

SKRIPSI

**RANCANG BANGUN SIMULATOR SCREENING STATION PT. VALE
INDONESIA TBK, TERINTEGRASI TEKNOLOGI VIRTUALISASI DAN
CLOUD COMPUTING**

Disusun dan diajukan oleh:

ISMAIL HASAN

D411 16 301



**DEPARTEMEN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS HASANUDDIN
MAKASSAR**

2021

HALAMAN JUDUL

SKRIPSI

**RANCANG BANGUN SIMULATOR SCREENING STATION PT. VALE
INDONESIA TBK, TERINTEGRASI TEKNOLOGI VIRTUALISASI DAN
CLOUD COMPUTING**

Disusun dan diajukan oleh:

ISMAIL HASAN

D411 16 301

DEPARTEMEN TEKNIK ELEKTRO

FAKULTAS TEKNIK

UNIVERSITAS HASANUDDIN

MAKASSAR

2021

LEMBAR PENGESAHAN SKRIPSI

**RANCANG BANGUN SIMULATOR SCREENING STATION PT. VALE
INDONESIA TBK, TERINTEGRASI TEKNOLOGI VIRTUALISASI DAN
CLOUD COMPUTING.**

Disusun dan diajukan oleh

ISMAIL HASAN
D411 16 301

Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka penyelesaian studi Program Sarjana Program Studi Teknik ELEktro Fakultas Teknik Universitas Hasanuddin Pada Tanggal 20 Januari 2021 dan dinyatakan telah memenuhi syarat kelulusan.

Menyetujui

Pembimbing Utama,

Pembimbing Pendamping,


Prof. Dr. Ir. Andani Achmad, MT
NIP. 19601231 198703 1 022


Ir. Samuel Panggalo., MT
NIP. 19620304 198811 1 001

Ketua Departemen Teknik Elektro
Fakultas Teknik
Universitas Hasanuddin


Dr. Eng. Ir. Dewiani, MT.
NIP. 19691026 199412 2 001

PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini;

Nama : Ismail Hasan
NIM : D41116301
Program Studi : Teknik Elektro
Jenjang : S1

Menyatakan dengan ini bahwa karya tulisan saya berjudul

Rancang Bangun Simulator Screening Station PT. Vale Indonesia TBK, Terintegrasi Teknologi Virtualisasi dan Cloud Computing

Adalah karya tulisan saya sendiri, bukan merupakan pengambilan alihan tulisan orang lain dan bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri.

Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan isi skripsi ini hasil karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Makassar, 6 Februari 2021

Yang Menyatakan,



Ismail Hasan

NIM. D411 16 301

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas segala berkat dan penyertaannya sehingga penulis berhasil menyelesaikan Tugas Akhir yang berjudul ” **Rancang Bangun Simulator Screening Station PT. Vale Indonesia TBK, Terintegrasi Teknologi Virtualisasi dan Cloud Computing.**”. Tugas Akhir ini dibuat sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata Satu (S1) di Universitas Hasanuddin, Departemen Teknik Elektro.

Bantuan dan dukungan datang dari berbagai pihak sehingga penulisan Tugas Akhir ini dapat berjalan dengan baik dan lancar. Oleh karena itu, penulis mengucapkan banyak terima kasih kepada:

1. Kedua orang tua yang senantiasa memberikan doa restu, motivasi, dan juga sebagai sumber dana utama selama penulis kuliah sampai penulisan Tugas Akhir.
2. Bapak **Prof. Dr. Ir. Andani Achmad, MT** selaku kepala Laboratorium Komputer dan Teknik Jaringan, juga selaku pembimbing Tugas Akhir.
3. Bapak **Ir. Samuel Panggalo., MT** selaku dosen pembimbing Tugas Akhir.
4. Bapak **Indin Hasan A.**, Bapak **Zulkifli**, Bapak **Agus Wahyuno**, dan Bapak **Busran Mahmud** selaku *Instrument* dan *IT Engineer* pada *Screening Station Centralized Control Room Project* PT. Vale Indonesia Tbk, sekaligus sebagai pembimbing lapangan.

5. Rekan-rekan mahasiswa Program Studi S1 Teknik Elektro Universitas Hasanuddin yang telah banyak memberikan masukan dan bantuan kepada penulis.
6. Seluruh dosen dan pegawai Departemen Teknik Elektro atas segala ilmu, bantuan, dan kemudahan yang diberikan selama kami menempuh proses perkuliahan.
7. Dan semua pihak yang telah membantu penulis menyelesaikan skripsi ini, yang tidak dapat penulis sebutkan satu-persatu.

Penulis menyadari bahwa masih banyak kekurangan yang terdapat di dalam Tugas Akhir ini. Oleh karena itu penulis mengharapkan kritik dan saran yang membangun.

Akhir kata, semoga Tugas Akhir ini dapat bermanfaat bagi para pembaca.

Gowa, 8 Januari 2021

Penulis,

Ismail Hasan
NIM. D411 16 301

DAFTAR ISI

KATA PENGANTAR	iv
DAFTAR ISI.....	vi
DAFTAR GAMBAR	viii
DAFTAR TABEL.....	ix
DAFTAR SINGKATAN DAN TERMINOLOGI.....	x
ABSTRAK	xi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	4
1.3 Tujuan dan Manfaat Penelitian	4
1.4 Batasan Masalah.....	5
1.5 Sistematika Penulisan.....	6
BAB II LANDASAN TEORI	7
2.1 Supervisory Control and Data Acquisition (SCADA).....	7
2.2.1 Komponen Sistem SCADA.....	7
2.2 Programmable <i>Logic</i> Controller.....	9
2.2.1 Prinsip kerja PLC	11
2.2.2 Perangkat Lunak.....	13
2.3 Ladder <i>Logic</i>	16
2.4 <i>Cloud Computing</i>	20
2.3.1 Virtualisasi	20
2.3.2 VMware vSphere <i>Cloud OS</i>	22
2.5 <i>Screening Station</i> PT. Vale Indonesia.....	24
2.4.1 Aktuator.....	26
2.4.2 Sensor Instrumentasi	28
BAB III METODE PENELITIAN.....	33
3.1 Studi Literatur	33
3.2 Lokasi dan Waktu Penelitian	33
3.3 Perancangan Simulator terintegrasi Infrastruktur <i>Cloud</i>	34
3.3.1 Rancangan Umum.....	34

3.3.2	Prosedur Perancangn Sistem.....	35
3.3.3	Pengambilan Data Lapangan.....	36
3.3.4	Perancangan Program Simulator.....	36
3.3.5	Perancangan HMI Simulator.....	40
3.3.6	Perancangan <i>Cloud</i> Infrastructure.....	46
3.4	Pengujian dan Analisa Sistem.....	49
BAB IV HASIL DAN PEMBAHASAN		50
4.1	Hasil Pengujian Simulator.....	50
4.4.1	Pengujian Mode Local	51
4.4.2	Pengujian Mode Remote	53
4.4.3	Pengujian Interlock	55
4.2	Hasil Pengujian Infrastruktur <i>Cloud</i>	56
4.5.1	Pengujian cloning pada virtual machine	56
BAB V KESIMPULAN DAN SARAN.....		59
DAFTAR PUSTAKA		61
LAMPIRAN.....		63

DAFTAR GAMBAR

Gambar 2.1 Proses kerja PLC. [7]	11
Gambar 2.2 Overview RSLogix 5000. [9].....	14
Gambar 2.3 tampilan RSLinx. [9].....	15
Gambar 2.4 Layout dan skematik relay sederhana. [10].....	17
Gambar 2.5 Pengontrol relay sederhana [10].....	18
Gambar 2.6 Sebuah PLC diilustrasikan dengan relay[10].....	19
Gambar 2.7 Perbandingan Tradisional komputer biasa dan Virtualisasi. [11]	22
Gambar 2.8 Topologi vSphere Datacenter. [12]	24
Gambar 2.9 <i>Screening station</i> #5 PTVI.....	25
Gambar 2.10 Konstruksi Motor induksi 3 Fasa. [13].....	27
Gambar 2.11 Konstruksi Stator Motor Induksi [13].....	27
Gambar 2.12 Rosemount 3051L Level Transmitter. [15].....	29
Gambar 2.13 Phoenix Contact MCR-SL-CUC-100-I. [17]	30
Gambar 2.14 Honeywell LCZ Speed Sensor. [19]	31
Gambar 2.15 Siemens Belt Scale Milltronics. [21].....	32
Gambar 3.1 Diagram rancangan umum simulator	34
Gambar 3.2 Flowchart Prosedur Perancangan Sistem.	35
Gambar 3.3 Langkah Perancangan Program Simulator.....	36
Gambar 3.4 Cuplikan Wiring Diagram Motor H8M07.	39
Gambar 3.5 Hasil Perancangan Ladder Logic.	39
Gambar 3.6 Hasil Perancangan Interkoneksi Program dengan Scada.	40
Gambar 3.7 Control Panel Apron Feeder Lube Pump.	41
Gambar 3.8 Hasil Perancangan HMI Simulator : Motor Simulator.....	43
Gambar 3.9 Hasil Perancangan HMI Simulator : Interlock Simulator.	45
Gambar 3.10 Tahap Perancangan Infrastructure <i>Cloud</i>	46
Gambar 3.11 Cuplikan daftar <i>virtual machine</i> pada VMware ESXi.	47
Gambar 4.1 Cuplikan HMI Motor Simulator pada H8M21.....	51
Gambar 4.2 Cuplikan Popup Motor pada Simulator.....	51
Gambar 4.3 Jenis Jenis Status pada Popup.	52
Gambar 4.4 Jenis Jenis Alarm pada Popup.....	52
Gambar 4.5 Set-up infrastruktur <i>cloud</i> dalam melakukan pengujian	56

DAFTAR TABEL

Tabel 3.1 Daftar <i>equipment</i> yang telah dirancang program simulasinya.....	37
Tabel 3.2 <i>Input</i> , Jenis dan Tag pada <i>Equipment</i>	42
Tabel 3.3 Jenis Sinyal, Range, dan Tag pada <i>Equipment</i> Interlock.....	44
Tabel 4.1 Hasil pengujian mode local: H8M21 Stacker Conveyor.	53
Tabel 4.2 Hasil pengujian mode remote manual: H8M21 Stacker Conveyor.	54
Tabel 4.3 Hasil pengujian Interlock: H8M21 Stacker Conveyor.....	55
Tabel 4.4 Hasil Pengujian <i>Cloning</i> VM: VM-HMI	57
Tabel 4.5 Hasil Pengujian <i>Cloning</i> VM: VM- <i>Logic</i>	57
Tabel 4.6 Hasil pengujian instalasi simulator secara manual	57

DAFTAR SINGKATAN DAN TERMINOLOGI

AC	:	<i>Alternating Current</i>
CPU	:	<i>Central Processing Unit</i>
DC	:	<i>Direct Current</i>
DCS	:	<i>Distributed Control System</i>
EWP	:	<i>Engineering Work Package</i>
FD	:	<i>Function description</i>
GUI	:	<i>Graphic User Interface</i>
HA	:	<i>High Availability</i>
HMI	:	<i>Human Machine Interface</i>
I/O	:	<i>Input and Output</i>
IT	:	<i>Information Technology</i>
LED	:	<i>Light Emitting Diode</i>
MMI	:	<i>Man Machine Interface</i>
NIC	:	<i>Network Interface Card</i>
OPC	:	<i>OLE for Process Control</i>
OS	:	<i>Operating Sistem</i>
P&ID	:	<i>Piping and Instrumentation Diagram</i>
PC	:	<i>Personal Computer</i>
PLC	:	<i>Programmable Logic Control</i>
PTVI	:	<i>PT. Vale Indonesia</i>
RAM	:	<i>Random Access Memory</i>
RTU	:	<i>Remote Terminal Unit</i>
SCADA	:	<i>Supervisory Control and Data Acquisition</i>
VLAN	:	<i>Virtual Local Area Network</i>
VM	:	<i>Virtual Machine</i>
VSD	:	<i>Variable Speed Drive</i>

ABSTRAK

Pembaharuan sistem kontrol pada fasilitas pemilahan ore PT. Vale Indonesia menjadi sistem SCADA memiliki banyak tantangan. Diantaranya diperlukannya pengetesan rancangan SCADA yang baru untuk memastikan rancangan telah sesuai dan aman sebelum dioperasikan, dan juga dibutuhkannya pelatihan pengoperasian sistem SCADA yang akan diimplementasikan kepada operator *screening station*. Semua hal ini harus dilakukan tanpa mengganggu sistem pengoperasian dari *screening station*, karena dapat mengganggu sistem produksi nikel dari PTVI. Hal ini mendorong penulis untuk berinovasi dalam menyelesaikan masalah ini, dengan merancang suatu simulator untuk menggantikan seluruh peralatan (*field devices*) yang ada pada *screening station*, sehingga *live testing* dan pelatihan operator secara langsung pada sistem SCADA yang baru dapat dilakukan tanpa mengganggu proses produksi dari *screening station*. Simulator yang telah dikembangkan terdiri dari dua komponen, yaitu program simulator, yang mensimulasikan *field devices* pada *screening station*, dan juga HMI simulator, yang mensimulasikan *interface* dari sistem SCADA. Simulator ini kemudian divirtualisasi pada sebuah infrastruktur *cloud* untuk memudahkan penggunaan dan *cloning* simulator pada pelatihan operator. Pengujian simulator dilakukan dengan metode BlackBox, dengan memberi inputan pada simulator dan membandingkan outputnya dengan output pada SCADA jika diberikan inputan yang serupa. Infrastruktur *cloud* juga diuji dengan membandingkan waktu penggandaan simulator tanpa dan dengan virtualisasi pada infrastruktur. Dari pengujian yang telah dilakukan, simulator yang dirancang telah bekerja seperti dengan SCADA dan *field devices* yang ada pada *screening station*, dan simulator yang telah divirtualisasi dapat di *clone* 21.71 lebih cepat dibandingkan dengan *cloning* tanpa virtualisasi. Dengan demikian dapat disimpulkan bahwa simulator yang dibuat telah mampu merepresentasikan *field devices* dan SCADA yang ada pada *screening station*, dan juga dilakukannya virtualisasi pada simulator mampu mempermudah dilakukannya pelatihan secara massal.

Kata Kunci: Simulator, SCADA, HMI, Infrastruktur Cloud, Sistem Kendali.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Tingginya persaingan dalam dunia industri saat ini menekan perusahaan untuk terus melakukan inovasi dan pembaharuan teknologi agar lebih efisien untuk mampu bertahan dan bersaing pada pasar global. Pembaharuan dan pengembangan pada sistem otomasi merupakan salah satu upaya yang dilakukan untuk menjalankan sebuah pabrik dengan lebih handal dan efisien. Hal inilah yang dilakukan oleh PTVI (PT. Vale Indonesia) pada salah satu fasilitas produksi yang berada di area tambang yaitu fasilitas pemilahan (*screening station plant*) dengan mengimplementasikan sistem kontrol SCADA (*Supervisory Control and Data Acquisition*).

Fasilitas pemilahan (*Screening station*) merupakan fasilitas produksi yang berada di area tambang dimana ore hasil tambang dipilah, disaring dan ditimbang untuk memisahkan material ore sesuai dengan spesifikasi yang dibutuhkan dalam proses produksi bijih nikel. PTVI mempunyai lima *screening station* dengan jarak yang berjauhan satu dengan yang lainnya. Setiap *screening station* tersebut memiliki *local control room* tersendiri dengan operator interface HMI (Human Machine Interface) untuk memonitor dan mengontrol *screening station* secara lokal. Untuk mengefisiensikan pengoperasian *screening station*, PTVI melakukan pembaharuan sistem kontrol dengan mengimplementasikan sistem SCADA yang mengintegrasikan sistem kontrol lokal dan monitoring kelima *screening station* tersebut menjadi satu ruang *control* terpusat. Dengan demikian, kelima *screening*

station dapat dikontrol dan monitor secara bersamaan dengan jumlah operator yang lebih sedikit dari sebelumnya, dan juga memudahkan koordinasi produksi antar *screening station*.

Namun, pembaharuan sistem ini memiliki banyak tantangan diantaranya:

1. Rancangan sistem SCADA yang baru harus diuji coba untuk memastikan rancang sudah sesuai dengan *function description* dan aman sebelum di operasikan.
2. Diperlukan training operasi SCADA baru pada seluruh operator *screening station* agar mempunyai kompetensi yang standar.

Untuk memenuhi kebutuhan di atas maka rancangan SCADA yang baru harus di interkoneksi pada PLC *screening station* dan konsekuensinya adalah menghentikan operasi untuk keperluan *testing* dan *training*. Terhentinya produksi *screening station* akan menghambat produksi ore yang dapat mengakibatkan penurunan produksi nikel pada PTVI. Hal ini menyebabkan melakukan *live testing* dan pelatihan operator secara langsung pada sistem baru tidak dapat dilakukan.

Hal ini mendorong penulis untuk berinovasi dalam menyelesaikan masalah ini, dengan merancang suatu program simulasi untuk menggantikan seluruh peralatan (*field devices*) yang ada pada *screening station*, sehingga *live testing* dan pelatihan operator secara langsung pada sistem SCADA yang baru dapat dilakukan tanpa mengganggu proses produksi dari *screening station*.

Program simulasi sistem kontrol *screening station* ini akan divirtualisasi dan dijalankan pada sebuah infrastruktur *cloud*, sehingga dapat dengan mudah diakses dan digunakan sebagai sarana pelatihan operator *screening station*.

Virtualisasi adalah proses pembuatan perangkat keras virtual dengan menggunakan perangkat lunak (hypervisor) yang digunakan diantara operating system dan perangkat keras yang sebenarnya. [1] Dengan menggunakan teknologi ini, simulator dapat diubah menjadi sebuah mesin virtual (*virtual machine*), diduplikasi, dan dijalankan secara bersamaan pada satu komputer yang sama, sehingga mempermudah melakukan kegiatan pelatihan yang memiliki banyak peserta.

Cloud computing adalah penggunaan kumpulan sumber daya komputasi melalui sebuah jaringan. Sumber daya ini dapat diatur secara dinamis sesuai dengan kebutuhan pengguna, sehingga sumber daya komputasi tersebut dapat digunakan secara lebih efisien. [2] Dengan menggunakan teknologi ini, simulator yang telah dibangun dapat diakses melalui jaringan komunikasi, sehingga dapat diakses dari jarak yang jauh. Semua komputasi dari simulator juga berjalan pada infrastruktur *cloud*, sehingga simulator dapat dijalankan dengan baik walaupun diakses dengan menggunakan komputer dengan performa yang rendah.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka penulis merumuskan permasalahan tugas akhir ini sebagai berikut:

1. Bagaimana merancang simulator *screening station* yang terinterkoneksi dengan sistem SCADA yang baru, sehingga dapat digunakan sebagai media pengujian dan pelatihan operator *screening station*?
2. Bagaimana cara memvirtualisasi dan menjalankan simulator pada infrastruktur *cloud* agar simulator dapat lebih efektif digunakan untuk media pelatihan operator?
3. Bagaimana memastikan simulator yang dibuat telah bekerja dengan benar dan telah merepresentasikan *device - device* yang ada pada *screening station*?

1.3 Tujuan dan Manfaat Penelitian

Tujuan dari penelitian ini adalah:

1. Merancang simulator *screening station* yang terinterkoneksi dengan sistem SCADA yang baru, sehingga dapat digunakan sebagai media pengujian dan pelatihan operator *screening station*.
2. Melakukan virtualisasi pada simulator dan menjalankannya pada infrastruktur *cloud* agar simulator dapat lebih efektif digunakan untuk media pelatihan operator.

3. Melakukan pengujian pada simulator yang telah dibuat, untuk memastikan simulator yang dibuat telah bekerja dengan benar dan telah merepresentasikan *device - device* yang ada pada *screening station*.

Adapun manfaat dari penelitian ini bagi PT. Vale Indonesia adalah:

1. Implementasi simulator *screening station* pada proses testing sistem SCADA tidak akan mengganggu proses produksi ore pada PT Vale Indonesia.
2. Proses pelatihan operator dalam mengoperasikan sistem SCADA *screening station* yang baru melalui simulator dapat dilakukan secara *live* dan interaktif kapanpun dan dimanapun tanpa mengganggu proses produksi pada PT Vale Indonesia.

1.4 Batasan Masalah

Agar Tugas Akhir ini mengarah pada tujuan dan menghindari melebar nya masalah yang kemungkinan muncul, maka perlu dilakukan pembatasan masalah agar sesuai dengan judul Tugas Akhir ini. Adapun batasan masalah sebagai berikut:

1. Simulator yang dirancang hanya pada fasilitas *screening station* #5 yang menggunakan PLC Allen-Bradley *Control Logix* sehingga hanya menggunakan perangkat lunak yang diperuntukkan untuk sistem tersebut.
2. Program *Logic dan HMI* dirancang mengikuti standar perancangan PT. Vale Indonesia.
3. Tugas Akhir ini tidak memperhitungkan analisis nilai ekonomi.

1.5 Sistematika Penulisan

Tujuan dari sistematika penulisan adalah untuk memudahkan pemahaman laporan tugas akhir ini secara keseluruhan. Garis besar pembahasan tiap-tiap bab diuraikan sebagai berikut:

BAB I PENDAHULUAN

Bab ini menguraikan tentang Latar belakang, Tujuan dan Manfaat, Perumusan masalah, Batasan masalah, dan Sistematika Penulisan.

BAB II LANDASAN TEORI

Pada bab ini akan dijelaskan tentang teori penunjang yang relevan untuk bahan penelitian yang diperoleh dari sumber referensi untuk menyusun kerangka teori dan konseptual.

BAB III METODE PENELITIAN

Memuat langkah-langkah yang dilakukan pada penelitian, diantaranya, alat dan bahan, komponen dan perangkat penelitian, prosedur kerja, dan perancangan.

BAB IV HASIL DAN PEMBAHASAN

Bagian ini berisi hasil dan pengujian dari perancangan sistem dan pembahasannya.

BAB V KESIMPULAN

Bab ini akan menyimpulkan semua kegiatan dan hasil-hasil yang diperoleh selama proses pembuatan dan pengujian sistem serta saran-saran yang sekiranya diperlukan untuk menyempurnakan penelitian berikutnya.

BAB II

LANDASAN TEORI

2.1 Supervisory Control and Data Acquisition (SCADA)

SCADA adalah singkatan dari *supervisory control and data acquisition* (pengawasan kontrol dan akuisisi data) yang merupakan salah satu arsitektur sistem kontrol yang menggunakan jaringan komunikasi data dan antarmuka grafis (GUI) untuk melakukan pengawasan proses *high-level*, dan menggunakan peralatan peripheral seperti PLC (*programmable logic controller*) untuk berinteraksi dengan mesin mesin pabrik. Sistem ini memungkinkan untuk melakukan pengawasan dan kontrol pada suatu plant atau pabrik yang kompleks dengan mudah melalui antarmuka grafis, dengan menghubungkan HMI (Human Machine Interface) pada komputer operator dengan sensor dan aktuator yang terhubung dengan PLC dan RTU pada plant.[3]

2.2.1 Komponen Sistem SCADA

Pada umumnya, sistem SCADA terdiri dari komponen utama berikut:

1. Supervisory computers

Pada sistem SCADA, komponen ini bertanggung jawab untuk melakukan pengambilan dan pemrosesan data, juga melakukan kontrol terhadap peralatan yang berada di lapangan. Komputer ini berisikan *software software* yang dapat berkomunikasi dengan *controller* di lapangan seperti PLC dan RTU. [3]

2. Remote terminal units (RTU)

Remote terminal unit merupakan perangkat kontrol dan komunikasi yang terhubung langsung dengan sensor dan aktuator. Perangkat berfungsi untuk mengkomunikasikan *supervisory computer* dengan sensor dan aktuator melalui infrastruktur komunikasi sistem SCADA.[3]

3. Programmable logic controllers (PLC)

Programmable logic controller atau yang lebih dikenal dengan PLC merupakan komponen yang terhubung dengan sensor dan aktuator untuk melakukan pengontrolan pada sebuah plant. Komponen ini juga berfungsi untuk menghubungkan *field devices* dengan *supervisory computer* melalui infrastruktur komunikasi sistem SCADA, sama seperti RTU. Hanya saja PLC lebih ekonomis, fleksibel, dan dapat dikonfigurasi, sehingga lebih sering digunakan dalam dunia industri. [3]

4. Communication infrastructure

Komponen ini berfungsi untuk menghubungkan RTU dan PLC dengan *supervisory computer* agar dapat berkomunikasi seperti menerima data dari sensor, dan juga memberi perintah pada aktuator. Komunikasi yang digunakan menggunakan protokol tertentu, sehingga RTU, PLC, dan *supervisory computer* dapat saling berkomunikasi. Pada umumnya sistem ini memiliki *redundancy* yang baik, sehingga disaat ada link yang rusak, sistem SCADA tidak tumbang begitu saja. [3]

5. Human-Machine interface (HMI)

Human machine interface atau yang lebih dikenal dengan istilah HMI merupakan antarmuka grafis yang digunakan operator untuk berinteraksi dengan sistem SCADA. HMI berisikan representasi *field devices* atau peralatan di lapangan beserta keadaan keadaannya, opsi opsi perintah kontrol, laporan laporan keadaan plant, trending data, peringatan peringatan, notifikasi, dll. Pada umumnya, tampilan HMI dibuat menyerupai plant itu sendiri, sehingga dapat dengan mudah dibaca oleh operator. [3]

2.2 Programmable Logic Controller

Control engineering telah berkembang seiring berjalannya waktu. Dulu, manusia adalah metode pengontrolan utama dalam pengendalian sistem. Saat ini, listrik telah digunakan untuk kontrol dan kontrol listrik di masa awal didasarkan pada sistem relay. Relay memungkinkan *power* untuk diaktifkan dan dinonaktifkan tanpa saklar mekanik. Sudah umum untuk menggunakan relay untuk membuat keputusan kontrol logis yang sederhana. Pengembangan *low cost computer* telah membawa revolusi baru, yakni *Programmable Logic Control (PLC)*. Munculnya PLC dimulai pada 1970, dan telah menjadi pilihan yang paling umum untuk kontrol industri. PLC telah mendapatkan popularitas di dunia industri dan mungkin akan tetap berjaya untuk beberapa waktu ke depan. Sebagian besar ini adalah karena keuntungan penggunaan PLC. [4]

PLC merupakan suatu bentuk khusus pengontrol berbasis *microprocessor* yang memanfaatkan memori yang dapat diprogram untuk menyimpan instruksi – instruksi dan untuk mengimplementasikan fungsi - fungsi semisal logika, *sequencing*, pewaktuan (*timing*), pencacahan (*counting*) dan aritmatika guna mengontrol mesin-mesin dan proses-proses serta dirancang untuk dioperasikan oleh para engineer yang hanya memiliki sedikit pengetahuan mengenai komputer dan bahasa pemrograman.[5]

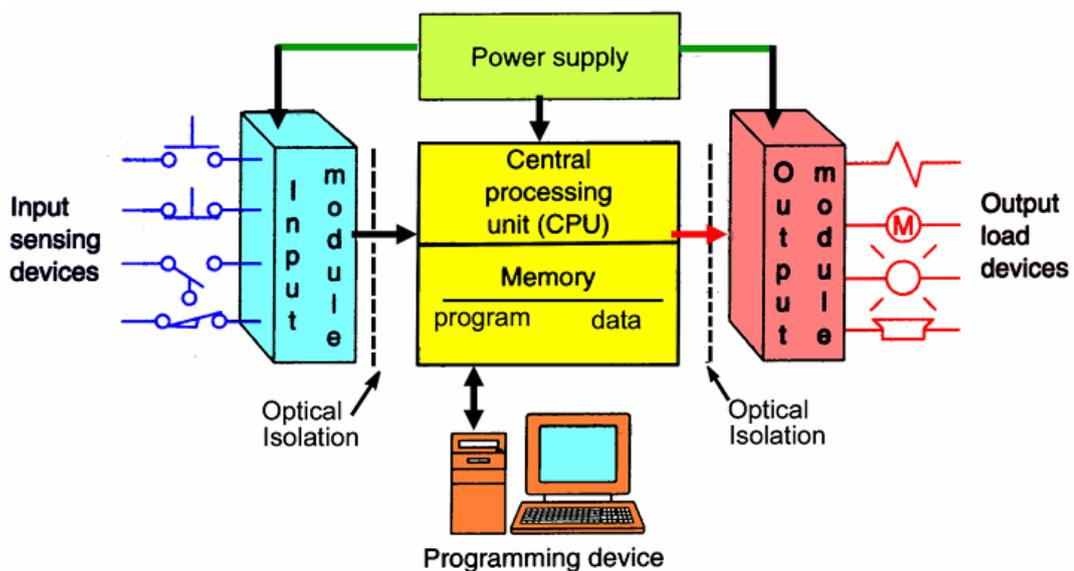
Sebagian besar industri telah menerapkan sistem otomatisasi dalam proses produksi. Pada umumnya sistem otomatis yang diterapkan terdiri atas dua metode yaitu otomatisasi berbasis kontrol relay dan otomatisasi berbasis *Programmable Logic Control (PLC)*.

Otomatisasi berbasis relay banyak digunakan pada mesin-mesin yang memiliki urutan-urutan (sekuens) yang sederhana, sedangkan otomatisasi PLC dapat memiliki sekuens yang lebih kompleks dari *relay*. Otomatisasi berbasis PLC dapat diintegrasikan dengan sistem monitoring. Sistem monitoring berbasis PLC adalah suatu sistem yang berguna untuk mengontrol proses suatu kerja tertentu., dimana parameter atau *inputan* data diambil dan diolah oleh Personal Computer (PC) dan melalui sebuah program tertentu.[5]

Sesuai namanya, PLC dapat dengan mudah diprogram ulang. Keunggulan PLC dibandingkan dengan sistem konvensional antara lain :

- a. Relatif mudah untuk melakukan perubahan pada strategi kendali yang akan diterapkan, karena logika kendali yang digunakan diwujudkan dalam bentuk perangkat lunak.
- b. Jumlah *relay* yang diperlukan dapat dikurangi sesuai dengan jumlah *input* maupun *output* yang diperlukan. Lebih mudah untuk proses instalasinya karena pengkabelan lebih sederhana.
- c. Lebih mudah dalam menemukan kesalahan dan kerusakan, karena memiliki fasilitas *self – diagnosis*.
- d. Tahan terhadap temperatur tinggi, tekanan tinggi dan kelembaban yang tinggi apabila dipakai secara terus - menerus, dan banyak dijumpai pada lingkungan industri. [6]

2.2.1 Prinsip kerja PLC



Gambar 2.1 Proses kerja PLC. [7]

Prinsip kerja PLC sistem diperlihatkan pada blok diagram gambar 2.1 dimana terdapat beberapa modul yang meliputi:

- ***Input***

Input yang akan masuk ke dalam CPU berupa sinyal dari sensor atau transduser. Sinyal sensor ini terdapat dua jenis, yaitu: sinyal diskrit dan sinyal analog. Sinyal diskrit berupa saklar biner dimana hanya sebuah ON atau OFF signal (1 atau 0, Benar atau salah), Contohnya: push button, limit switch dan level sensor. Sedangkan sinyal analog menggunakan prinsip rentang suatu nilai antara nol hingga skala penuh. Contohnya dalam kehidupan sehari-hari adalah ketika sedang memutar volume speaker atau radio. Rentang nilai dari sensor ini akan diinterpretasikan sebagai nilai-nilai integer oleh CPU PLC. CPU PLC pada saat ini sering menggunakan prosesor 16 bit sehingga nilai integer nya memiliki rentang “-32768 hingga 32767”. Contoh dari sinyal analog ini adalah sinyal yang dihasilkan dari sensor tekanan, sensor temperatur, level, dan sensor aliran. Sinyal analog dapat berupa tegangan atau arus listrik dan nilai ini akan diproporsionalkan dengan nilai integer CPU, contohnya: sebuah sinyal analog 0-5 V atau 4–20 mA akan dikonversi menjadi nilai integer 0 – 32767.[8]

- **CPU**

Semua aktivitas atau pemrosesan data yang diambil dari sensor (*data input*) terjadi pada Central Processing Unit (CPU). CPU ini memiliki tiga bagian utama, yaitu: Prosesor, sistem memori dan Sistem Power Supply. Prosesor akan memproses sinyal *input* secara aritmatika dan logika, yaitu: melakukan operasi

logika, sequential, timer, counter dan mengolah fungsi-fungsi yang diinginkan berdasarkan program yang telah ditentukan. Selain itu, processor juga mengolah program yang ada di dalam memori, serta mengatur komunikasi antara *input-output*, memori dengan processor itu sendiri. [8]

- ***Output***

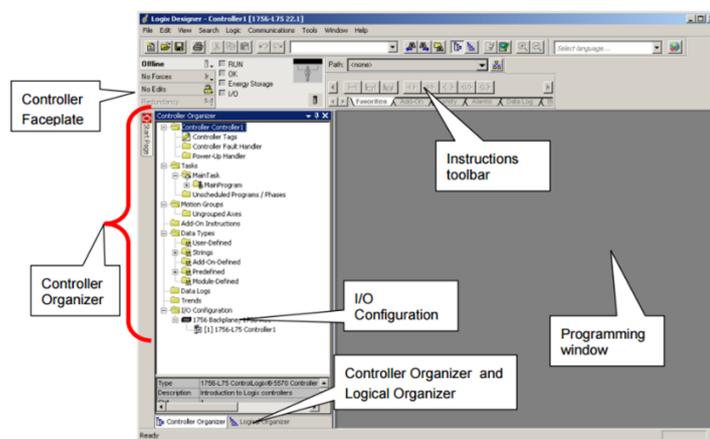
Hasil pemrosesan data yang diolah pada CPU akan berupa signal keluaran yang dikirim ke modul *output* untuk menjalankan aktuator. Untuk *output* digital, aktuator dapat berupa motor listrik, solenoid, heater, led display, injector, pompa dan lain-lain. Sedangkan *output* analog dapat berupa *control valve*, dosing pump, motor vsd, dll. Actuator ini akan berfungsi sesuai instruksi dari CPU, jika pada CPU telah di-program timer ON dari lampu selama dua detik maka lampu pada aktuator akan menyala selama dua detik dan kemudian setelah dua detik lampu akan OFF. [8]

2.2.2 Perangkat Lunak

Dalam melakukan pemrograman, simulasi, dan perancangan HMI pada *Screening Station #5*, digunakan proprietary *software* atau perangkat lunak yang dikhususkan untuk sistem kontrol PLC Allen-Bradley agar dapat bekerja dengan baik. Semua perangkat lunak ini di develop oleh Rockwell automation, yang merupakan perusahaan pemilik lisensi dari Allen-Bradley *control system*. Perangkat lunak yang digunakan antara lain; RSLogix 5000 yang digunakan untuk memprogram PLC, RSLinx yang digunakan untuk menghubungkan PLC, HMI, dan programming *devices*, dan FactoryTalk View untuk mendesain HMI dan membuat HMI server.[9]

RSLogix 5000

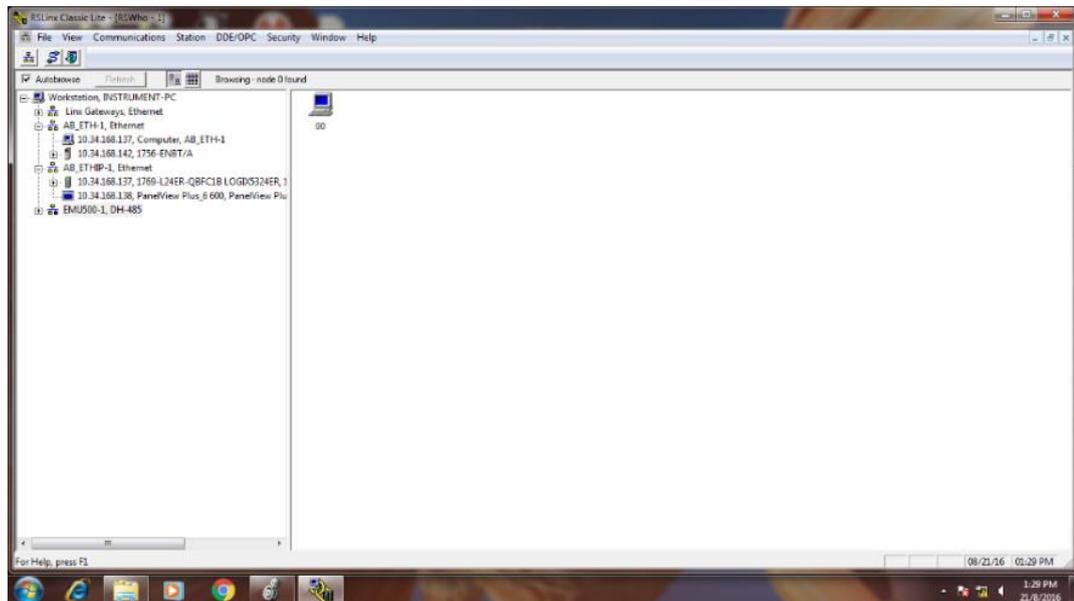
RSLogix 5000 diperlihatkan pada gambar 2.2 adalah aplikasi untuk mengkonfigurasi, memprogram, dan *maintain* semua produk dan *device* yang memiliki relasi dengan seri *controller Allen-Bradley® Logix 5000 family*. Dengan aplikasi *Logix Designer*, hanya dibutuhkan satu *software programming* untuk memprogram *sequential, process, drive, dan motion control*. [9]



Gambar 2.2 Overview RSLogix 5000. [9]

RSLinx Classic

RSLinx Classic adalah *software* yang menghubungkan antara hardware *control logix device* dengan aplikasi Microsoft Windows seperti RS Logix 5000 dan HMI *software* RS View SE. RS Link berfungsi mengkonfigurasi driver apa yang diperlukan user untuk berkomunikasi dengan Logix *controller*. *Software* ini digunakan untuk komunikasi antara komputer programmer dan *controller*. Gambar 2.3 di bawah ini memperlihatkan layar RS Link berkomunikasi dengan beberapa Control Logix *device*. [9]



Gambar 2.3 tampilan RSLinx. [9]

FactoryTalk View

FactoryTalk View adalah *software* untuk mengembangkan dan menjalankan operator interface atau Human-machine interface (HMI). FactoryTalk View dirancang untuk monitoring dan mengontrol proses-proses otomatisasi pada mesin-mesin industri. [9]

FactoryTalk View meliputi produk-produk berikut:

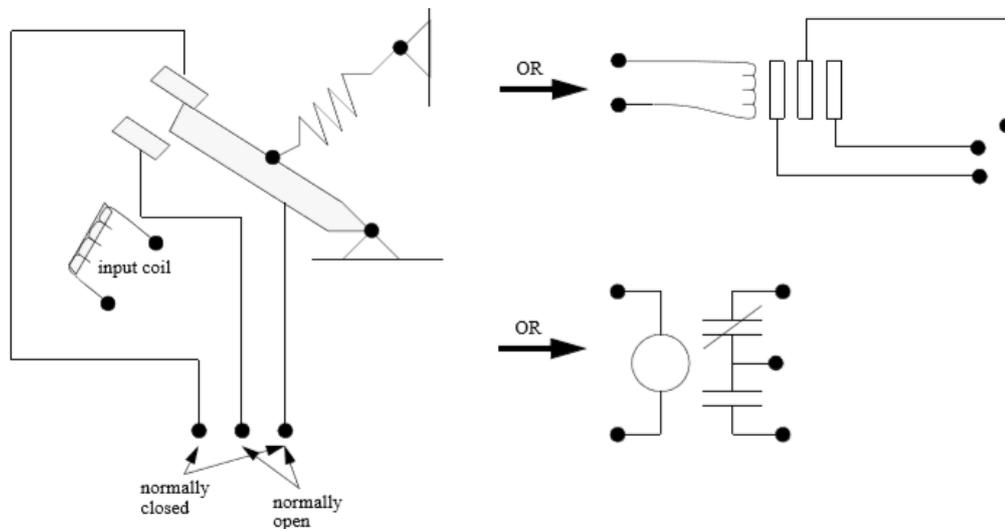
- FactoryTalk View Studio adalah *software* konfigurasi untuk developing machine-level applications. *Software* ini berjalan pada operating system seperti Microsoft Windows 7, Windows 8, Windows 8.1, Windows Server 2008, dan Windows Server 2012 operating systems. [9]

- FactoryTalk View SE Station adalah sebuah PC based standalone atau distributed HMI runtime environment applications. Dalam TA ini penulis menggunakan *software* HMI type ini. [9]

2.3 Ladder Logic

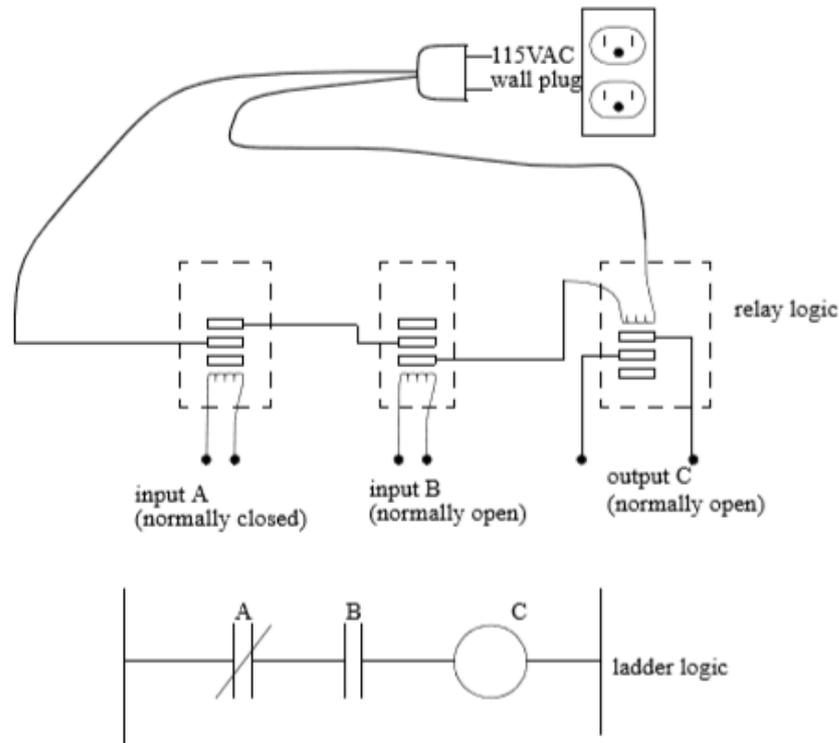
Ladder logic adalah metode pemrograman yang paling umum digunakan untuk PLC. *Ladder logic* sengaja dikembangkan untuk meniru logika relay. Keputusan yang strategis untuk menggunakan *relay logic diagrams*. Dengan memilih *Ladder logic* sebagai metode pemrograman utama, pelatihan ulang yang diperlukan untuk *engineers* dan *tradespeople* menjadi berkurang. Dalam sistem kontrol modern masih terdapat adanya relay, tapi jarang digunakan untuk *logic*.

Relay adalah perangkat sederhana yang menggunakan medan magnet untuk mengendalikan *switch*, seperti yang ditunjukkan pada Gambar 2.3. Ketika *input coil* diberi tegangan, arus yang dihasilkan menciptakan medan magnet. Medan magnet menarik saklar metal (atau *reed*) ke arahnya dan terjadi kontak, menutup saklar. Kontak yang menutup saat kumparan (*coil*) di-*energize* disebut *normally open*. Sedangkan kontak *normally closed* terkontak ketika *input coil* tidak di-*energize*. Relay biasanya digambarkan secara skematik dengan lingkaran sebagai simbol *input coil*. Kontak *output* dilambangkan dengan dua garis sejajar. Kontak *Normally open* dilambangkann sebagai dua garis, dan dalam keadaan *open (non-conducting)* ketika *input* tidak di-*energize*. Kontak *Normally closed* dilambangkan dengan dua garis dengan garis diagonal melaluinya. Ketika *input coil* tidak di-*energize* kontak *Normally closed* berada dalam keadaan *closed (conducting)*.



Gambar 2.4 Layout dan skematik relay sederhana. [10]

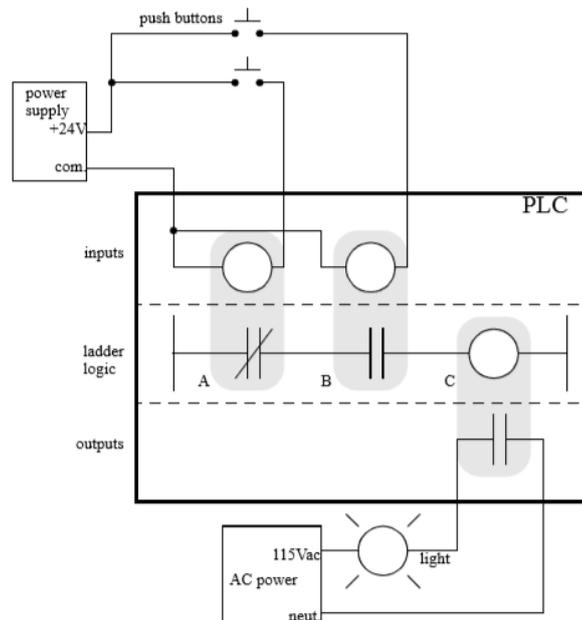
Relay digunakan untuk membiarkan suatu sumber *power* menutup saklar untuk suatu sumber *power* yang lain (biasanya yang memiliki arus besar), dalam keadaan kedua sumber terisolasi. Contoh aplikasi relay dalam kontrol sederhana ditunjukkan pada Gambar 2.4. Dalam sistem ini, relay pertama di sebelah kiri digunakan sebagai *normally closed*, dan akan melewatkan aliran arus saat *input A* diberi tegangan. Relay kedua adalah *normally open* dan tidak akan melewatkan arus sampai *input B* diberi tegangan. Jika arus berhasil melalui kedua relay pertama maka arus akan mengalir ke *coil* relay ketiga, dan menutup saklar untuk rangkaian *output C*. Rangkaian ini yang biasanya digambarkan dalam bentuk *Ladder logic*. Hal ini dapat dibaca secara logis sebagai C akan *on* jika A *off* dan B *on*. [10]



Gambar 2.5 Pengontrol relay sederhana [10]

Contoh pada Gambar 2.3 tidak menunjukkan keseluruhan dari sistem kontrol, tetapi hanya logika berpikirnya. Sistem PLC terdiri dari *input*, *output*, dan *logic* itu sendiri. Gambar 2.4 menunjukkan representasi yang lebih lengkap dari PLC. Di sini, terdapat dua *input* dari *push button*. *Input* ini sebagai aktifator kumparan relay 24V DC di PLC. *Input* ini yang nantinya akan membuat *output relay* mengaktifkan switch 115V AC, dan menyalakan lampu. Catatan, di PLC yang sebenarnya *input* tidak pernah berupa relay, tetapi *output* biasanya relay. *Ladder logic* pada PLC sebenarnya adalah program komputer yang pengguna dapat *input* dan ubah. Perhatikan bahwa kedua *input push* ini adalah *normally open*, tapi *Ladder logic* dalam PLC memiliki satu kontak *normally open* dan satu kontak *normally closed*. Jangan berpikir bahwa *Ladder logic* dalam PLC harus sesuai dengan *input* atau pun

output. Banyak pemula mencoba untuk membuat *Ladder logic* sesuai dengan jenis *input*. [10]



Gambar 2.6 Sebuah PLC diilustrasikan dengan relay[10]

Banyak relay juga memiliki *output* lebih dar satu (*throws*) dan ini memungkinkan *output* relay di saat yang sama menjadi *input*. Rangkaian yang ditunjukkan pada Gambar 2.5 adalah contoh dari ini, ini disebut *seal* atau *latch* pada sirkuit. Dalam rangkaian ini, arus dapat mengalir melalui salah satu cabang dari sirkuit, melalui kontak berlabel A atau B. *Input* B hanya akan *on* ketika *output* B *on*. Jika *input* B *off*, dan A *energized*, maka *output* B akan aktif. Jika *output* B aktif maka *input* B akan *on*, dan mempertahankan *output* B aktif bahkan jika *input* A telah *off*. Setelah *input* B aktif *output* B tidak akan *off*. [10]

2.4 Cloud Computing

Cloud computing, atau dalam Bahasa Indonesia “komputasi awan”, adalah penggunaan kumpulan sumber daya komputasi (seperti CPU, RAM, dan Storage) melalui sebuah jaringan. Sumber daya ini dapat diatur secara dinamis sesuai dengan kebutuhan pengguna, sehingga sumber daya komputasi tersebut dapat digunakan secara lebih efisien. [2]

Hal ini memungkinkan pengguna untuk melakukan komputasi dengan sumber daya yang besar menggunakan komputer dengan sumber daya yang kecil, bahkan smartphone sekalipun. Hal ini dikarenakan semua komputasi dilakukan pada *cloud server*, dan komputer pengguna hanya digunakan untuk berinteraksi dengan *cloud server*.

Pada umumnya, dalam *cloud computing* digunakan teknologi virtualisasi, dimana resource dari *cloud server* dapat dialokasikan secara fleksibel pada *virtual machine*. Dengan begitu, sebuah *cloud server* dapat menjalankan banyak *virtual machine*, atau “mesin virtual”, dan membagi resourcenya pada *virtual machines* tersebut sesuai kebutuhan. Hal ini sangat bermanfaat dikalangan perusahaan perusahaan besar, dimana kebutuhan server selalu bertambah seiring berkembangnya perusahaan tersebut. dengan teknologi ini, dapat dengan mudah melakukan “*scale up*” dan penambahan server pada sebuah *virtual machine*. [2]

2.3.1 Virtualisasi

Virtualisasi adalah teknologi perangkat lunak yang memisahkan perangkat keras fisik komputer dari sistem operasi (OS) dan aplikasi perangkat lunaknya,