produk yang dihasilkan tersebut tidak sesuai dengan standar mutu yang ditetapkan, tetapi masih bisa diperbaiki dengan mengeluarkan biaya tertentu. [8]

f) Algoritma *Selection Sort*

Algoritma *selection sort* adalah suatu metode pengurutan yang membandingkan elemen yang sekarang dengan elemen berikutnya sampai ke elemen yang terakhir. Jika ditemukan elemen lain yang lebih kecil dari elemen sekarang maka dicatat posisinya dan langsung ditukar. Konsep proses *selection sort* adalah mencari (memilih) nilai terkecil dan menukarnya dengan elemen paling awal (paling kiri) pada setiap tahap. Proses *sort* dilakukan tahap per tahap. [9]

Konsep yang digunakan yaitu *selection sort* secara *descending*. Konsep ini dilakukan dengan cara pemilihan dari suatu nilai yang terbesar dan kemudian menukarnya dengan elemen paling awal, lalu membandingkan dengan elemen yang sekarang dengan elemen berikutnya sampai dengan elemen terakhir, perbandingan dilakukan terus sampai tidak ada lagi pertukaran data. [3]

g) Website

Website merupakan kumpulan dari halaman-halaman yang berhubungan dengan file-file lain yang saling terkait. Dalam sebuah *website* terdapat satu halaman yang dikenal dengan sebutan *homepage*. *Homepage* adalah sebuah halaman yang pertama kali dilihat ketika seseorang mengunjungi sebuah *website*. [10]

h) *Hypertext Preprocessor (PHP)*

PHP adalah *script* yang digunakan untuk membangun halaman web yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu

diminta oleh *client*, mekanisme ini menyebabkan informasi yang diterima *client* selalu yang terbaru. Semua script *PHP* dieksekusi pada server di mana *script* tersebut dijalankan.[11]

Framework Codeigniter adalah sebuah *framework* dari *php* yang menggunakan konsep *MVC (Model, View, dan Control)*, yang bertujuan supaya pada pengembangan perangkat lunak yang besar mudah untuk dilakukan *maintenance*. Konsep *MVC* sendiri merupakan konsep yang memisahkan komponen utama menjadi tiga komponen lainnya yang dapat dilihat penjelasannya sebagai berikut:

1. *Model* berfungsi untuk penanganan yang berhubungan langsung dengan *database*. Seperti memasukan data, pembaharuan data, penghapusan data atau sinkronisasi data dengan *database* yang digunakan.
2. *View* berfungsi untuk menangani dan mengatur tampilan *website* yang disesuaikan dengan kebutuhan, sehingga pengguna tertarik dan nyaman menggunakan perangkat lunak tersebut.
3. *Controller* berfungsi untuk menjembatani antara *model* dan *view*. *Controller* berisi instruksi atau perintah yang berfungsi menerima *request* dari data pengguna, setelah itu diproses dengan cara menentukan langkah selanjutnya yang akan di proses oleh perangkat lunak.

i) *phpMyAdmin*

phpMyAdmin merupakan sebuah aplikasi berbasis web yang dibuat dari pemrograman *PHP* dan diramu dengan *JavaScript*. *phpMyAdmin* disebut sebagai *tools* yang berguna untuk mengakses *databaseMySQLServer* dalam bentuk tampilan web.[12]

III. ANALISIS SISTEM YANG BERJALAN

A. Tinjauan Organisasi

PT Mada Wikri Tunggal didirikan pada tahun 1984 oleh Ir.Achmad Sarbini, salah satu proses produksinya yaitu proses *injectionmoulding* yang menghasilkan komponen-komponen plastik. PT Mada Wikri Tunggal dengan seluruh jajaran manajemen dan staf berusaha untuk terus maju dan berkembang di bidang manufaktur.Nilai-nilai perusahaan yang dikembangkan berfokus pada beberapa poin sebagai yaitu : fokus pada pendekatan *QCD*, integritas,transparan,kerjasama tim dan inovasi.

B. Analisis Sistem Yang Berjalan

Dari analisis pengumpulan data yang dilakukan oleh penulis, maka diperoleh informasi bahwa proses dokumentasi di bagian produksi meliputi: produktivitas karyawan dan produktivitas berdasarkan pembagian jam kerja (*shift*), *downtime* mesin, *downtimeprocess*, produk cacat/*NotGood* dan pengendalian material sudah lebih baik dari sebelumnya.

Berdasarkan data yang diperoleh dan dikumpulkan melalui proses pengamatan terhadap permasalahan yang terjadi, maka dapat disimpulkan bahwa masih terdapat kekurangan dalam dokumentasi laporan harian di departemen produksi.

Dalam mengelola hasil produksi tersebut, operator diharuskan menuliskan laporan harian produksi sesuai dengan jumlah produk yang didapatkan, baik dalam kondisi produk OK maupun produk cacat/*NotGood* yang telah melalui proses inspeksi sebelumnya, serta penulisan data *downtime* sesuai dengan PIC mesin yang telah ditetapkan. Setelah itu, admin

produksi melakukan proses input data laporan harian produksi ke *Microsoft excel* sesuai dengan format yang tersedia, sebagai bagian dari proses dokumentasi maupun hasil dari perhitungan produktivitas karyawan, jumlah produk cacat/*NotGood*, termasuk data *downtime* mesin atau *downtimeprocess* yang terjadi.

Berdasarkan analisis data serta melalui proses pengamatan sistem yang berjalan di departemen produksi, maka dapat disimpulkan sebagai berikut:

1. Pengolahan hasil produksi proses *injectionmoulding* menggunakan *Microsoft excel*
2. Penyimpanan hasil proses produksi *injectionmoulding* atau laporan harian produksi disimpan dalam bentuk *softcopy* maupun *hardcopy*
3. Masih terjadi keterlambatan data yang diterima oleh departemen terkait, untuk memperoleh data yang diperlukan. Hal ini dikarenakan data belum terintegrasi antara satu dengan yang lain
4. Perhitungan hasil proses produksi yang belum akurat, dikarenakan rumus pada *Microsoft excel* masih terjadi *error* atau belum saling terhubung
5. Penentuan rating tertinggi untuk kategori produk cacat/*NotGood*, *downtime*, dan penggunaan material membutuhkan waktu yang lama.

Dalam hal ini, penulis mengambil kesimpulan untuk membuat suatu aplikasi *monitoring* produktivitas proses *injectionmoulding* dengan menggunakan algoritma *selectionsort* berbasis *website* untuk membantu departemen produksi dalam mengelola atau mendokumentasikan hasil proses produksi dengan mudah, cepat dan akurat

C. Sistem Yang Diusulkan

Berdasarkan hasil analisis sistem yang berjalan diatas, maka sistem yang diusulkan sebagai berikut:

Tabel 1
Sistem Input

| Aktivitas saat ini | Sistem yang diusulkan |
|--|--|
| File belum terdokumentasi dengan baik. | File tersimpan pada <i>database</i> yang dapat di akses oleh <i>user/pengguna</i> yang terdaftar pada aplikasi <i>monitoring</i> secara <i>realtime</i> . |
| Data hasil produksi hanya bisa di akses oleh departemen produksi. Contohnya : data <i>downtime process</i> yang terjadi pada proses produksi. | Data hasil produksi di <i>input</i> dan disimpan kedalam <i>database</i> , dan secara otomatis akan tampil pada aplikasi <i>monitoring</i> berbasis <i>website</i> . Contoh : data <i>downtime process</i> yang terjadi pada proses produksi hanya dapat diakses bagi yang memiliki hak akses sesuai dengan kebutuhan. |
| Bagianproduksi melakukan <i>input</i> data terkait produk cacat/ <i>Not Good</i> , akan tetapi identifikasi lokasi pada saat penyimpanan data masih terpisah-pisah, sehingga saat data dibutuhkan akan sulit untuk mencarinya. | Bagianproduksi melakukan <i>input</i> data terkait produk cacat/ <i>Not Good</i> pada aplikasi yang telah disediakan, dan secara otomatis tersimpan pada <i>database</i> . Data tersebut dapat diakses secara cepat dan mudah oleh bagian terkait jika sewaktu-waktu dibutuhkan. |

| | |
|--|--|
| <p>Penentuan rating tertinggi untuk kategori produk cacat/Not Good, downtime, dan penggunaan material membutuhkan waktu yang lama.</p> | <p>Penggunaan algoritma <i>selection sort</i> menggunakan metode <i>descending</i>, untuk mempermudah dalam menentukan rating tertinggi untuk kategori cacat/Not Good, downtime, dan penggunaan material membuat proses perhitungan menjadi lebih mudah dan cepat.</p> |
|--|--|

Tabel 2
Sistem Proses

| Aktivitas saat ini | Sistem yang diusulkan |
|---|--|
| <p>Membutuhkan waktu yang lama dan kurang efektif dalam mengakses data, serta sulit dalam pencarian data yang dibutuhkan.</p> | <p>Mempercepat proses pencarian data yang dibutuhkan, dan memudahkan user/pengguna dalam mengakses data untuk menyediakan informasi yang dibutuhkan.</p> |
| <p>Proses kerja yang kurang efektif, menyebabkan user melakukan pekerjaan berulang atau lebih dari satu kali.</p> | <p>Membuat alur atau proses kerja yang ada menjadi lebih efektif, sehingga dapat memberikan solusi untuk memperbaiki aktivitas pekerjaan menjadi lebih baik.</p> |

Tabel 3
Sistem Output

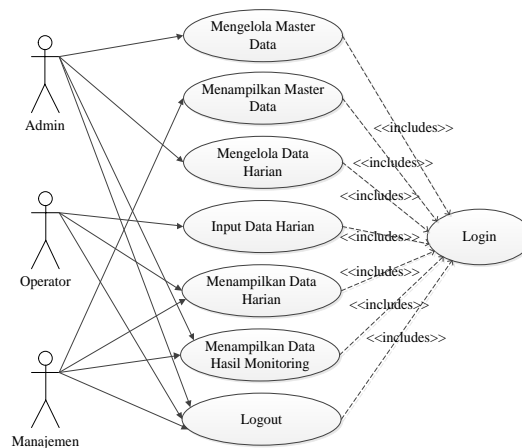
| Aktivitas saat ini | Sistem yang diusulkan |
|--|---|
| <p>Data hasil produksi, data produk cacat/NotGood, downtime, serta produktivitas karyawan hanya berbentuk dokumen <i>hardcopy</i> sehingga dapat rusak dan mudah hilang.</p> | <p>Pihak terkait dapat mengakses data hasil produksi, data produk cacat/NotGood, downtime, serta produktivitas karyawan yang telah tersimpan di dalam <i>database</i> secara mudah & cepat.</p> |

IV. HASIL DAN PEMBAHASAN

Pembahasan terhadap hasil penelitian dan pengujian ini dimulai dari tahap perancangan sistem hingga tahap implementasi sistem sebagai berikut:

A. Perancangan Sistem

Berdasarkan analisis terhadap sistem yang berjalan pada PT Mada Wikri Tunggal penulis melakukan perbaikan melalui perencanaan sistem dari aktivitas yang ada. Penjelasan dari masing-masing diagram yang digunakan penulis dalam melakukan perancangan aplikasi *monitoring* produktivitas proses *injection moulding* menggunakan algoritma *selection sort* berbasis *website* pada PT Mada Wikri Tunggal adalah sebagai berikut:



Gambar 1
Use Case Diagram

Tabel 4
Deskripsi Use Case

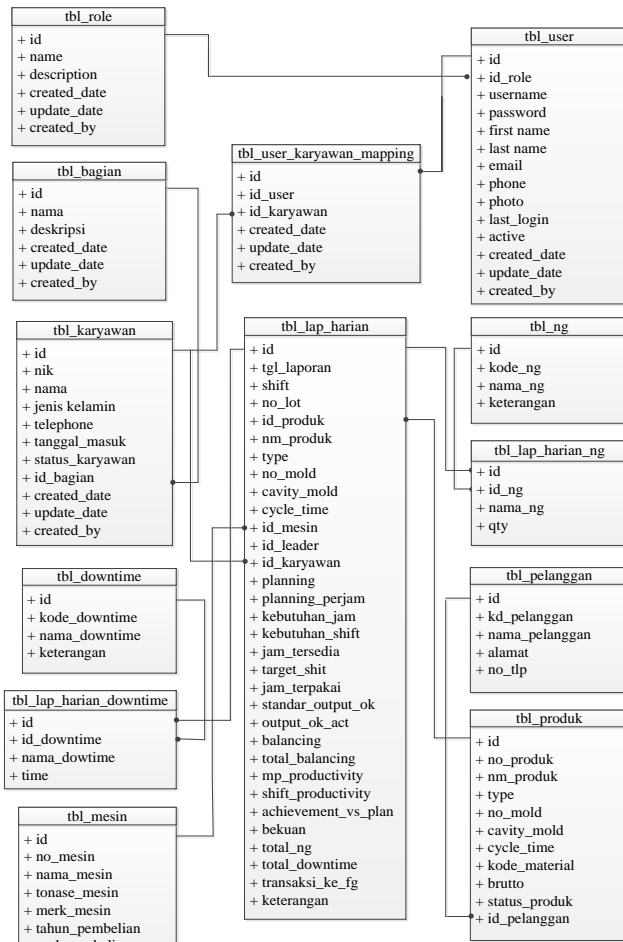
| No | Use Case | Deskripsi |
|-------|--|---|
| UC-01 | <i>Login</i> | Fungsionalitas untuk masuk ke halaman utama aplikasi |
| UC-02 | Mengelola Master Data | Fungsionalitas untuk mengelola master data |
| UC-03 | Menampilkan Master Data | Fungsionalitas untuk menampilkan master data |
| UC-04 | Mengelola Data Harian | Fungsionalitas untuk mengelola data harian |
| UC-05 | Input Data Harian | Fungsionalitas untuk melakukan <i>input</i> data harian |
| UC-06 | Menampilkan Data Harian | Fungsionalitas untuk melihat data harian |
| UC-07 | Menampilkan Data Hasil <i>Monitoring</i> | Fungsionalitas untuk melihat data hasil monitoring |
| UC-08 | <i>Logout</i> | Fungsionalitas untuk keluar dari aplikasi |

Tabel 5
Deskripsi Aktor Pada Sistem

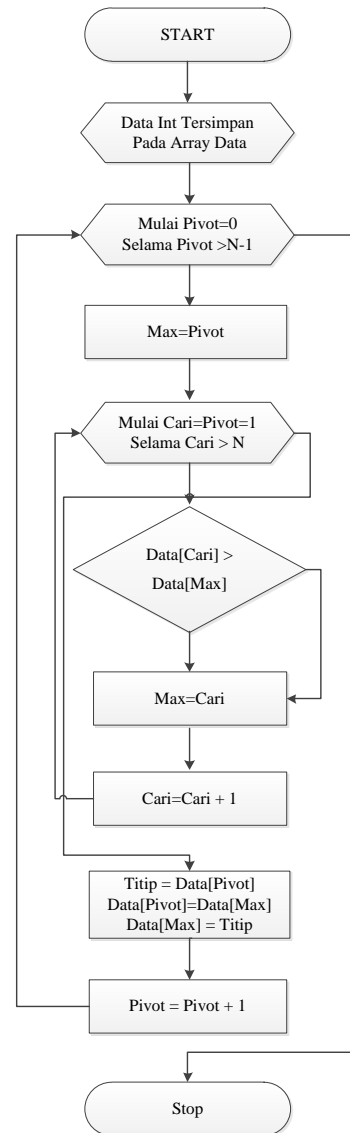
| No | Use Case | Deskripsi |
|------|-----------|--|
| P-01 | Admin | Merupakan aktor yang mengelola aplikasi <i>monitoring</i> produktivitas proses <i>injection moulding</i> |
| P-02 | Operator | Merupakan aktor yang menggunakan aplikasi <i>monitoring</i> produktivitas proses <i>injection moulding</i> |
| P-03 | Manajemen | Merupakan aktor yang menggunakan aplikasi <i>monitoring</i> produktivitas proses <i>injection moulding</i> |

B. Class Diagram

Class diagram ini berguna untuk membantu menjelaskan dalam visualisasikan struktur kelas dari suatu sistem dapat dilihat sebagai berikut:



Gambar 2
Class Diagram



Gambar 3
Algoritma Program

C. Spesifikasi Sistem

Adapun spesifikasi dan sarana pendukung yang digunakan adalah sebagai berikut:

Spesifikasi Perangkat Keras (Hardware)

- 1 Platform Laptop Asus A456U
- 2 Display Monitor 14 inch
- 3 Processor Intel® Core™ i5-7200U, @ 2.50GHz (4 CPUs), -2.7GHz
- 4 Memory 8192MB RAM
- 5 Harddisk 1 TB

Spesifikasi Perangkat Lunak (Software)

- 1 Operating System Windows 10, 64-Bit
- 2 Pembuatan UML Microsoft Visio 2010
- 3 Xampp Xampp Versi 3.2.2
- 4 Browser Mozilla Firefox
- 5 Text Editor Sublime Text

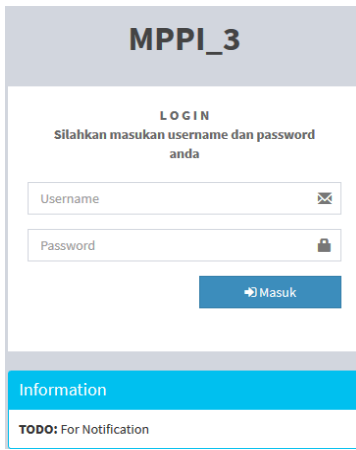
D. Algoritma Program

Algoritma program yang digunakan pada aplikasi monitoring produktivitas proses injectionmoulding menggunakan algoritma selectionsort berbasis website adalah sebagai berikut:

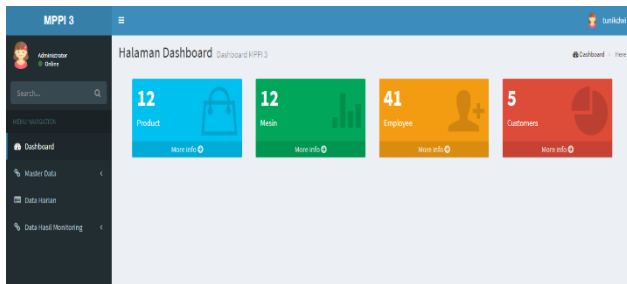
E. Uji Perangkat Lunak



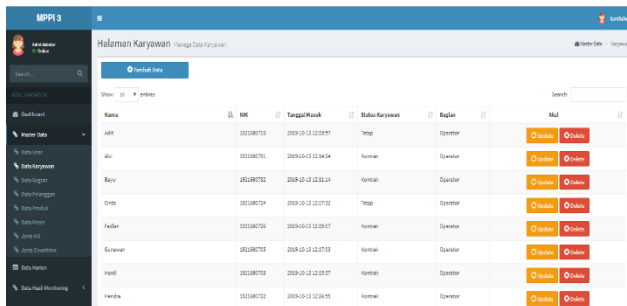
Gambar 4
Halaman Home



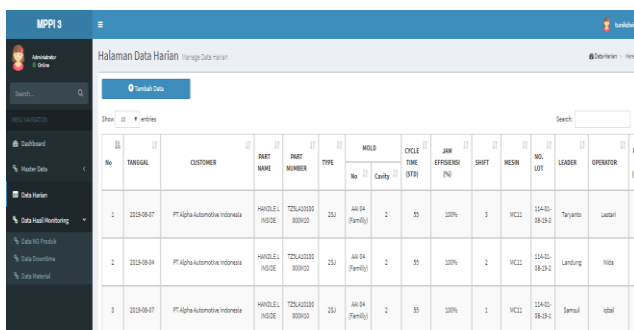
Gambar 5
Halaman Login



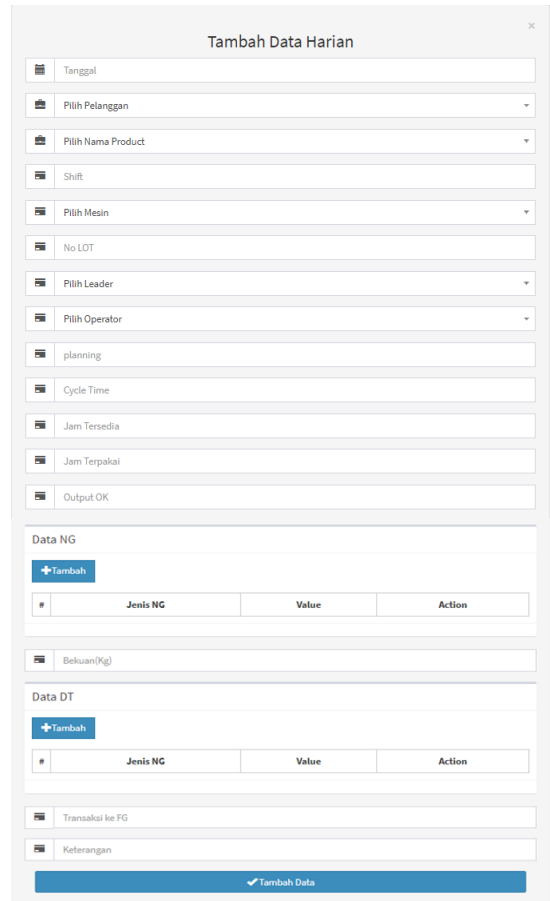
Gambar 6
Halaman Dashboard



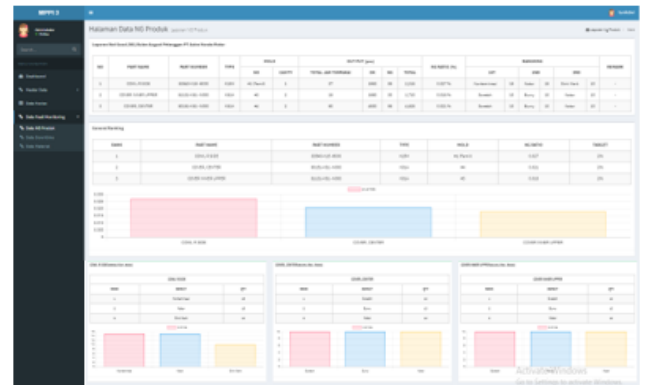
Gambar 7
Halaman Master Data Karyawan



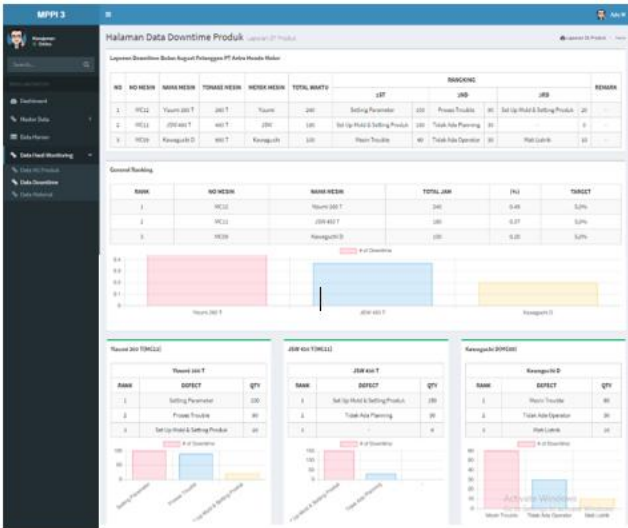
Gambar 8
Halaman Data Harian



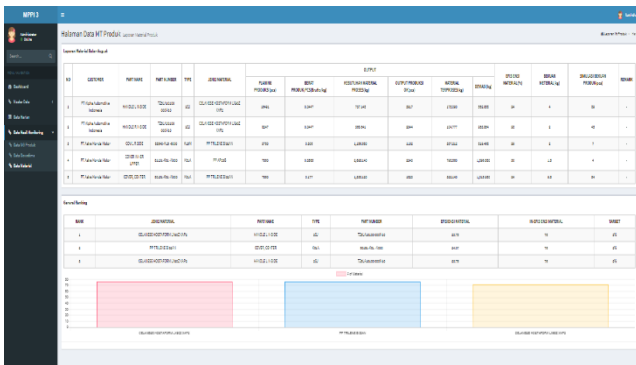
Gambar 9
Halaman Input Data Harian



Gambar 10
Halaman Hasil Monitoring Data NG



Gambar 11
Halaman Hasil Monitoring Data Downtime



Gambar 12
Halaman Hasil Monitoring Data Penggunaan Material

F. Hasil Uji Perangkat Lunak

Pengujian *black box/black box testing* merupakan pengujian perangkat lunak dari segi spesifikasi fungsional tanpa menguji *desain* dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. [13]

Hasil uji *black box testing* adalah sebagai berikut:

Tabel 6
Hasil Uji Black Box Testing

| Item Pengujian | Detail Pengujian | Hasil yang diinginkan | Hasil pengujian |
|-------------------|------------------|---|-----------------|
| Menu Login | Username | User tidak dapat login jika <i>username</i> dan <i>password</i> salah atau tidak sesuai dengan yang ada di <i>database</i> . | OK |
| | Password | User bisa masuk kedalam sistem jika <i>username</i> dan <i>password</i> benar dan sesuai dengan yang ada di <i>database</i> . | OK |
| Dashboard (Admin) | Menu Utama | Menampilkan form-form menu yang di inginkan. | OK |

| | | | |
|-------------------------------|-------------|--|----|
| Master Data User (Admin) | Tambah Data | Menambah data <i>user</i> kedalam <i>database</i> . | OK |
| | Update | Menghapus data <i>uer</i> kedalam <i>database</i> . | OK |
| | Delete | Mengubah data <i>user</i> kedalam <i>database</i> . | OK |
| Master Data Karyawan (Admin) | Tambah Data | Menambah data karyawan kedalam <i>database</i> . | OK |
| | Update | Menghapus data karyawan kedalam <i>database</i> . | OK |
| | Delete | Mengubah data karyawan kedalam <i>database</i> . | OK |
| Master Data Bagian (Admin) | Tambah Data | Menambah data bagian kedalam <i>database</i> . | OK |
| | Update | Menghapus data bagian kedalam <i>database</i> . | OK |
| Master Data Pelanggan (Admin) | Delete | Mengubah data bagian kedalam <i>database</i> . | OK |
| | Tambah Data | Menambah data pelanggan kedalam <i>database</i> . | OK |
| | Update | Menghapus data pelanggan kedalam <i>database</i> . | OK |
| Master Data Produk (Admin) | Delete | Mengubah data pelanggan kedalam <i>database</i> . | OK |
| | Tambah Data | Menambah data produk kedalam <i>database</i> . | OK |
| | Update | Menghapus data produk kedalam <i>database</i> . | OK |
| Master Data Mesin (Admin) | Delete | Mengubah data produk kedalam <i>database</i> . | OK |
| | Tambah Data | Menambah data mesin kedalam <i>database</i> . | OK |
| | Update | Menghapus data mesin kedalam <i>database</i> . | OK |
| Master Jenis NG (Admin) | Delete | Mengubah data mesin kedalam <i>database</i> . | OK |
| | Tambah Data | Menambah data Jenis NG kedalam <i>database</i> . | OK |
| | Update | Menghapus data Jenis NG kedalam <i>database</i> . | OK |
| Master Jenis Downtime (Admin) | Delete | Mengubah data Jenis NG kedalam <i>database</i> . | OK |
| | Tambah Data | Menambah data Jenis <i>Downtime</i> kedalam <i>database</i> . | OK |
| Master Jenis Downtime (Admin) | Update | Menghapus data Jenis <i>Downtime</i> kedalam <i>database</i> . | OK |

| | | | |
|--|-------------------------|---|----|
| | | <i>database.</i> | |
| | Delete | Mengubah data Jenis <i>Downtime</i> kedalam <i>database.</i> | OK |
| Master Data Harian (Admin) | Tambah Data | Menambah data harian kedalam <i>database.</i> | OK |
| | Update | Menghapus data harian kedalam <i>database.</i> | OK |
| | Delete | Mengubah data harian kedalam <i>database.</i> | OK |
| Data Hasil Monitoring (Admin) | Data NG Produk | Menampilkan rating tiga terbesar NG Produk | OK |
| | Data <i>Downtime</i> | Menampilkan rating tiga terbesar <i>Downtime</i> | OK |
| | Data Material | Menampilkan rating tiga terbesar Penggunaan Material | OK |
| Dashboard (Operator) | Menu Utama | Menampilkan form-form menu yang di inginkan. | OK |
| Master Data Harian (Operator) | Tambah Data | Menambah data harian kedalam <i>database.</i> | OK |
| | Tampil Data | Menampilkan data harian sesuai dengan <i>database.</i> | OK |
| | Delete | Mengubah data harian kedalam <i>database.</i> | OK |
| Dashboard (Manajemen) | Menu Utama | Menampilkan form-form menu yang di inginkan. | OK |
| Master Data User (Manajemen) | Tampil Data | Menampilkan data <i>user</i> sesuai dengan <i>database.</i> | OK |
| Master Data Karyawan (Manajemen) | Tampil Data | Menampilkan data karyawan sesuai dengan <i>database.</i> | OK |
| Master Data Bagian (Manajemen) | Tampil Data | Menampilkan data bagian sesuai dengan <i>database.</i> | OK |
| Master Data Pelanggan (Manajemen) | Tampil Data | Menampilkan data pelanggan sesuai dengan <i>database.</i> | OK |
| Master Data Produk (Manajemen) | Tampil Data | Menampilkan data produk sesuai dengan <i>database.</i> | OK |
| Master Data Mesin (Manajemen) | Tampil Data | Menampilkan data mesin sesuai dengan <i>database.</i> | OK |
| Master Data Jenis NG (Manajemen) | Tampil Data | Menampilkan data jenis NG sesuai dengan <i>database.</i> | OK |
| Master Data Jenis <i>Downtime</i> (Manajemen) | Tampil Data | Menampilkan data jenis <i>downtime</i> sesuai dengan <i>database.</i> | OK |

| | | | |
|---|-------------------------|--|----|
| Data Hasil Monitoring (Manajemen) | Data NG Produk | Menampilkan rating tiga terbesar NG Produk | OK |
| | Data <i>Downtime</i> | Menampilkan rating tiga terbesar <i>Downtime</i> | OK |
| | Data Material | Menampilkan rating tiga terbesar Penggunaan Material | OK |
| Menu Logout | Logout | Pengguna berhasil keluar dari sistem aplikasi | OK |

V. KESIMPULAN

A. Kesimpulan

Kesimpulan aplikasi *monitoring* produktivitas proses *injection moulding* menggunakan algoritma *selection sort* berbasis *website* adalah sebagai berikut:

1. Informasi yang dibutuhkan oleh departemen terkait produktivitas proses *injection moulding* menjadi lebih informatif, mudah, cepat dan akurat.
2. Lebih efektif dalam melakukan aktivitas pengolahan data-data terkait produktivitas proses *injection moulding*.
3. Algoritma *selection sort* pada aplikasi ini berfungsi untuk menentukan *rating 3* terbesar menggunakan metode *descending* untuk produk cacat/NotGood, *downtime*, dan *downtime process* yang terjadi selama proses produksi.

UCAPAN TERIMA KASIH

Ucapan terima kasih penulis kepada pihak yang membantu ataupun memberikan dukungan terkait dengan penelitian yang dilakukan seperti bantuan fasilitas penelitian, bimbingan, saran, masukan dan lainnya.

DAFTAR PUSTAKA

- [1] Faiza, Zaima Hakim. 2017. "Implementasi Metode Selection Sort Untuk Menentukan Barang Yang Harus Di Stok Ulang Dalam Sistem Informasi Penjualan". Universitas Negeri Surabaya. Journal Information Engineering and Educational Technology
- [2] Mesterjon, Bernado dan Natalia, Leni Zulita. 2015. "Implementasi Metode Selection Sort Untuk Menentukan Prestasi Siswa Kelas 3 Dan 4 SD Negeri 107 SELUMA". Fakultas Ilmu Komputer Universitas Dehasen Bengkulu. Jurnal Media Infotama. Vol.11 No.1
- [3] Agustia, Panny. 2016. "Analisis Perbandingan Kompleksitas Algoritma Pengurutan Nilai (Sorting)". Jurnal Evolusi Vol. IV No 2
- [4] Sutabri, T. (2012). "Konsep Sistem Informasi. Yogyakarta" : CV. Andi Offset
- [5] Sinungan, Muchdarsah. 2003. "Produktivitas Apa dan Bagaimana". Jakarta: Bumi Aksara
- [6] Oktaviandi, S D.2012. "Analisis Pengaruh Parameter Tekan dan Waktu Penekanan Terhadap Sifat Mekanik dan Cacat Penyusutan dari Produk Injection Molding Berbahan Polyethylene". Skripsi Program Strata-1 Universitas Sultan Agung Tirtayasa
- [7] Kulsum, Febianti, Evi dan Supriatna Irpan H. 2018. "Usulan Waktu Preventive Maintenance Untuk Menurunkan Downtime Mesin Paper Mill 1 Dengan

- Reliability Block Diagram”. Universitas Sultan Ageng. Seminar Nasional Sains dan Teknologi
- [8] Bustami, Bastian dan Nurlela. 2007. “Akuntansi Biaya” Yogyakarta: Graha ilmu
- [9] Yahya, Sofyansyah, Yusari. 2014. “Analisa Perbandingan Algoritma Bubble Sort dan Selection Sort Dengan Perbandingan Eksponensial”. Jurnal Pelita Informatika Budi Darma, Vol. VI No3
- [10] Jhonsen, 2004. “WEB DESIGNER untuk PEMULA”. Jakarta : Elex Media Komputindo Kelompok Gramedia. Anggota IKAPI
- [11] Anhar. 2010. “PHP & MySql Secara Otodidak”. Jakarta: PT TransMedia
- [12] Nugroho, Bunafit. 2009. “Latihan Membuat Aplikasi Web PHP dan MySQL dengan Dreameaver”. Yogyakarta: Penerbit Gava Media
- [13] A. S., Rosa dan Shalahuddin, M. 2013. “Rekayasa Perangkat Lunak Terstruktur Dan Berorientasi Objek”. Bandung: Informatika