

**SISTEM *DATABASE* UNTUK MENCARI
SAMPLE VERIFIKASI *PART NUMBER*
PADA PERUSAHAAN PRODUKSI
KOTA BATAM**

SKRIPSI



**Oleh:
Irwan Susanto Hia
150210212**

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN KOMPUTER
UNIVERSITAS PUTERA BATAM
TAHUN 2019**

**SISTEM *DATABASE* UNTUK MENCARI
SAMPLE VERIFIKASI PART NUMBER
PADA PERUSAHAAN PRODUKSI
KOTA BATAM**

SKRIPSI

**Untuk memenuhi salah satu syarat
guna memperoleh gelar Sarjana**



**Oleh:
Irwan Susanto Hia
150210212**

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN KOMPUTER
UNIVERSITAS PUTERA BATAM
TAHUN 2019**

PERNYATAAN

Dengan ini saya menyatakan bahwa :

1. Skripsi ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik (sarjana, dan/atau magister), baik di Universitas Putera Batam maupun di perguruan tinggi lain.
2. Skripsi ini adalah murni gagasan, rumusan dan penelitian saya sendiri, tanpa bantuan pihak lain, kecuali arah pembimbing.
3. Dalam skripsi ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan dicantumkan dalam daftar pustaka.
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di perguruan tinggi.

Batam, 6 September 2019

Yang membuat pernyataan,

Materai 6000

Irwan Susanto Hia
150210212

**SISTEM *DATABASE* UNTUK Mencari
SAMPLE VERIFIKASI *PART NUMBER*
PADA PERUSAHAAN PRODUKSI
KOTA BATAM**

Oleh:
Irwan Susanto Hia
150210212

SKRIPSI
Untuk memenuhi salah satu syarat
guna memperoleh gelar Sarjana

Telah disetujui oleh Pembimbing pada tanggal
seperti tertera dibawah ini

Batam, 6 September 2019

Rahmat Fauzi, S.Kom., M.Kom.
Pembimbing

ABSTRAK

Teknologi saat ini sudah mencapai fase dimana sangat dibutuhkan. Banyak kegiatan industri sudah sangat membutuhkan teknologi dalam operasionalnya. Saat ini permintaan pasar akan teknologi sangat tinggi. Ini menghasilkan permintaan produksi yang sangat tinggi di perusahaan. Tetapi ada masalah, yaitu jumlah data barang di perusahaan. Tanpa sistem yang dapat merekam semua item di perusahaan, masalah akan muncul. Solusi paling tepat untuk masalah seperti ini adalah, dengan membuat sistem *database*. Penelitian ini bertujuan untuk membuat sistem *database* untuk mencari hasil verifikasi untuk nomor bagian dari mesin *auto vision* dan mesin *hypot*. Sistem basis data ini akan diterapkan pada PT Jovan Technologies. Dalam membuat sistem *database* ini, peneliti menggunakan bahasa pemrograman PHP 7 dan menggunakan MYSQL sebagai *database* sistem ini. Hasil pencarian *part number* sampel verifikasi adalah *barcode* yang berisi data dari *part number*. Sehingga dengan adanya sistem *database* ini, pengumpulan data barang yang ada di perusahaan Jovan Technologies menjadi lebih sistematis dan terstruktur. Penelitian ini sudah ditanggapi oleh beberapa responden. Diantara dari responden tersebut menyatakan bahwa pekerjaan mereka menjadi lebih terbantu dengan adanya sistem *database* ini. Sehingga dengan adanya sistem *database* ini, produksi ganda yang akan mengakibatkan hilangnya sumber daya dan waktu di perusahaan dapat dikurangi dan dihindari agar terciptanya efisiensi sumber daya dari perusahaan.

Kata Kunci : *Mesin Auto Vision, Mesin Hypot, Sistem Database, Barcode*

ABSTRACT

At present the market demand for technology is very high. It results in very high production demand at the company. But there are problems, namely the amount of data on goods in the company. Without a system that can record all the items in the company, a problem will arise. The most appropriate solution to a problem like this is, by creating a database system. This study aims to create a database system to search the verification results for part number of auto vision machine and hypot machine. This database system will be applied to PT Jovan Technologies. In making this database system, researchers use the PHP 7 programming language and use MYSQL as a database of this system. The results of the verification sample part number search are a barcode containing the data from the part number. So that with the existence of this database system, data collection of goods that exist in the company Jovan Technologies becomes more systematic and structured. With the existence of this database system, double production which will result in loss of resources and time at the company can be reduced and avoided.

Keyword: *Auto Vision Machine, Hypot Machine, Database System, Barcode*

KATA PENGANTAR

Puji dan syukur kepada Tuhan Yang Maha Esa yang telah melimpahkan segala rahmat dan karuniaNya, sehingga penulis dapat menyelesaikan laporan tugas akhir yang merupakan salah satu persyaratan untuk menyelesaikan program studi strata satu (S1) pada Program Studi Teknik Informatika Universitas Putera Batam.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna. Karena itu, kritik dan saran akan senantiasa penulis terima dengan senang hati.

Dengan segala keterbatasan, penulis menyadari pula bahwa skripsi ini takkan terwujud tanpa bantuan, bimbingan dan dorongan dari berbagai pihak. Untuk itu, dengan segala kerendahan hati, penulis menyampaikan ucapan terima kasih kepada:

1. Rektor Universitas Putera Batam.
2. Ketua Program Studi Teknik Informatika Universitas Putera Batam.
3. Rahmat Fauzi, S.Kom., M.Kom. selaku pembimbing Skripsi pada Program Studi Teknik Informatika Universitas Putera Batam.
4. Dosen dan Staff Universitas Putera Batam.
5. Kepada Orang Tua penulis selalu mendoakan, dan pengorbanannya baik segi moral maupun materil kepada penulis sehingga dapat menyelesaikan penulisan skripsi ini.
6. Kepada sahabat terbaik penulis yang selalu memberikan motivasi.
7. Teman seperjuangan Program Studi Teknik Informatika Universitas Putera Batam.
8. Kepada pihak PT Jovan Technologies telah bersedia memberikan informasi dan data *part number*.
9. Pihak-pihak yang membantu dalam penyelesaian skripsi ini yang tidak bisa penulis ucapkan satu persatu

Semoga Tuhan membalas kebaikan dan selalu mencurahkan hidayah serta taufikNya, Aamiin.

Batam, 6 September 2019

Penulis

DAFTAR ISI

	HALAMAN
HALAMAN SAMPEL DEPAN	
HALAMAN JUDUL	ii
HALAMAN PERNYATAAN	iii
ABSTRAK	v
ABSTRACT	vi
KATA PENGANTAR	vii
DAFTAR ISI	ix
DAFTAR TABEL	xi
DAFTAR GAMBAR	xii
BAB I PENDAHULUAN	
1.1 Latar Belakang Penelitian	1
1.2 Identifikasi Masalah.....	4
1.3 Pembatasan Masalah.....	5
1.4 Perumusan Masalah	5
1.5 Tujuan Penelitian	6
1.6 Manfaat Penelitian	6
BAB II KAJIAN PUSTAKA	
2.1 Teori Dasar	8
2.2 Variabel	10
2.3 <i>Software</i> Pendukung	13
2.4 Penelitian Terdahulu	29
2.5 Kerangka Pemikiran	33
BAB III METODE PENELITIAN	
3.1 Desain Penelitian	35
3.2 Pengumpulan Data.....	36
3.3 Operasional Variabel	37
3.4 Metode Perancangan Sistem	38
3.5 Lokasi Dan Jadwal Penelitian	44
BAB IV HASIL DAN PEMBAHASAN	
4.1 Hasil Penelitian.....	46
4.2 Pembahasan.....	57
BAB V SIMPULAN DAN SARAN	
5.1 Simpulan	63
5.2 Saran	64
DAFTAR PUSTAKA	
DAFTAR RIWAYAT HIDUP	
SURAT KETERANGAN PENELITIAN	
LAMPIRAN	

DAFTAR GAMBAR

	HALAMAN
Gambar 2.1 Mesin <i>Auto Vision</i>	11
Gambar 2.2 Mesin <i>Auto Hypot</i>	12
Gambar 2.3 PHP.....	13
Gambar 2.4 PHP 7.....	14
Gambar 2.5 <i>MySQL Database</i>	20
Gambar 2.6 Contoh Tabel Informasi.....	21
Gambar 2.7 <i>Unified Modeling Language (UML)</i>	22
Gambar 2.8 Kerangka Pemikiran	33
Gambar 3.1 Desain Penelitian.....	35
Gambar 3.2 <i>Use Case Diagram</i> Aplikasi.....	39
Gambar 3.3 Memasukan <i>Part Number</i>	40
Gambar 3.4 Mencari Verifikasi <i>Part Number</i>	41
Gambar 3.5 <i>Input Part Number</i>	42
Gambar 3.6 Pencarian Hasil Verifikasi	43
Gambar 3.7 <i>Class Diagram</i> Aplikasi.....	44
Gambar 4.1 Tampilan Halaman Utama	46
Gambar 4.2 Halaman <i>Operator</i>	47
Gambar 4.3 Fitur Pencarian	48
Gambar 4.4 Fitur <i>Show Barcode</i>	49
Gambar 4.5 Halaman <i>Login Admin</i>	50
Gambar 4.6 Tampilan <i>Admin Panel</i>	51
Gambar 4.7 Halaman Administrator	52
Gambar 4.8 Halaman Tambah <i>User</i>	53
Gambar 4.9 Halaman Edit Data <i>User</i>	54
Gambar 4.10 Halaman Verifikasi	55
Gambar 4.11 Halaman Tambah Data Barang	56
Gambar 4.12 Halaman Edit Data Barang	57
Gambar 4.13 Halaman Verifikasi	58
Gambar 4.14 Tambah Barang	59
Gambar 4.15 Edit Data Barang	60
Gambar 4.16 Proses Pencarian <i>Part Number</i>	61
Gambar 4.17 <i>Barcode Result</i>	62
Gambar 4.18 Hasil <i>Barcode Reader</i>	63

DAFTAR TABEL

	HALAMAN
Tabel 2.1 Simbol <i>Use Case</i> Diagram	23
Tabel 2.2 Simbol <i>Class</i> Diagram	25
Tabel 2.3 Simbol <i>Activity</i> Diagram	27
Tabel 2.4 Simbol <i>Sequence</i> Diagram	28
Tabel 3.1 Jadwal Penelitian	34

BAB I

PENDAHULUAN

1.1 Latar Belakang Penelitian

Pada zaman modern ini perkembangan teknologi sudah semakin maju terutama dibidang komputer yang semakin canggih. Komputer merupakan suatu alat atau mesin untuk mempermudah aktifitas manusia untuk mengakses segala sesuatu dalam pekerjaan setiap harinya misalnya di dalam rumah, dilingkungan sekolah dan dilingkungan pekerjaan terutama di perusahaan. Perusahaan sekarang ini sangat membutuhkan yang namanya komputer untuk menyelesaikan masalah-masalah yang ada di dalam perusahaan tersebut seperti menyusun data karyawan, mengolah data *input/output* barang, mengolah serta mencari data verifikasi *sample* barang dan lain sebagainya. Ada banyak perusahaan-perusahaan yang ada di batam contohnya di PT JOVAN TECHNOLOGIES mengalami masalah hal yang sama seperti yang sudah diuraikan diatas.

Untuk memproduksi satu jenis barang yang ada di perusahaan PT JOVAN TECHNOLOGIES dari tahap awal mengerjakan sampai selesai berbeda-beda ada yang menggunakan manual (operator sendiri yang mengerjakan) dan ada juga yang menggunakan *auto* (mesin yang mengerjakan) sesuai kebutuhan yang di perlukan untuk menyelesaikan produksi barang tersebut.

Ada beberapa mesin *auto* yang digunakan untuk memproduksi barang di perusahaan PT JOVAN TECHNOLOGIES diantaranya mesin *Vision* dan mesin *Hypot*. Fungsi kedua mesin ini berbeda satu sama lain. Fungsi mesin *vision* ini

untuk melakukan pengecekan apakah barang yang di kerjakan proses sebelumnya sudah OK atau belum, sedangkan fungsi mesin *Hypot* untuk melakukan pengecekan *short* atau tidaknya satu barang tersebut. Sebelum dijalankan kedua mesin tersebut harus melakukan *sample* verifikasi terlebih dahulu untuk mengetahui apakah mesin itu sudah berfungsi dengan baik sesuai dengan *Part Number* yang dijalankan dan sesuai *Work Intruction (WI)*. *Part Number* merupakan suatu identitas barang dari beberapa banyaknya model barang yang di produksi di perusahaan PT JOVAN TECHNOLOGIES. Di dalam *sample* verifikasi terdapat beberapa jenis *reject* yang dideteksi oleh kedua mesin di atas.

Untuk mencari *sample* verifikasi tidaklah mudah dikarenakan terlalu banyaknya model *part number* yang dijalankan sehingga membutuhkan tenaga manusia untuk mencari di tempat penyimpanan yang sudah disediakan, serta ada perselisihan antara *technician* dengan *leader* apabila ada *part number* yang sudah lama tidak di jalankan dan di produksi ulang lagi. Perselisihan yang dimaksud adalah *leader* menyuruh *technician* untuk membuat *sample* verifikasi baru supaya *part number* yang sudah lama tidak di produksi itu bisa di jalankan pada kedua mesin diatas, dan *technician* tidak mengikuti apa yang di intrusikan oleh *leader* di karenakan *sample* verifikasi *part number* yang dimaksud sudah dibuat sebelumnya. Disinilah terjadi perdebatan antara *leader* dan *technician*, keduanya saling mempertahankan pendapat masing-masing. Menurut pemikiran *leader* *sample* verifikasi yang dimaksud belum dibuat dan harus dibuat baru, penyebabnya adalah *leader* tidak mau mencari *sample* verifikasi yang lama itu dikarenakan terlalu banyaknya model *sample* verifikasi yang di produksi dan

memakai waktu untuk mencari di tempat penyimpanannya. Menurut pemikiran *technician sample* verifikasi yang dimaksud tidak perlu dibuat lagi dikarenakan sudah dibuat sebelumnya. Dan apabila dibuat ulang lagi terjadi yang namanya *double* pembuatannya maksudnya ini *sample* verifikasi yang sudah lama dibuat dan *sample* verifikasi yang baru dibuat, sehingga terjadi pemborosan barang dan tempat penyimpanannya.

Dari permasalahan yang sudah diuraikan diatas, penerapan sistem *database* yang tepat untuk mengolah data *sample* verifikasi sehingga para penggunanya bisa lebih efektif dan efisien dalam menggunakannya. *Database* merupakan sekumpulan beberapa data dan informasi yang tersusun dan tersimpan di dalam sebuah komputer secara sistematis yang dapat diperiksa, diolah atau dimanipulasi. *Database* merupakan sebuah sistem yang dapat digunakan untuk menyimpan informasi terstruktur dimana informasi tersebut disusun dan disimpan sedemikian sehingga bisa diambil dengan mudah dan efisien (Enterprise, 2014). Ada beberapa bagian yang diperlukan untuk membuat sebuah sistem *database* diantaranya, *MySQL* yang fungsinya untuk pengelolaan data dan *PHP* untuk menginput program di dalam sebuah *database*. Dengan menggunakan sistem *database* masalah yang terjadi di PT JOVAN TECHNOLOGIES bisa di aplikasikan untuk melakukan pencarian serta mengolah data sampel verifikasi tersebut. Berdasarkan permasalahan tersebut, ada beberapa perusahaan sekarang ini memiliki keinginan untuk memanfaatkan teknologi informasi secara baik melalui pembangunan sistem persediaan barang yang salah satunya rancang bangun sistem informasi pengolahan data persediaan barang berbasis *desktop*

dengan model *waterfall* (Nawang, Kurniawati, & Duta, 2017). Jadi dengan mengimplementasikan sebuah sistem pada perusahaan segala sesuatu pekerjaannya semakin cepat, efektif dan efisien.

Berdasarkan permasalahan yang telah dipaparkan di atas, peneliti mengangkat judul penelitian dengan judul “**Sistem Database Untuk Mencari Sample Verifikasi Part Number Pada Perusahaan Produksi Kota Batam**”.

1.2 Identifikasi Masalah

Berdasarkan latar belakang permasalahan di atas, ada beberapa poin-poin permasalahan yang akan menjadi fokus penelitian sebagai berikut :

1. Perusahaan PT JOVAN TECHNOLOGIES memiliki permasalahan untuk mencari sampel verifikasi pada mesin produksinya yang berupa mesin *Vision* dan mesin *Hypot*.
2. Masih ditemukan kesulitan dalam mencari *sample verifikasi part number*, dikarenakan terlalu banyaknya *part number* yang dijalankan setiap mesin produksinya.
3. Perdebatan antara *technician* dan *leader* untuk membuat *sample verifikasi*, apabila ada barang yang sudah lama tidak dijalankan dan diproduksi ulang lagi.
4. Terjadi pemborosan barang dan tempat apabila *sample verifikasinya double* pembuatannya.
5. Tidak adanya sebuah sistem untuk mencari dan mengolah data *sample verifikasi part number* di perusahaan PT JOVAN TECHNOLOGIES.

1.3 Pembatasan Masalah

Beberapa hal yang dijadikan pembatasan masalah agar penelitian ini tidak menyimpang dan hasil sesuai target. Adapun batasan masalah dari penelitian ini sebagai berikut :

1. Penelitian ini dilakukan di perusahaan PT JOVAN TECHNOLOGIES.
2. Penelitian ini mengambil data dari *sample* verifikasi *part number* pada mesin auto *Vision* dan *Hypot*.
3. Aplikasi yang digunakan untuk membuat *database* dalam penelitian ini *MySQL*.
4. Sistem *database* ini berupa sebuah pencarian dan mengolah data *sample* verifikasi mesin auto *Vision* dan mesin *Hypot*.
5. Untuk penginputan program di dalam *database* peneliti memakai PHP sehingga nantinya sistem *database* ini berbasis web.

1.4 Perumusan Masalah

Berdasarkan latar belakang yang sudah dijelaskan di atas, dapat dirumuskan permasalahan yang akan diselesaikan di dalam penelitian ini adalah :

1. Bagaimana mengatasi permasalahan untuk mencari *sample* verifikasi pada mesin produksi PT.JOVAN TECHNOLOGIES yang berupa mesin *Vision* mesin *Hypot*?
2. Permasalahan apa aja yang terjadi pada perusahaan PT JOVAN TECHNOLOGIES untuk mengolah data *sample* verifikasi *part numbernya*?

3. Bagaimana cara kerja sebuah sistem untuk mencari dan mengolah data *sample* verifikasi *part number* di perusahaan PT JOVAN TECHNOLOGIES?

1.5 Tujuan Penelitian

Berdasarkan perumusan masalah yang sudah dipaparkan di atas, adapun tujuan dari penelitian ini sebagai berikut :

1. Menerapkan sistem *database* untuk melakukan pencarian dan mengolah data *sample* verifikasi *part number* pada perusahaan PT JOVAN TECHNOLOGIES.
2. Mengurangi permasalahan yang terjadi di PT JOVAN TECHHOLOGIES dalam mengolah data *sample* verifikasi *part number* pada setiap mesin produksinya.
3. Mengetahui cara kerja sebuah sistem untuk mencari dan mengolah data *sample* verifikasi *part number* pada perusahaan PT JOVAN TECHNOLOGIES.

1.6 Manfaat Penelitian

Dengan tercapainya tujuan dari penelitian, maka diharapkan penelitian ini bermanfaat bagi berbagai pihak. Manfaat dari penelitian ini adalah :

1.6.1 Bagi Perusahaan PT JOVAN TECHNOLOGIES

Manfaat penelitian bagi objek penelitian adalah :

1. Mempercepat mencari dan mengolah data *sample* verifikasi *part number* pada perusahaan PT JOVAN TECHNOLOGIES.
2. Pihak terkait yang berhadapan langsung dengan pengolahan data *sample* verifikasi *part number* menjadi efektif dan efisien.

1.6.2 Bagi Peneliti

Manfaat penelitian bagi peneliti adalah :

1. Penelitian ini dapat dijadikan peningkatan kemampuan dari peneliti dalam menggunakan sistem *database* di dalam dunia perusahaan untuk mencari dan mengolah data menjadi lebih baik.
2. Sebagai syarat dalam tugas akhir dan syarat pengambilan gelar Sarjana.

BAB II

KAJIAN PUSTAKA

2.1 Teori Dasar

2.1.1 Interaksi Manusia dan Komputer

IMK adalah salah satu bidang ilmu yang biasa mempelajari interaksi antara manusia dan komputer. IMK mempelajari bagaimana mendesain, mengimplementasikan sistem komputer yang interaktif agar dapat digunakan oleh manusia dengan mudah. Sebuah program aplikasi terdiri atas 2 bagian utama, diantaranya adalah :

1. Bagian Antarmuka (*Interface*)
2. Bagian Aplikasi

Menurut (Santosa, 2010) Interaksi manusia dan Komputer adalah suatu rancangan sistem komputer yang mendukung/membantu manusia. IMK bertujuan untuk mempermudah manusia dalam mengoperasikan komputer dan mendapatkan berbagai umpan balik yang manusia perlukan selama bekerja pada sebuah sistem komputer. Sebagai contoh, misalnya sebuah komputer lengkap dipasang pada sebuah tempat yang tidak nyaman bagi seorang pengguna yang menggunakan.

A. Komponen IMK

1) *User* (Pengguna)

Pengguna individu, group dari pengguna yang bekerja sama atau sekelompok pengguna di dalam organisasi dimana masing-masing berhubungan dengan bagian yang sama dari suatu pekerjaan atau proses yang berlangsung. Pengguna berusaha menyelesaikan tugasnya dengan bantuan dari teknologi (komputer).

2) Komputer

Teknologi yang mengontrol suatu proses atau sebuah sistem mulai dari komputer yang berskala kecil / *desktop* sampai dengan komputer dengan skala yang besar seperti super komputer atau suatu teknologi yang dapat dicangkokkan atau ditanamkan (*embedded system*) terhadap benda.

3) Interaksi

Setiap komunikasi yang dilakukan antara pengguna (*user*) dengan teknologi (komputer) secara langsung atau secara tidak langsung. Interaksi tersebut dapat berupa sebuah dialog timbal balik (*feed-back*) atau kontrol dari kinerja suatu pekerjaan. Interaksi secara tidak langsung melibatkan background atau *batch-processing*.

2.1.2 Sistem

Sistem adalah sebuah tata perintah yang tersusun secara sistematis sehingga dapat digunakan untuk melakukan sebuah pekerjaan berdasarkan perintah yang dimasukkan. Menurut (Huda, Winarno, & Lutfi, 2017) sistem dapat didefinisikan sebagai kumpulan dari komponen yang saling berhubungan satu dengan yang lainnya

membentuk satu kesatuan untuk mencapai tujuan tertentu. Sistem adalah sekelompok bagian-bagian alat dan sebagainya yang bekerja bersama-sama untuk melakukan sesuatu maksud. Beberapa aksi *user* yang sesuai dalam hubungannya dengan sebuah sistem antara lain (Sudarmawan & Ariyus, 2007).

1. Formalisasi Standar

Notasi rekayasa dari *software* yang digunakan untuk menjelaskan perilaku yang dibutuhkan oleh sistem interaktif tertentu yang berbasis dengan model.

2. Model Interaksi

Model matematika yang memiliki tujuan tertentu di dalam sistem interaktif yang digunakan untuk menjelaskan kegunaan properti.

3. Perilaku Terus-Menerus

Aktivitas antara *event*, objek dengan gerakan yang konstan.

- A. Tipe Model Sistem

1. *Spesific System* : dialog sebagai mode utama dan definisi kondisi penuh
2. *Generic Issues* : model interaksi abstrak

2.2 Variabel

Dalam penelitian ini, peneliti mengambil 2 variabel sebagai fokus dalam penelitian ini. Variabel yang menjadi fokus dalam penelitian ini antara lain :

1. Mesin *auto Vision*

Fungsi mesin *vision* ini digunakan untuk melakukan proses pengecekan barang apakah yang di kerjakan sebelumnya sudah “OK” atau belum. Adapun bagian-bagian

yang dicek mesin *auto vision* pada barang produksi di PT.JOVAN TECHNOLOGIES

diantaranya :

- 1) *Missing signal terminal*
- 2) *Missing power terminal*
- 3) *Wrong configuration signal terminal*
- 4) *Wrong configuration power terminal*
- 5) *Power gap out maximal*
- 6) *Wrong aligner*
- 7) *Wrong void power terminal*
- 8) *Good part*



Gambar 2.1 Mesin *Auto Vision*
Sumber Data Olahan Peneliti 2019

2. Mesin *auto Hypot*

Fungsi mesin *Hypot* ini digunakan untuk melakukan pengecekan *short* apa tidaknya barang tersebut. Adapun bagian-bagian yang dicek mesin *auto Hypot* pada barang produksi di PT.JOVAN TECHNOLOGIES diantaranya :

- 1) *1st row short circuit*
- 2) *2nd row short circuit*
- 3) *3rd row short circuit*
- 4) *4th row short circuit*
- 5) *5th row short circuit*
- 6) *1st with 2nd row short circuit*
- 7) *2nd with 3rd row short circuit*
- 8) *3rd with 4th row short circuit*
- 9) *4th with 5th row short circuit*
- 10) *Good part*



Gambar 2.2 Mesin *Auto Hypot*
Sumber Data Olahan Peneliti 2019

2.3 Software Pendukung

2.3.1 PHP 7



Gambar 2.3 PHP
Sumber Data Olahan Peneliti 2019

PHP banyak digunakan oleh *programmer* dalam membuat sebuah *website*. Menurut (Enterprise J. , 2017) PHP digunakan untuk membuat *website* dengan tampilan statis dan dinamis. Pada dasarnya, bahasa pemrograman yang sangat mendasar dipakai oleh para *programmer* adalah HTML, namun HTML memiliki kekurangan. HTML hanya dapat menciptakan *website* yang statis saja, sedangkan PHP dapat menciptakan sebuah *website* yang statis maupun dinamis.

User banyak mengharapkan sebuah *website* memiliki tampilan yang statis dan dinamis. *User* akan memilih sebuah *website* yang memiliki tampilan yang menarik daripada *website* yang memiliki tampilan yang monoton. Maksud dari *website* yang

dinamis itu sendiri adalah, tampilan *website* tersebut dapat berubah-ubah dengan menyesuaikan kondisi lingkungannya. Secara *default*, *file* PHP memiliki ekstensi *.php*, sehingga apabila saat *server* web menemukan sebuah *file* dengan ekstensi tersebut, *file* dengan jenis ekstensi *.php* akan langsung diproses oleh prosessor PHP.

Untuk membuat sebuah *file* PHP, pada awal bagian isi *file* tersebut harus dibuka dengan *tag* `<?php` dan ditutup dengan *tag* `?>`, sehingga semua isi yang terdapat diantara *tag* tersebut akan diproses oleh prosessor PHP. Perbedaan lainnya antara PHP dengan HTML adalah penggunaan *tag*, dimana untuk HTML *tag* yang digunakan hanya dapat 1 kali tiap *file* HTML, sedangkan pada *file* PHP, *tag* PHP dapat dipanggil beberapa kali dalam 1 *file*.



Gambar 2.4 PHP 7
Sumber Data Olahan Peneliti 2019

Saat ini PHP telah masuk ke versi PHP 7 dimana PHP 7 hanya dapat membaca *coding database* MySQLi. PHP 7 adalah penyempurnaan dari versi PHP sebelumnya

(PHP 5). Saat ini *browser-browser* yang ada juga sudah menyesuaikan *engine* nya untuk *support* dengan PHP 7 yang diyakini lebih ringan daripada PHP 5.

Didalam PHP terdapat tipe-tipe data yang dapat dibaca oleh prosessor PHP. Tipe data tersebut diantaranya adalah, tipe data *integer*, tipe data *float*, tipe data *string* dan tipe data *boolean*. Berikut adalah penjelasan dari masing-masing tipe data yang ada pada PHP.

A. Tipe Data *Integer*

Menurut (Enterprise J. , 2017) menyatakan bahwa tipe data *integer* adalah semua data-data yang memiliki angka bulat, mulai dari *range* angka positif sampe *range* angka *negative*. *Range* nilai positif pada tipe data *integer* adalah 2.147.483.647 dan *range* nilai *negative* pada tipe data *integer* adalah -2.147.483.647. Untuk mengecek nilai maximal pada tipe data *integer* dapat menggunakan perintah sebagai berikut.

```
<?php
    Print PHP_INT_MAX;
?>
```

Penggunaan tipe data *integer* adalah untuk permasalahan perhitungan, namun yang tidak menghasilkan angka berkoma (*decimal*). Untuk dapat menampilkan angka berkoma (*decimal*) dapat menggunakan tipe data lainnya, yaitu tipe data *float*.

B. Tipe Data *Float*

Tipe data *float* diciptakan untuk mengatasi kekurangan dari tipe data *integer*. Kekurangan tipe data *integer* adalah tidak dapat menampilkan nilai berkoma

(*decimal*). Tipe data *float* dapat menampilkan hasil dengan bilangan berkoma (*decimal*) berbeda dengan tipe data *integer* yang hanya mampu menampilkan bilangan bulat. Saat menulis angka pecahan, pada tipe data *float* menggunakan symbol titik (.) untuk menandakan bahwa bilangan tersebut adalah bilangan berkoma (*decimal*).

Selain dengan menggunakan angka berkoma (*decimal*), *programmer* juga dapat menggunakan angka dalam bentuk E notasi untuk menyingkat *script*.

C. Tipe Data *String*

Tipe data *string* ditujukan untuk penggunaan teks / tulisan. Tipe data *string* dapat menampung segala jenis *symbol* dan *alphabet* yang ada, itu dikarenakan tipe data ini akan membaca semua hasil masukan menjadi sebuah teks / tulisan. Penulisan *string* pada tipe data ini dibagi menjadi 4 macam, yaitu *single quoted* (tanda kutip tunggal), *double quoted* (tanda kutip ganda), *heredoc*, dan *nowdoc*.

1) *Single Quoted*

Single quoted memanfaatkan kutip tunggal untuk menuliskan *string* pada *coding* PHP. Contoh penggunaan *single quoted string* adalah sebagai berikut.

```
<?php
```

```
    Echo 'Pemograman PHP';
```

```
?>
```

Berdasarkan *coding* tersebut maka *browser* akan menampilkan tulisan “Pemograman PHP”. Apabila *programmer* ingin menampilkan tanda kutip satu pada *string* maka *programmer* dapat menggunakan *backslash* (\) sedangkan apabila

programmer ingin menampilkan tanda garis miring, maka *programmer* harus menggunakan *double backslash* (\).

2) *Double Quoted*

Double quoted adalah teknik penulisan *string* dengan cara menggunakan tanda kutip ganda. Perbedaan dengan *single quoted* adalah posisi penggunaannya. *Single quoted* digunakan untuk penulisan *string* biasa, sedangkan *double quoted* digunakan untuk penulisan *string* yang memiliki *variable*. Contoh penulisan *coding* PHP untuk *double quoted* adalah sebagai berikut.

```
<?php  
  
    $teks = "Pemograman PHP";  
  
    Echo $teks;  
  
?>
```

Berdasarkan *coding* tersebut, maka *browser* akan menampilkan teks "Pemograman PHP".

3) *Heredoc*

Heredoc adalah sebuah fitur yang disediakan oleh PHP untuk menampilkan sebuah teks / tulisan. Penggunaan *heredoc* memiliki keuntungan, yaitu dapat membuat tulisan dengan beberapa baris dengan leluasa, namun penggunaan *heredoc* membutuhkan tata cara penulisan tertentu yang unik yaitu penggunaan tanda <<<. Contoh penulisan *coding* dengan fitur *heredoc* adalah sebagai berikut.

```
<?php  
  
    $nama = "Ahmad Riyandi";
```

```
$kalimat = <<<TULIS
```

```
$nama adalah seorang mahasiswa teknik informatika
```

```
Pada salah satu Universitas Negeri di Batam.
```

```
TULIS;
```

```
Echo $kalimat;
```

```
?>
```

Maka pada *browser* akan menampilkan teks “Ahmad Riyandi adalah seorang mahasiswa teknik informatika Pada salah satu Universitas Negeri di Batam. Berikut adalah beberapa aturan dalam penggunaan *heredoc*.

- *Heredoc* diawali dengan penulisan *symbol* <<<
- Kemudian dilanjutkan dengan penulisan sebagai penanda *string*, sebagai contoh adalah TULIS
- Akhiri penulisan penanda *string* dengan *symbol* titik koma (;)
- Setelah menulis <<<TULIS, *programmer* harus langsung menekan *enter* untuk meyakinkan tidak ada spasi setelahnya
- Setelah menulis TULIS; harus langsung menekan *enter* untuk memastikan tidak ada spasi setelah kalimat tersebut.

4) *Nowdoc*

Nowdoc adalah tipe panjang dari *single quoted*, karena *nowdoc* memiliki karakteristik yang sama dengan *single quoted*. Dalam menulis *string*, *nowdoc*

menggunakan struktur *heredoc* dengan menggunakan kutip tunggal (*single quoted*).

Contoh penulisan *string* dengan *nowdoc* adalah sebagai berikut.

```
<?php
```

```
    $nama = "Ahmad Riyandi";
```

```
    $kalimat = <<< 'TULIS'
```

```
        $nama adalah seorang mahasiswa teknik informatika
```

```
        Pada salah satu Universitas Negeri di Batam.
```

```
    TULIS;
```

```
    Echo $kalimat;
```

```
?>
```

D. Tipe Data *Boolean*

Tipe data *boolean* adalah tipe data pada PHP yang memiliki ciri khas. Berbeda dengan tipe data lainnya yang boleh memasukan angka atau *symbol* maupun huruf dengan bebas, tipe data *boolean* hanya memperbolehkan memasukan teks *True* atau *False* saja.

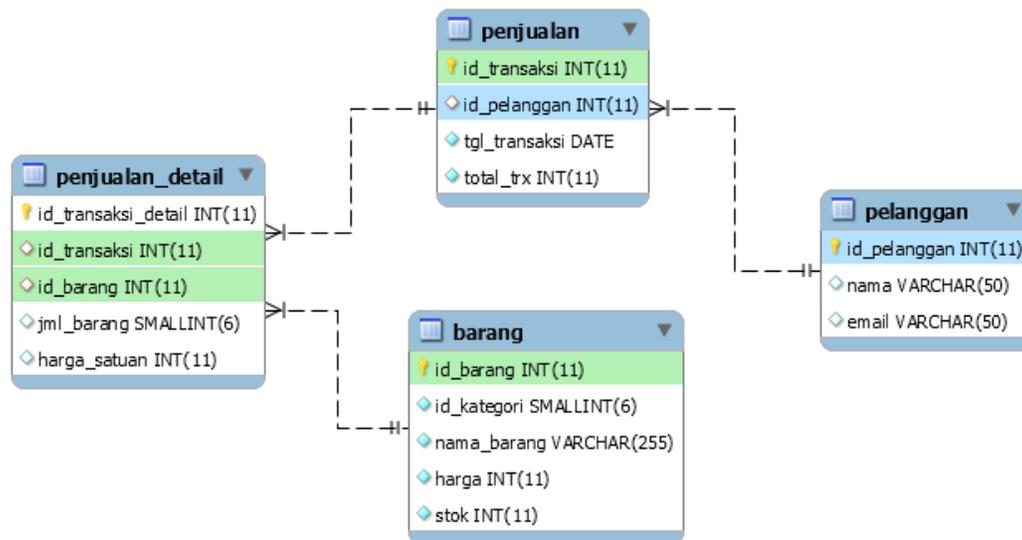
Tipe data *boolean* banyak digunakan untuk pemograman yang menggunakan pernyataan kondisional (*If ... Then*) dan juga pengulangan (*Looping*).

2.3.2 MySQL Database



Gambar 2.5 *MySQL Database*
Sumber Data Olahan Peneliti 2019

Database adalah sebuah tempat penyimpanan dalam pemrograman. Menurut (Wibowo & Enterprise, 2014) mengemukakan bahwa *Database* adalah sebuah sistem yang digunakan dalam penyimpanan informasi yang terstruktur dimana informasi tersebut disimpan dan disusun agar memudahkan dalam pengambilan informasinya. Informasi tersebut dipecah dalam bentuk tabel seperti gambar di bawah ini.



Gambar 2.6 Contoh Tabel Informasi
Sumber Data Olahan Peneliti 2019

Setiap tabel menyimpan data yang berbeda dengan tabel lainnya. *Programmer* juga dapat mengatur hubungan atau *relationship* antar tabel. *Relationship* memungkinkan *user* untuk dapat menggabungkan beberapa informasi menjadi beberapa tabel. Sebuah *database* yang bergantung pada *relationship* antar tabel disebut dengan *relational database* yang direpresentasikan dalam bentuk visual diagram. Sebuah *program* pembantu yang dapat mengolah informasi yang ada di dalam tabel *database* tersebut adalah sebuah *database management system* (DBMS). DBMS akan mengatur informasi di dalam *database* dengan *statement* yang telah ditulis sebelumnya di dalam *Structure Query Language* (SQL).

Instruksi SQL (*Structure Query Language*) dikelompokan berdasarkan jenis dan fungsinya masing-masing. Berikut merupakan 3 perintah dasar dari SQL diantaranya adalah :

1. *Data Definition Language*
2. *Data Manipulation Language*
3. *Data Control Language*

2.3.3 UML (*Unified Modeling Language*)



Gambar 2.7 *Unified Modeling Language (UML)*
Sumber Data Olahan Peneliti 2019

UML atau *Unified Modelling Language* adalah salah satu standar bahasa yang banyak digunakan dalam dunia industri untuk mendefinisikan kebutuhan, pembuatan analisis dan desain, serta untuk merepresentasikan arsitektur dalam pemograman berbasis objek (OOP) (S & Salahuddin, 2013). Terdapat beberapa jenis diagram yang dapat digunakan dalam merepresentasikan sebuah sistem dalam *software*, berikut penjelasan dari beberapa jenis diagram di dalam UML.

A. *Use Case Diagram*

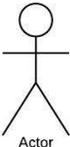
Use case diagram adalah pemodelan yang dikhususkan untuk kelakuan (*behavior*) dari sistem informasi yang akan dibuat. *Use case* diagram akan

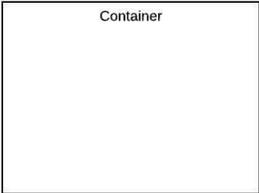
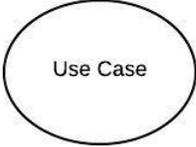
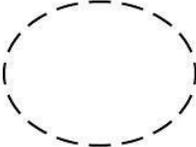
mendeskripsikan interaksi antar satu aktor atau beberapa aktor. Ada 2 hal utama pada *use case* yaitu pendefinisian dari aktor dan pendefinisian dari *use case*.

1. Aktor merupakan orang, proses atau sistem yang berinteraksi dengan sistem yang akan dibuat.
2. *Use case* merupakan fungsionalitas yang disediakan oleh sistem.

Berikut simbol-simbol yang terdapat di dalam *use case* diagram.

Tabel 2.1 Simbol *Use Case* Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		Aktor	Merupakan orang atau proses yang ada di dalam sistem
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada <i>dependent element</i> akan mempengaruhi <i>independent element</i>
3		<i>Generalization</i>	hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>)
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber

			pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Sumber Data Olahan Peneliti 2019

B. Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat di dalam sistem. Diagram kelas dibuat agar *programmer* membuat kelas-kelas sesuai rancangan yang ada di dalam diagram kelas agar antara dokumentasi dengan implementasi pembuatan sistem terdapat kecocokan. Susunan kelas yang baik pada diagram kelas sebaiknya memiliki jenis kelas sebagai berikut.

1. Kelas *main*

Kelas yang memiliki fungsi awal ketika sistem dijalankan.

2. Kelas *view*

Kelas yang mengatur tampilan kepada pemakai sistem.

3. Kelas *controller*

Kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian *use case*.

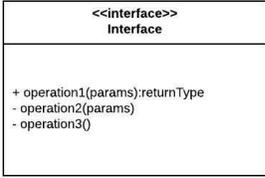
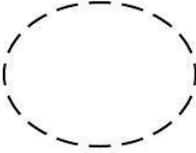
4. Kelas *model*

Kelas yang digunakan untuk membungkus atau menyatukan data menjadi satu kesatuan yang diambil atau disimpan di dalam *database*.

Berikut simbol-simbol yang terdapat di dalam diagram kelas.

Tabel 2.2 Simbol *Class* Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek

			induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Association</i>	Yang menghubungkan antara objek satu dengan objek lainnya

Sumber Data Olahan Peneliti 2019

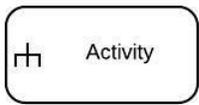
C. *Activity Diagram*

Diagram aktivitas digunakan untuk menggambarkan alur kerja atau aktivitas dari kerja sebuah sistem atau proses bisnis atau menu yang terdapat di dalam perangkat lunak. Fokus dalam diagram aktivitas ini adalah aktivitas dari sistem yang ada bukan aktivitas dari aktor yang terdapat di dalam sistem. Diagram aktivitas banyak digunakan untuk mendefinisikan hal-hal sebagai berikut.

1. Rancangan pada proses bisnis dimana setiap aktivitas bisnis yang digambarkan merupakan proses bisnis dari sistem yang didefinisikan.
2. Pengelompokan tampilan dari sistem (UI) dimana setiap aktivitas yang terjadi memiliki tampilan sendiri.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
4. Rancangan menu yang ditampilkan pada *software*.

Berikut adalah simbol-simbol yang terdapat di dalam diagram aktivitas.

Tabel 2.3 Simbol *Activity Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	<i>State</i> dari sistem yang mencerminkan eksekusi dari suatu aksi

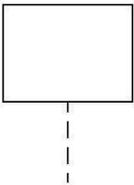
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

Sumber Data Olahan Peneliti 2019

D. *Sequence Diagram*

Secara mudahnya *sequence diagram* adalah gambaran tahap demi tahap, termasuk kronologi (urutan) perubahan secara logis yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan *use case diagram*. Dalam *sequence diagram* terdapat simbol-simbol yang merepresentasikan kegiatan yang digambarkan, berikut adalah penjelasan dari simbol-simbol yang ada di dalam *sequence diagram*.

Tabel 2.4 Simbol *Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
---	---	----------------	--

Sumber Data Olahan Peneliti 2019

2.4 Penelitian Terdahulu

Dalam melakukan penelitian untuk membuat sistem *database* untuk mencari *sample* verifikasi *part number*, peneliti menggunakan beberapa jurnal terdahulu sebagai referensi dalam penelitian ini.

1. Abdul Mufti (2015). “**Rancangan Layar Sebagai Alat Bantu Pendewasa Interaksi Manusia dan Komputer**”. ISSN: 1979-276X. Perkembangan teknologi komputer diikuti dengan pengembangan *interface* (antar muka), antar muka perangkat lunak meliputi ragam dialog sebuah program aplikasi komputer. Untuk membentuk program aplikasi komputer seorang *programer/* analis harus membuat tampilan yang menarik dan baik untuk digunakan. Tujuan penulisan ini untuk memberi gambaran tentang rancangan layar yang baik dan terstruktur. Penelitian ini merupakan penelitian deskriptif yaitu suatu metode penelitian yang ditujukan untuk menggambarkan fenomena-fenomena yang ada, yang berlangsung saat ini atau saat yang lampau. Para perancang antar muka (*interface*) banyak yang kurang memperhatikan langkah-langkah dalam merancang layar sesuai langkah-langkah yang dapat mendewasakan

tampilan, sedangkan rancangan layar yang baik yaitu adalah rancangan yang sesuai dengan kebutuhan pengguna dan dapat dijadikan referensi kedepan untuk dijadikan sebagai alat bantu pengembangan/ pendewasa Interaksi Manusia dengan Komputer (IMK).

2. Sucipto (2017). “**Perancangan Active Database System Pada Sistem Informasi Pelayanan Harga Pasar**”. ISSN: 2549-6824. Sistem informasi merupakan sistem yang berkaitan dengan adanya pemrograman dan *database*. Sebuah sistem informasi yang baik bersifat dinamis. Sistem informasi yang dinamis tidak mungkin terlepas dengan adanya *database* sebagai tempat penyimpanan data. Pada umumnya *database* digunakan secara pasif sehingga fungsinya hanya untuk penyimpanan data, padahal *database* juga dapat sebagai pengendalian sebuah sistem informasi. Beberapa *database open source* sudah mendukung untuk pengendalian sistem atau yang dapat disebut juga *active database system* seperti *MySQL*, *MariaDB* dan *PosgreSQL*. *Active database system* memberikan fungsi *database* untuk mendapatkan perilaku reaktif dan melakukan pengendalian sistem informasi pada tingkat *database*. Proses bisnis yang biasanya dilakukan oleh bahasa pemrograman sebagai aturan dimasukkan diaplikasi dapat dimasukkan langsung ke dalam *database*. Penelitian ini dilakukan untuk perancangan *active database* sistem terhadap pelayanan harga pasar. Pendekatan *active database system* ini dirancang guna mendapatkan sistem pelayanan aplikatif dan *responsive* untuk berkolaborasi dengan sistem kendali pada modern.

3. Muhammad Faizal dan Sanda Listya Putri (2017). “**Sistem Informasi Pengolahan Data Pegawai Berbasis Web**”. ISSN: 2252-4517. Penggunaan teknologi komputer khususnya dalam hal pengolahan data pada saat ini sangat diperlukan. Baik untuk kepentingan perorangan maupun suatu instansi termasuk pada Sistem Informasi Pengolahan data Pegawai di PT Perkebunan Nusantara VIII (Tambaksari). Proses pengolahan data pegawai ini memerlukan suatu pencatatan, penyimpanan, serta pelaporan tentang data-data pegawai . Dari permasalahan diatas maka dibuatlah Sistem Informasi Pengolahan data Pegawai di PT Perkebunan Nusantara VIII (Tambaksari) dengan menggunakan bahasa pemrograman PHP dan *database MySQL*. Adapun pembuatan sistem ini menggunakan beberapa metode untuk pengumpulan data, metode ini meliputi observasi dan wawancara kepada pihak instansi guna memenuhi data untuk kebutuhan sistem yang akan dirancang. Perancangan sistem dilakukan dengan metode *waterfall* yaitu analisa kebutuhan, desain sistem, penulisan kode program, pengujian program, dan penerapan program serta pemeliharaan program. Dengan dibuatnya sistem atau program aplikasi yang sudah terkomputerisasi ini, pengolahan data pegawai dan informasi data pegawai ini dapat berjalan dengan baik serta dapat memenuhi kebutuhan pihak-pihak yang membutuhkan informasi.
4. Natalie Segure Velandia, Ruben D. Hernandez Beleno, dan Robinson Jimenez Moreno (2017). “*Application of Depth Neural Netwarok*”. ISSN: 1997-5422. Penelitian ini menjelaskan tentang metode komputasi yang diimplementasikan

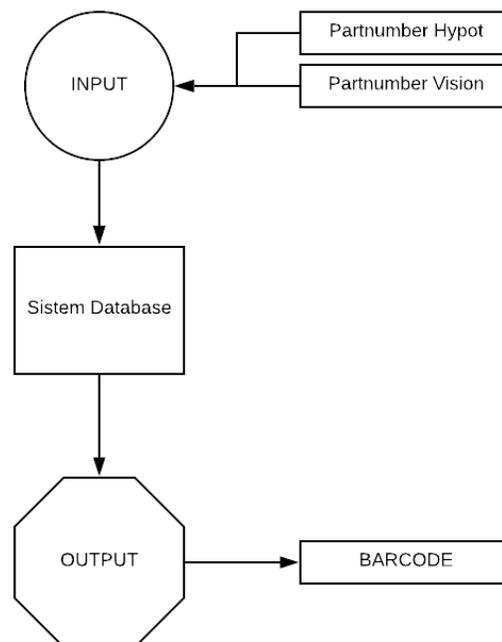
dengan teknik pemahaman yang menjelaskan tentang arsitektur dari jaringan syaraf. Berdasarkan metode tersebut dipergunakan untuk mengidentifikasi dari gambar dan pola, setiap karakter dalam kegiatan. Penelitian ini membangun sebuah aplikasi yang menggantikan sebuah jaringan syaraf yang terdapat pada manusia yang diimplementasikan kedalam sebuah aplikasi yang terkomputerisasi.

5. Innocent Mapanga dan Prudence Kadebu (2013). **“Database Management Systems: A NoSQL Analysis”**. ISSN: 2321-0850. Generasi baru aplikasi seperti intelijen bisnis, analitik perusahaan, Manajemen Hubungan Pelanggan, pemrosesan dokumen, Jejaring Sosial, Web 2.0 dan *Cloud Computing* membutuhkan penskalaan horizontal ribuan *node* seperti yang diminta saat menangani koleksi besar kumpulan data terstruktur dan tidak terstruktur yang gagal RDBMS tradisional gagal untuk mengelola. Tingkat di mana data dihasilkan melalui aplikasi interaktif oleh sejumlah besar pengguna secara bersamaan dalam pemrosesan terdistribusi yang melibatkan sejumlah besar *server* dan penanganan aplikasi *Big Data* telah melampaui kemampuan *database* relasional sehingga mendorong fokus kearah adopsi basis data NoSQL. Sistem *database* NoSQL telah menangani penskalaan dan tantangan kinerja yang melekat dalam RDBMS tradisional dengan mengeksploitasi partisi, melonggarkan protokol konsistensi yang ketat dan melalui sistem terdistribusi yang dapat menjangkau pusat data sambil menangani skenario kegagalan tanpa hambatan. Dalam makalah ini dibahas berbagai sistem

manajemen basis data dan prinsip-prinsip desain dasar mereka yaitu ACID, CAP dan teorema BASE, dievaluasi.

2.5 Kerangka Pemikiran

Di dalam melakukan penelitian, peneliti melakukan argumentasi sementara terhadap permasalahan yang diangkat. Kerangka berpikir digunakan untuk menjelaskan secara sementara terhadap suatu gejala yang menjadi objek permasalahan. Kerangka pikir penelitian ini disajikan dalam bentuk diagram sebagai berikut :



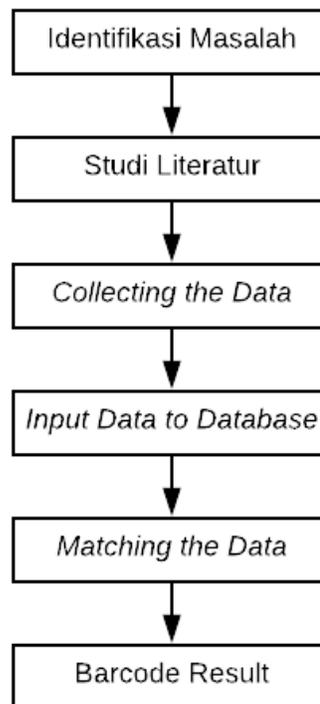
Gambar 2.8 Kerangka Pemikiran
Sumber Data Olahan Peneliti 2019

Penjelasan dari gambar diatas adalah, *part number* dari mesin *hypot* dan mesin *vision* akan dimasukkan kedalam *database* ini. *Part number* tersebut akan disimpan didalam *database* beserta data-data nya, seperti nama teknisi, jam masuk barang, hingga lokasi tempat barang tersebut disimpan. Hasil atau *output* dari sistem *database* ini adalah sebuah *barcode* yang memuat data-data dari barang tersebut. Sehingga operator yang disini sebagai *user* dapat mencari posisi barang tersebut sesuai data-data yang ada pada *barcode* hasil *output* dari sistem *database* ini.

BAB III METODE PENELITIAN

3.1 Desain Penelitian

Dalam melakukan penelitian, peneliti membuat sebuah desain penelitian yang merepresentasikan seluruh kegiatan yang dilakukan peneliti dalam melakukan penelitian untuk membuat sebuah sistem dari awal penelitian sampai akhir dari penelitian. Desain penelitian ini akan direpresentasikan di dalam gambar dibawah ini.



Gambar 3.1 Desain Penelitian
Sumber Data Olahan Peneliti 2019

1. Peneliti melakukan identifikasi permasalahan sebelum memulai penelitian, dimana ditemukan kesulitan dalam pencarian sampel verifikasi terhadap mesin *hypot* dan *vision* yang ada di PT Jovan Technologies.
2. Peneliti mengumpulkan dan membaca sumber referensi yang terkait terhadap bidang penelitian yang akan dilakukan oleh peneliti. Sumber referensi ini dibutuhkan agar memperdalam pengetahuan peneliti terhadap penelitian yang akan dilakukan.
3. Peneliti mengumpulkan data-data *part number* baik dari mesin *hypot* dan mesin *vision* dari PT Jovan Technologies.
4. Peneliti menginput semua data yang didapatkan dari PT Jovan Technologies ke dalam *database* yang akan dikoneksikan dengan sistem yang akan dibuat.
5. Data *input* an yang akan dicari oleh *user* akan dicocokkan dengan data yang ada di dalam *database*,
6. Setelah dicocokkan dengan data yang ada didalam *database*, maka akan muncul *barcode* berisi data-data dari barang yang dicari oleh *user*.

3.2 Pengumpulan Data

Pengumpulan data adalah sebuah cara atau langkah yang dilakukan untuk mendapatkan sebuah informasi dari sumber tertentu. Dalam penelitian ini, peneliti menggunakan wawancara dan observasi dalam mengumpulkan data dari sumber data.

3.2.1 Teknik Wawancara

Peneliti melakukan wawancara kepada pihak kepala *supervisor* dalam PT Jovan Technologies yaitu bapak Harry Purba, untuk mendapatkan informasi. Wawancara dilakukan agar peneliti mendapatkan informasi yang tidak di dapatkan dari metode pengumpulan data yang lainnya.

3.2.2 Teknik Observasi

Pengumpulan data menggunakan observasi dilakukan peneliti dengan cara terjun langsung di lapangan yaitu peneliti secara langsung mengamati proses pencarian sampel verifikasi *part number* di PT Jovan Technologies. Dengan dilakukannya observasi, peneliti dapat menemukan kekurangan yang terjadi dalam proses pencarian sampel verifikasi *part number* yang tidak di dapatkan dari metode wawancara.

3.3 Operasional Variabel

Di dalam penelitian dalam pencarian sampel verifikasi *part number*, peneliti menggunakan *part number* mesin *Hypot* dan *auto Vision*.

3.3.1 Part Number Auto Vision

Mesin *vision* digunakan oleh PT Jovan Technologies sebagai pengecekan *reject* terhadap barang sebelumnya apakah sudah “OK” atau tidaknya barang tersebut. *Part number* mesin *vision* digunakan sebagai variabel *input* an untuk dijadikan *input item* dalam sistem penelitian ini.

3.3.2 *Part Number Mesin Hypot*

Mesin *hypot* digunakan oleh PT Jovan Technologies untuk pengecekan *short* atau tidaknya didalam satu barang tersebut. *Part number* mesin *hypot* dipergunakan dalam penelitian ini sebagai *item input* an dalam sistem yang akan dibuat.

3.4 Metode Perancangan Sistem

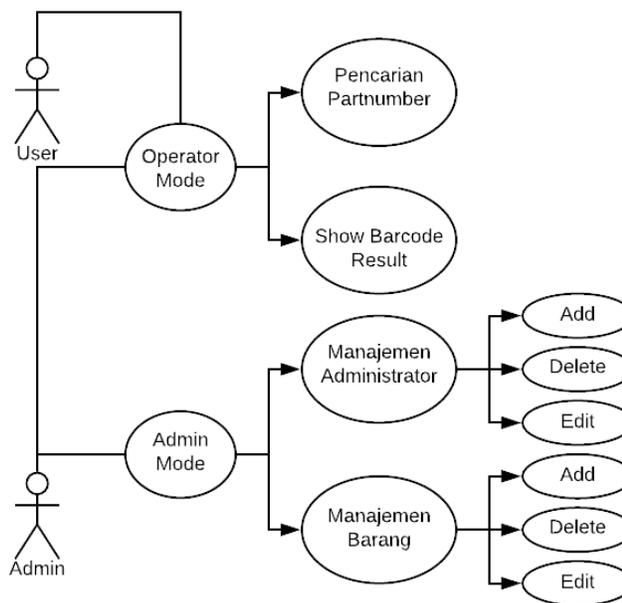
Perancangan sistem di sini berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi. Perancangan sistem menentukan bagaimana suatu sistem akan menyelesaikan apa yang mesti diselesaikan. Tahap ini termasuk mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem sehingga setelah dilakukan instalasi akan benar-benar memuaskan rancang bangun yang telah ditetapkan pada akhir tahap analisis sistem. Berikut ini merupakan tahapan-tahapan yang ada di dalam perancangan sistem.

3.4.1 Perancangan UML (*Unified Modeling Language*)

Pemodelan (*modelling*) adalah tahap merancang perangkat lunak sebelum melakukan tahap pembuatan program (*coding*). Pada penelitian ini, perancangan sistem dilakukan dengan memodelkan permasalahan dalam bentuk diagram-diagram UML.

A. Use Case Diagram

Use case diagram dibawah ini menggambarkan dua aktor yang sedang menggunakan sistem, yaitu aktor *admin* dan aktor karyawan sehingga pembuatan *use case* diagram ini lebih dititikberatkan pada fungsionalitas yang ada pada sistem, bukan berdasarkan alur atau urutan kejadian. Dalam aplikasi ini terdapat 4 interaksi, diantaranya “Pencarian *Part Number*”, “Menampilkan Hasil *Barcode*”, “Manajemen Administrator”, “Manajemen Barang”. Penjelasan *use case* diagram tentang aplikasi verifikasi *part number* akan dijelaskan pada gambar dibawah ini.



Gambar 3.2 *Use Case* Diagram Aplikasi
Sumber Data Olahan Peneliti 2019

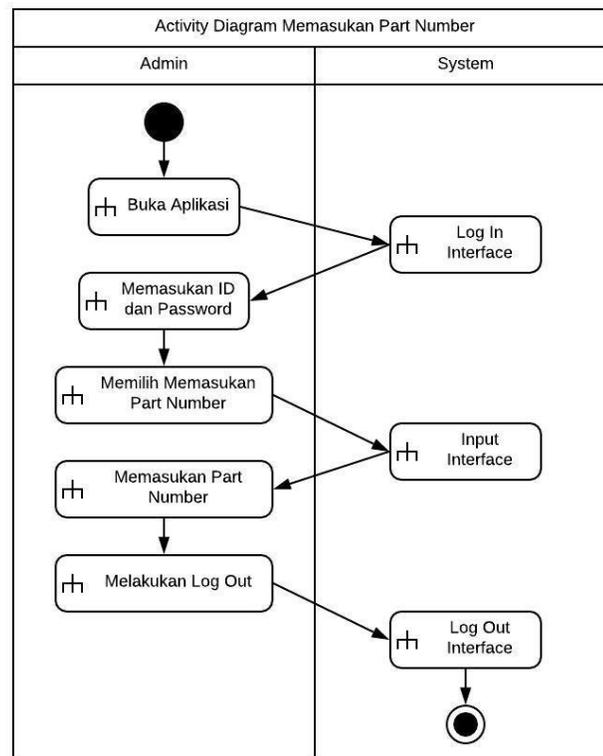
B. Activity Diagram

Activity diagram menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktivitas yang dibentuk dalam satu operasi sehingga dapat

juga untuk aktivitas lainnya. Berikut adalah penjelasan dari *activity* diagram pada aplikasi verifikasi ini.

1. *Activity* Diagram Memasukan *Part Number*

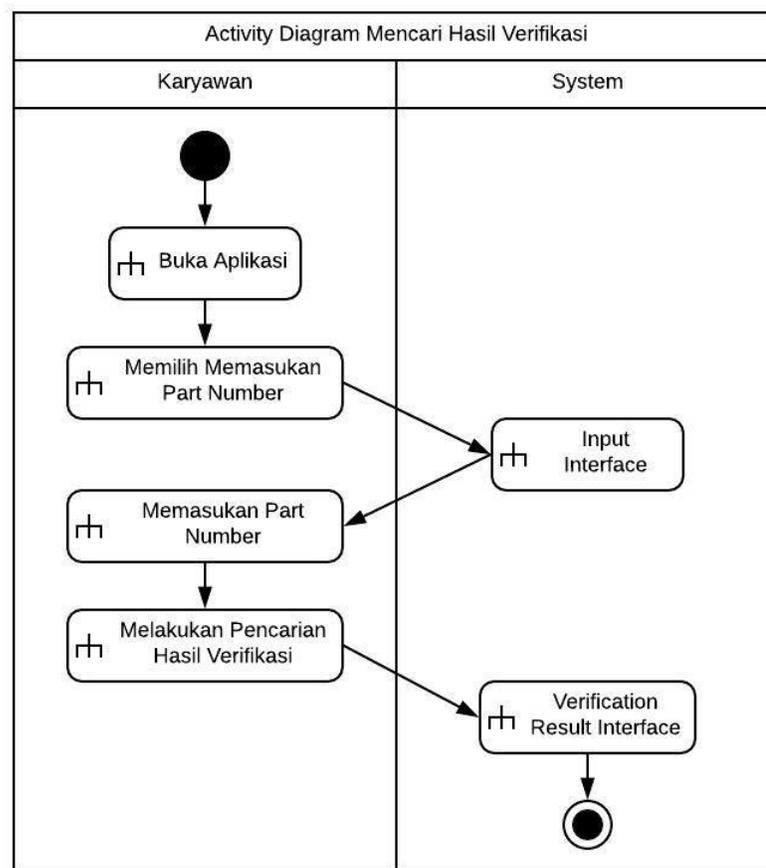
Activity diagram memasukan *part number* dimulai dengan *admin* sebagai pemegang akses penuh terhadap aplikasi membuka aplikasi verifikasi. *Admin* melakukan *login* ke dalam aplikasi untuk masuk ke dalam *mode admin* yang dapat memasukan data ke dalam *database*. *Admin* memasukan *part number* yang baru ke dalam *database*. Setelah itu *admin* melakukan *logout* dari aplikasi. Berikut adalah gambar dari *activity* diagram untuk memasukan *part number* oleh admin.



Gambar 3.3 Memasukan *Part Number*
Sumber Data Olahan Peneliti 2019

2. Activity Mencari Hasil Verifikasi *Part Number*

Activity diagram dalam mencari hasil verifikasi dimulai dengan karyawan sebagai *user* membuka aplikasi verifikasi. Karyawan memasukkan *part number* yang akan dicek verifikasinya oleh aplikasi. Setelah itu aplikasi akan menampilkan hasil verifikasi terhadap *part number* yang dimasukan oleh karyawan. *Activity* diagram dalam mencari verifikasi *part number* akan dijelaskan pada gambar dibawah ini.



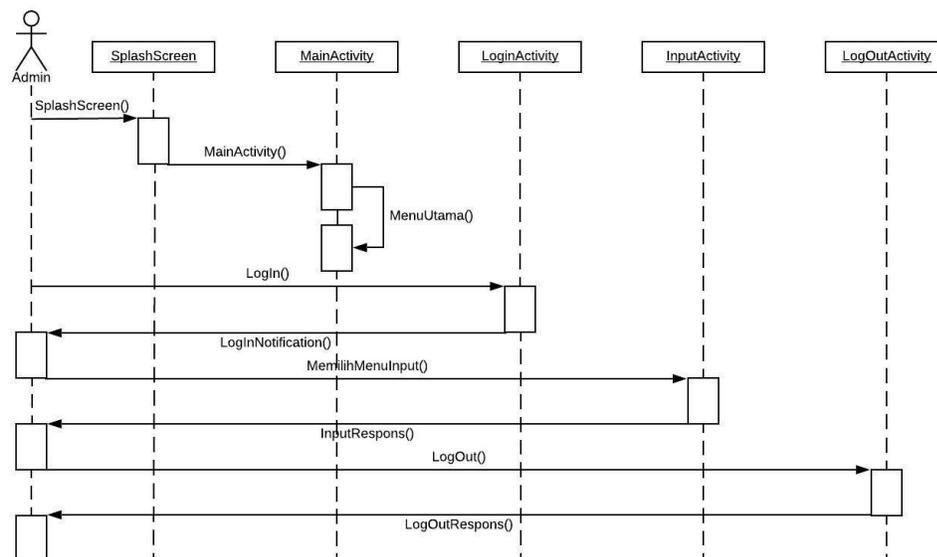
Gambar 3.4 Mencari Verifikasi *Part Number*
Sumber Data Olahan Peneliti 2019

C. Sequence Diagram

Menggambarkan interaksi antara sejumlah objek dalam urutan waktu. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antar objek yang terjadi pada titik tertentu dalam eksekusi sistem.. Berikut penjelasan dari diagram *sequence* dalam aplikasi verifikasi *part number*.

1. Memasukan *Part Number* Oleh Admin

Untuk dapat memilih menu *input part number*, admin harus memilih menu *login*, menu *input part number*, dan terakhir *log out*. Berikut adalah penjelasan gambar dari *input part number* pada aplikasi.

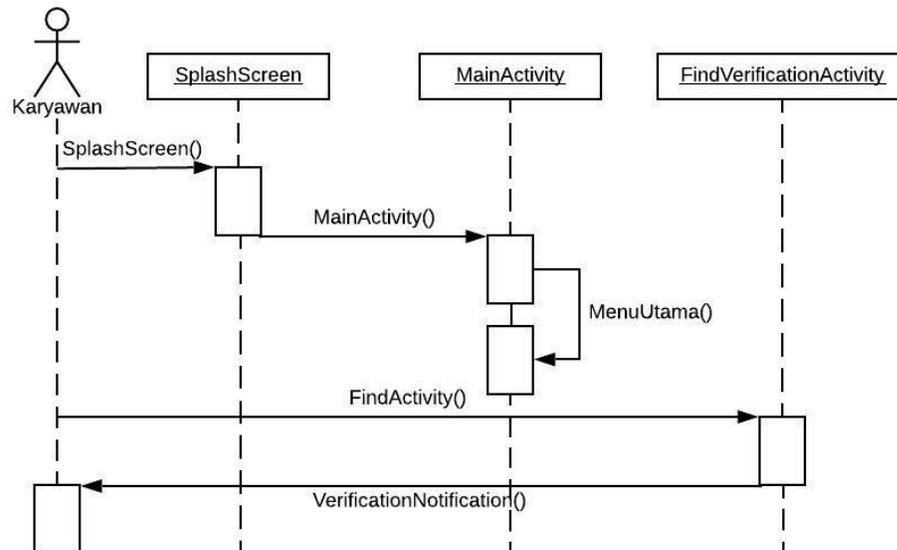


Gambar 3.5 *Input Part Number*
Sumber Data Olahan Peneliti 2019

2. Melakukan Pencarian Hasil Verifikasi

Untuk melakukan pencarian hasil verifikasi, karyawan hanya membuka aplikasi. Pada menu utama terdapat pilihan pencarian hasil verifikasi dengan cara

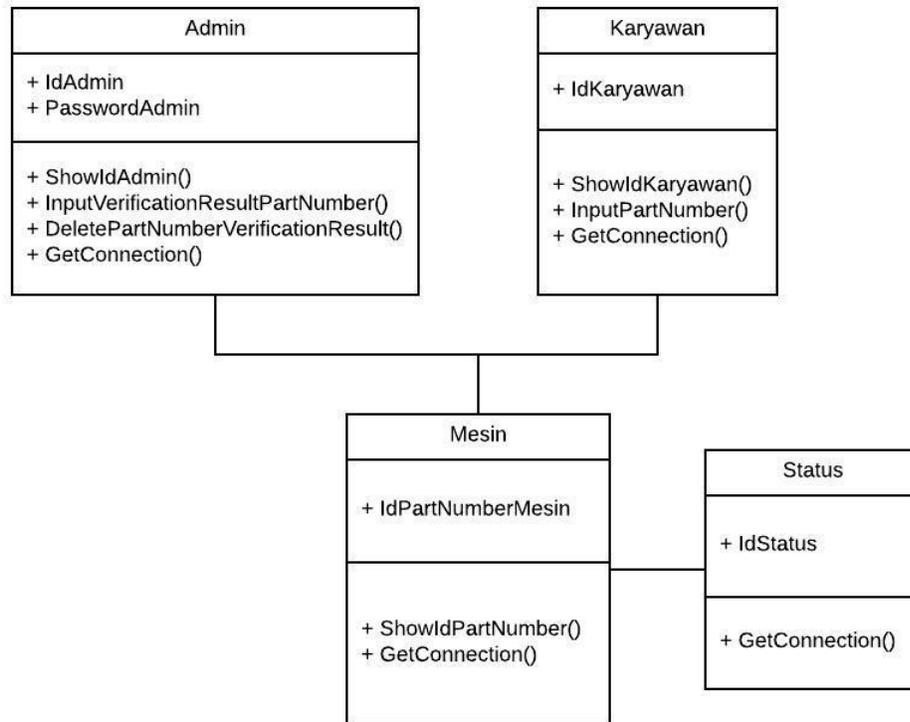
memasukan *part number* yang akan dicari. Setelah itu aplikasi akan memberikan hasil pencarian verifikasi yang akan dikirimkan pada layar aplikasi. Berikut merupakan penjelasan dari melakukan pencarian hasil verifikasi.



Gambar 3.6 Pencarian Hasil Verifikasi
Sumber Data Olahan Peneliti 2019

D. Class Diagram

Class diagram mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terjadi. Pada aplikasi verifikasi ini terdapat 4 kelas utama, yaitu kelas *admin*, kelas *karyawan*, kelas *mesin*, dan kelas *status*. *Class* diagram pada aplikasi ini akan dijelaskan pada gambar dibawah ini.



Gambar 3.7 Class Diagram Aplikasi
Sumber Data Olahan Peneliti 2019

3.5 Lokasi dan Jadwal Penelitian

Lokasi dan jadwal penelitian yang akan dilakukan oleh peneliti akan dijelaskan pada poin dibawah ini.

3.5.1 Lokasi Penelitian

Dalam melakukan penelitian untuk pembuatan sistem yang digunakan untuk mencari sampel verifikasi *part number*, peneliti mengambil lokasi penelitian di PT Jovan Technologies yang terletak di Union Industrial Park Blok B1 No 1-2, Batu Ampar, Kota Batam.

3.5.2 Jadwal Penelitian

Penelitian ini dilakukan dari bulan Maret 2019 sampai dengan bulan Juli 2019.

Jadwal penelitian akan disajikan di dalam tabel dibawah ini.

Tabel 3.1 Jadwal Penelitian

No	Kegiatan	MAR 2019				APR 2019				MEI 2019				JUN 2019				JUL 2019			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Input Judul	■																			
2	Pengumpulan Data		■	■																	
3	Pembuatan BAB 1				■	■															
4	Pembuatan BAB 2						■	■													
5	Pembuatan BAB 3								■	■	■	■									
6	Pembuatan BAB 4												■	■	■	■	■				
7	Pembuatan BAB 5																		■	■	
8	Pengumpulan Skripsi																				■

Sumber Data Olahan Peneliti 2019