



UNIVERSITAS DIPONEGORO

**RANCANG BANGUN CHATBOT UNTUK
AKUISISI DATA PADA GREENHOUSE DENGAN
MENGGUNAKAN TELEGRAM DAN GOOGLE DRIVE API DI
PT PLN NUSANTARA POWER UP REMBANG**

LAPORAN KERJA PRAKTIK

NAUFAL YAFI SUSANTO

21060120120011

**FAKULTAS TEKNIK
DEPARTEMEN TEKNIK ELEKTRO
PROGRAM STUDI SARJANA**

**SEMARANG
FEBRUARI 2023**

LEMBAR PENGESAHAN DEPARTEMEN

RANCANG BANGUN CHATBOT UNTUK

AKUISISI DATA PADA GREENHOUSE DENGAN

MENGGUNAKAN TELEGRAM DAN GOOGLE DRIVE API DI

PT PLN NUSANTARA POWER UP REMBANG

Disusun oleh:

Naufal Yafi Susanto

21060120120011

PT PLN Nusantara Power UP Rembang

09 Januari 2023 s/d 17 Februari 2023

Telah diperiksa pada tanggal:

.....

Mengetahui

Ketua Departemen Teknik Elektro
Universitas Diponegoro

Dosen Pembimbing
Kerja Praktik

Aghus Sofwan, S.T., M.T., Ph.D.
NIP. 197302041997021001

Budi Setiyono, S.T., M.T.
NIP. 197005212000121001

LEMBAR PENGESAHAN PERUSAHAAN

RANCANG BANGUN CHATBOT UNTUK

AKUISISI DATA PADA GREENHOUSE DENGAN

MENGGUNAKAN TELEGRAM DAN GOOGLE DRIVE API DI

PT PLN NUSANTARA POWER UP REMBANG

Disusun oleh:

Naufal Yafi Susanto

21060120120011

PT PLN Nusantara Power UP Rembang

09 Januari 2023 s/d 17 Februari 2023

Telah diperiksa pada tanggal:

.....

Mengetahui

Manager Pemeliharaan

Pembimbing Lapangan

PT PLN Nusantara Power UP Rembang

Asmen Lingkungan

Ali Anwar

Uslah Hidayati

NID. 7294123JA

NID. 8614159ZJY

KATA PENGANTAR

Segala puji syukur penulis panjatkan kepada Allah Subhanahu wa ta'ala, Tuhan Yang Maha Esa atas rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan laporan kerja praktik dengan judul "**RANCANG BANGUN CHATBOT UNTUK AKUISISI DATA PADA GREENHOUSE DENGAN MENGGUNAKAN TELEGRAM DAN GOOGLE DRIVE API DI PT PLN NUSANTARA POWER UP REMBANG**".

Adapun kerja praktik ini adalah salah satu mata kuliah yang harus dipenuhi oleh mahasiswa Program Studi S1 Teknik Elektro sebagai salah satu prasyarat untuk dapat menyelesaikan program pendidikan sarjana yang ada pada Departemen Teknik Elektro Fakultas Teknik Universitas Diponegoro serta menyusun laporan kerja praktik sebagai pertanggungjawaban setelah melakukan kerja praktik.

Tidak lupa penulis juga mengucapkan terima kasih dan apresiasi setinggi tingginya kepada seluruh pihak yang berkontribusi sehingga pelaksanaan kerja praktik ini dapat berjalan dengan baik. Maka penulis ingin menyampaikan terima kasih kepada:

1. Allah Subhanahu wa ta'ala, Tuhan YME atas rahmat dan karunia-Nya.
2. Bapak, Ibu serta keluarga penulis yang senantiasa mendukung dan mendoakan demi kelancaran pelaksanaan kerja praktik ini.
3. Bapak Aghus Sofwan, S.T., M.T., Ph.D. selaku Ketua Departemen Teknik Elektro Universitas Diponegoro.
4. Bapak Munawar Agus Riyadi, S.T., M.T., Ph.D. selaku Ketua Program Studi S-1 Teknik Elektro Universitas Diponegoro.
5. Bapak Budi Setiyono, S.T., M.T. selaku Dosen Pembimbing dalam pelaksanaan kerja praktik.
6. PT PLN Nusantara Power UP Rembang yang telah memberikan kesempatan kepada penulis untuk kerja praktik.
7. Bapak Ali Anwar selaku Manager Pemeliharaan PT PLN Nusantara Power UP Rembang.

8. Ibu Uslah Hidayati selaku Asmen Lingkungan sekaligus pembimbing lapangan kerja praktik.
9. Mas Frenda dan Mbak Nafi selaku Staf SDM yang telah membantu proses administrasi.
10. Mas Sa'id, Mbak Uri, Mas Qory, Mas Mumu, Mas Nasikhin, Mas Budi, dan seluruh Staf Divisi Lingkungan.
11. Serta pihak-pihak lainnya yang tidak dapat penulis sebutkan satu per satu.

Penulis menyadari masih banyak kekurangan dalam penulisan laporan akhir kerja praktik ini. Oleh karena itu, saran dan kritik yang membangun diperlukan oleh penulis untuk kebaikan dan penyusunan laporan di masa mendatang.

Akhir kata, semoga laporan ini dapat bermanfaat dan menjadi referensi bagi semua pihak khususnya bagi mahasiswa yang ingin melakukan kerja praktik di instansi yang sama ataupun yang berbeda.

Rembang, 17 Februari 2023

Naufal Yafi Susanto

DAFTAR ISI

LEMBAR JUDUL	i
LEMBAR PENGESAHAN DEPARTEMEN	ii
LEMBAR PENGESAHAN PERUSAHAAN	iii
KATA PENGANTAR	iv
DAFTAR ISI.....	vi
DAFTAR GAMBAR	ix
DAFTAR TABEL.....	xi
ABSTRAK	xii
ABSTRACT	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Batasan Masalah.....	2
1.3 Tujuan	2
1.4 Manfaat	3
1.4.1 Manfaat bagi Mahasiswa.....	3
1.4.2 Manfaat bagi Perguruan Tinggi.....	3
1.4.3 Manfaat bagi Perusahaan.....	3
1.5 Waktu dan Tempat Pelaksanaan	3
1.6 Metodologi Penelitian	4
1.7 Sistematika Penulisan.....	4
BAB II PROFIL UMUM PERUSAHAAN	6
2.1 Sejarah Singkat PT PLN Nusantara Power	6
2.2 Anak Perusahaan	9
2.2.1 PT PJB <i>Services</i>	9
2.2.2 PT Rekadaya Elektrika	9

2.2.3	PT Navigat <i>Innovative</i> Indonesia	9
2.3	Visi, Misi dan Motto Perusahaan	9
2.3.1	Visi Perusahaan.....	9
2.3.2	Misi Perusahaan	10
2.3.3	Motto Perusahaan	10
2.4	PT PLN Nusantara Power UP Rembang.....	10
2.4.1	Profil Perusahaan	10
2.4.2	Sejarah Singkat PT PLN Nusantara Power UP Rembang.....	11
2.4.3	Kebijakan LK3 (Lingkungan Keselamatan Kesehatan Kerja)	12
2.4.4	Persyaratan LK3	12
2.4.5	Struktur Organisasi	13
2.5	Siklus PT PLN Nusantara Power UP Rembang	14
2.5.1	Siklus Proses Alur Produksi Listrik	14
2.5.2	Siklus Batu Bara.....	15
2.5.3	Siklus Uap dan Air	17
2.6	Lokasi dan Layout PT PLN Nusantara Power UP Rembang.....	20
2.7	Sarana Penunjang PT PLN Nusantara Power UP Rembang	23
BAB III DASAR TEORI		28
3.1	Akuisisi Data.....	28
3.2	Chatbot	28
3.3	Telegram	29
3.4	Asynchronous Programming.....	30
3.5	JSON	31
3.6	CSV	31
3.7	OAuth 2.0.....	32

3.8	Google Drive API	33
BAB IV RANCANG BANGUN CHATBOT	35	
4.1	Analisis Kebutuhan	35
4.2	Perancangan	35
4.2.1	Use Case Diagram	35
4.2.2	Activity Diagram	36
4.3	Pengkodean	42
4.3.1	Flowchart Program	42
4.3.2	Memasukkan Library	42
4.3.3	Mengatur Parameter	43
4.3.4	Deklarasi Fungsi.....	44
4.3.5	Membuat Objek dan Menjalankan Chatbot.....	51
4.4	Hasil Chatbot.....	52
BAB V PENUTUP.....	53	
5.1	Kesimpulan	53
5.2	Saran.....	54
DAFTAR PUSTAKA	55	
BIODATA PENULIS	57	
LAMPIRAN	58	

DAFTAR GAMBAR

Gambar 2.1 Logo PT PLN Nusantara Power	10
Gambar 2.2 PT PLN Nusantara Power UP Rembang.....	11
Gambar 2.3 Struktur Organisasi PT PLN Nusantara Power UP Rembang	13
Gambar 2.4 Alur Produksi Listrik PT PLN Nusantara Power UP Rembang.....	14
Gambar 2.5 Siklus Batu Bara PT PLN Nusantara Power UP Rembang	15
Gambar 2.6 Siklus Uap dan Air PT PLN Nusantara Power UP Rembang	17
Gambar 2.7 Lokasi PT PLN Nusantara Power UP Rembang.....	21
Gambar 2.8 Layout PT PLN Nusantara Power UP Rembang.....	21
Gambar 2.9 Gedung Administrasi PT PLN Nusantara Power UP Rembang	23
Gambar 2.10 Masjid PT PLN Nusantara Power UP Rembang	23
Gambar 2.11 Kantin PT PLN Nusantara Power UP Rembang	24
Gambar 2.12 <i>Workshop</i> PT PLN Nusantara Power UP Rembang	24
Gambar 2.13 Lapangan Tenis PT PLN Nusantara Power UP Rembang.....	25
Gambar 2.14 Lapangan Parkir PT PLN Nusantara Power UP Rembang.....	25
Gambar 2.15 Perpustakaan PT PLN Nusantara Power UP Rembang	26
Gambar 2.16 Unit Kesehatan PT PLN Nusantara Power UP Rembang	26
Gambar 2.17 <i>Safety Induction</i> (K3) PT PLN Nusantara Power UP Rembang....	27
Gambar 3.1 Perbedaan <i>Synchronous</i> dan <i>Asynchronous Programming</i>	30
Gambar 3.2 Perbedaan Eksekusi <i>Concurrent</i> dan <i>Parallel</i>	30
Gambar 3.3 Contoh Format JSON	31
Gambar 3.4 Contoh Format CSV	31
Gambar 3.5 <i>Sequence Diagram</i> dari OAuth 2.0	32
Gambar 3.6 Alur Google Drive API	33
Gambar 4.1 <i>Use Case Diagram Chatbot Greenhouse</i>	35
Gambar 4.2 <i>Activity Diagram</i> Melihat Data secara <i>Realtime</i>	36
Gambar 4.3 <i>Activity Diagram</i> Mengakuisisi Data melalui Telegram.....	37
Gambar 4.4 <i>Activity Diagram</i> Mengakuisisi Data melalui Google Drive	39
Gambar 4.5 <i>Activity Diagram</i> Mengecek Koneksi <i>Database</i>	41
Gambar 4.6 <i>Flowchart</i> Program	42

Gambar 4.7 Program untuk Memasukkan <i>Library</i>	42
Gambar 4.8 Program untuk Mengatur Parameter	43
Gambar 4.9 Fungsi verify	44
Gambar 4.10 Fungsi show_data Bagian 1.....	44
Gambar 4.11 Fungsi show_data Bagian 2.....	45
Gambar 4.12 Fungsi get_csv Bagian 1	45
Gambar 4.13 Fungsi get_csv Bagian 2	46
Gambar 4.14 Fungsi get_drive Bagian 1	47
Gambar 4.15 Fungsi get_drive Bagian 2	48
Gambar 4.16 Fungsi check_db.....	49
Gambar 4.17 Fungsi error_handler	49
Gambar 4.18 Fungsi delete_drive	50
Gambar 4.19 Program untuk Membuat Objek dan Menjalankan <i>Chatbot</i>	51
Gambar 4.20 Hasil <i>Chatbot</i>	52

DAFTAR TABEL

Tabel 2.1 Unit Pembangkitan PT PLN Nusantara Power	6
Tabel 2.2 UBJ O&M PT PLN Nusantara Power	7
Tabel 2.3 UBJ O&M PT PLN Nusantara Power melalui PT PJB Services.....	8
Tabel 3.1 Daftar <i>Scopes</i> Google Drive API.....	33

ABSTRAK

Sebagai salah satu perusahaan yang peduli terhadap lingkungan, PT PLN Nusantara Power UP Rembang selalu berupaya untuk menggunakan energi baru terbarukan. Dalam penggunaan energi ini, perusahaan perlu melakukan akuisisi data secara berkala agar dapat dikaji lebih lanjut. Akan tetapi, proses akuisisi data tersebut masih dilakukan secara manual yaitu dengan menggunakan aplikasi kendali jarak jauh (TeamViewer). Cara ini tidak efektif mengingat berkas hasil akuisisi data tidak tersusun dengan rapi. Selain itu, koneksi internet yang tidak baik juga akan memperlambat dalam proses kendali jarak jauh tersebut. Dengan demikian, dibuatlah sebuah *chatbot* Telegram yang terintegrasi dengan Google Drive API untuk memudahkan proses akuisisi data tersebut. *Chatbot* akan mengirimkan berkas hasil akuisisi data berdasarkan rentang tanggal yang diinginkan. Selain itu, *chatbot* juga dibekali dengan *command* untuk menampilkan data yang terakhir disimpan secara *realtime*. Hal tersebut bertujuan untuk memudahkan dalam pemantauan akuisisi data tersebut. *Chatbot* tersedia di dalam aplikasi Telegram pada *smartphone* sehingga dapat diakses dari mana pun dan kapan pun tanpa harus menggunakan TeamViewer.

Kata kunci : akuisisi data, *chatbot*, Telegram, Google Drive API

ABSTRACT

As a company that cares about the environment, PT PLN Nusantara Power UP Rembang always strives to use new renewable energy. In using this energy, the company needs to do data acquisition periodically so that it can be studied further. However, the data acquisition process is still done manually using a remote control application (TeamViewer). This method is not effective considering that the files of data acquisition results are not neatly arranged. In addition, a poor internet connection will also slow down the remote control process. Thus, a Telegram chatbot integrated with Google Drive API was created to facilitate the data acquisition process. Chatbot will send files of data acquisition results based on the desired date range. In addition, chatbot is also equipped with a command to display the last saved data in realtime. This aims to facilitate the monitoring of data acquisition. This chatbot is available in the Telegram application on smartphones so that it can be accessed from anywhere and anytime without having to use TeamViewer.

Keywords : *data acquisition, chatbot, Telegram, Google Drive API*

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kemajuan teknologi di Indonesia menyebabkan kebutuhan akan energi listrik terus meningkat. Semua kegiatan di era modern saat ini tidak lepas dari energi listrik. Oleh karena itu, perlu adanya pembangkit listrik untuk memenuhi kebutuhan tersebut. Terdapat beberapa jenis pembangkit seperti, PLTA (Pembangkit Listrik Tenaga Air), PLTG (Pembangkit Listrik Tenaga Gas), PLTD (Pembangkit Listrik Tenaga Diesel), PLTU (Pembangkit Listrik Tenaga Uap), dan lain sebagainya.

PLTU merupakan jenis pembangkit yang paling banyak di Indonesia karena berbagai macam kelebihan yang dimilikinya, yaitu dapat dioperasikan dengan berbagai jenis bahan bakar, dapat dibangun dengan kapasitas yang bervariasi, dapat dioperasikan dengan berbagai pembebanan, serta mempunyai kontinuitas operasi dan usia pakai yang relatif lama. PLTU Rembang merupakan salah satu PLTU yang beroperasi di Indonesia dengan bahan bakar batu bara. Pembangkit ini dikelola dan dimiliki oleh PT PLN Nusantara Power yang sebelumnya merupakan PT Pembangkitan Jawa Bali (PT PJB) dengan daya yang dapat dibangkitkan sebesar 2×315 MW [1].

Pembangkit listrik ini tentunya tidak lepas dari berbagai macam polusi yang dihasilkan. Oleh karena itu, PT PLN Nusantara Power UP Rembang berkomitmen untuk menjaga lingkungan dengan menggunakan Energi Baru Terbarukan (EBT). Komitmen yang dilakukan yaitu dengan menggunakan PV (*Photovoltaic*) dan VAWT (*Vertical Axis Wind Turbine*) pada sebuah *greenhouse* yang dimiliki oleh perusahaan ini. Energi yang dihasilkan nantinya akan diakuisisi datanya untuk dikaji lebih lanjut.

Proses akuisisi data tersebut masih dilakukan dengan cara manual yaitu dengan menggunakan aplikasi kendali jarak jauh (TeamViewer). Pengguna diharuskan *login* ke dalam sistem agar dapat menyalin berkas hasil akuisisi data yang diinginkan. Cara ini tidak efektif karena pengguna harus mencari berkas data

yang tidak tersusun rapi. Selain itu, koneksi internet yang tidak baik juga akan memperlambat dalam proses kendali jarak jauh saat menggunakan aplikasi TeamViewer tersebut.

Adanya sebuah sistem yang berupa *chatbot* tentunya akan memudahkan pengguna dalam melakukan akuisisi data. Dengan demikian, dibuatlah sebuah *chatbot* Telegram yang terintegrasi dengan Google Drive API untuk memudahkan proses akuisisi data pada *greenhouse* tersebut. *Chatbot* ini akan mengirimkan berkas hasil akuisisi data berdasarkan rentang tanggal yang diinginkan sehingga pengguna tidak harus mencari berkasnya satu per satu. Selain itu, *chatbot* ini juga memiliki fitur untuk menampilkan data yang terakhir disimpan secara *realtime*. Hal tersebut bertujuan agar pengguna dapat memantau secara langsung proses akuisisi data tersebut. *Chatbot* ini dapat diakses menggunakan aplikasi Telegram pada *smartphone* sehingga pengguna dapat mengakses dari mana pun dan kapan pun tanpa harus melakukan kendali jarak jauh menggunakan TeamViewer.

1.2 Batasan Masalah

Batasan masalah dari laporan ini adalah sebagai berikut.

1. Pengembangan *chatbot* menggunakan bahasa pemrograman Python 3.9.2 dengan *library* python-telegram-bot versi 20.0.
2. *Mini Computer* yang digunakan yaitu Raspberry Pi 4 dengan sistem operasi Raspbian yang berbasis Linux Debian dengan RAM 2 GB.
3. *Database* yang digunakan yaitu MariaDB yang bersifat *local server*.
4. Ekstensi berkas yang digunakan untuk menyimpan hasil akuisisi data yaitu CSV (*Comma Separated Values*).

1.3 Tujuan

Tujuan dari pelaksanaan kerja praktik adalah sebagai berikut.

1. Memenuhi salah satu program mata kuliah wajib pada Pendidikan S-1 Teknik Elektro Fakultas Teknik Universitas Diponegoro.
2. Mendapatkan pengetahuan dan pengalaman mengenai dunia kerja sebagai bekal ketika lulus.

3. Penulis dapat menerapkan ilmu pengetahuan serta teknologi informasi yang didapatkan selama perkuliahan ke lingkungan kerja sehingga dapat mengembangkan ide dan kreativitas yang berguna.

1.4 Manfaat

1.4.1 Manfaat bagi Mahasiswa

Terbentuknya pemahaman yang terkait dengan konsep akuisisi data pada energi baru terbarukan, hubungan erat dengan implementasi teknologi informasi, serta pengalaman terkait dunia kerja dan seluk beluknya yang sangat penting untuk diketahui sebelum menginjak dunia tersebut.

1.4.2 Manfaat bagi Perguruan Tinggi

Dihasilkannya lulusan yang kompeten, menguasai bidang keilmuan, serta berpengalaman menjalani pekerjaan sesuai kebutuhan instansi di ranah teknologi informasi, khususnya pada pembuatan sistem *chatbot* untuk akuisisi data.

1.4.3 Manfaat bagi Perusahaan

Perusahaan dapat merencanakan pengembangan internal dengan memantau para mahasiswa kerja praktik yang dinilai berpotensi untuk dijadikan sebagai suksesor jangka panjang perusahaan. Selain itu, juga terdapat hubungan baik yang terjalin antara perusahaan dengan Universitas Diponegoro.

1.5 Waktu dan Tempat Pelaksanaan

Kerja Praktik ini dilaksanakan selama 30 hari, mulai tanggal 9 Januari 2023 hingga 17 Februari 2023 bertempat di PT. PLN Nusantara Power UP Rembang yang berlokasi di Kecamatan Sluke, Kabupaten Rembang.

1.6 Metodologi Penelitian

Metode pelaksanaan yang digunakan oleh penulis selama melaksanakan kerja praktik adalah sebagai berikut.

- 1. Metode wawancara dan diskusi**

Metode wawancara dan diskusi dilakukan dengan bertanya kepada pembimbing lapangan atau staf-staf lainnya untuk mendapatkan data yang tidak diperoleh di lapangan.

- 2. Metode studi literatur**

Metode studi literatur dilakukan menambah wawasan atau pengetahuan mengenai topik bahasan yang dikerjakan selama kerja praktik dengan mencari referensi dari berbagai sumber literatur.

- 3. Metode observasi**

Metode observasi dilakukan untuk mengetahui kondisi aktual mengenai sistem yang berjalan. Metode ini dilakukan dengan cara mengakses Raspberry Pi yang ada pada *greenhouse*.

- 4. Metode pengembangan sistem**

Metode pengembangan sistem yang digunakan yaitu model *incremental development*. Model *incremental development* adalah metode yang memecah *requirement software* menjadi beberapa fungsi atau bagian sehingga proses pengembangannya dapat dilakukan secara bertahap [2]. Penggunaan metode pengembangan ini bertujuan agar penulis mudah dalam mendapatkan *feedback* dari pengguna secara langsung sehingga dapat mengurangi risiko kesalahan.

1.7 Sistematika Penulisan

Dalam penyusunan laporan kerja praktik, penulis menggunakan sistematika penulisan yang dibagi menjadi beberapa bab sebagai berikut:

BAB I PENDAHULUAN

Bab pertama berisi mengenai latar belakang, batasan masalah, tujuan, manfaat, waktu dan tempat pelaksanaan, metodologi penelitian, dan sistematika penulisan.

BAB II PROFIL UMUM PERUSAHAAN

Bab kedua berisi mengenai profil umum, anak perusahaan, serta visi misi dan motto perusahaan PT PLN Nusantara Power. Selain itu, bab ini juga berisi mengenai profil perusahaan, sejarah, kebijakan LK3, persyaratan LK3, siklus, lokasi dan *layout*, serta sarana penunjang PT PLN NP Unit Pembangkitan Rembang.

BAB III DASAR TEORI

Bab ketiga berisi mengenai teori pendukung pembahasan yang meliputi *chatbot*, Telegram, *asynchronous programming*, JSON, CSV, OAuth2.0, serta Google Drive API.

BAB IV HASIL DAN PEMBAHASAN

Bab keempat berisi mengenai pembahasan dari topik kerja praktik yang meliputi analisis kebutuhan, perancangan, pengkodean, serta hasil *chatbot*.

BAB V PENUTUP

Bab kelima berisi mengenai kesimpulan yang dapat diambil dari laporan kerja praktik dan saran dalam pengembangan *chatbot* selanjutnya.

BAB II

PROFIL UMUM PERUSAHAAN

2.1 Sejarah Singkat PT PLN Nusantara Power

PT PLN Nusantara Power merupakan *subholding* dari PT PLN (Persero) yang dikukuhkan sejak September 2022 yang sebelumnya merupakan anak perusahaan dari PT PLN (Persero) dengan nama PT Pembangkitan Jawa-Bali [3]. Perusahaan ini didirikan pada tahun 1995 dengan tujuan awal melaksanakan kegiatan usaha penyediaan tenaga listrik berupa kegiatan pembangkitan tenaga listrik yang ekonomis, bermutu tinggi dan dengan keandalan yang baik. Seiring berjalannya waktu PT PLN Nusantara Power juga melaksanakan kegiatan usaha pembangunan dan pemasangan peralatan ketenagalistrikan, pemeliharaan dan pengoperasian peralatan ketenagalistrikan, serta usaha yang berkaitan dengan kegiatan perseroan dalam rangka memanfaatkan secara maksimal potensi yang dimiliki.

PT PLN Nusantara Power mengoperasikan beberapa Aset Pembangkit milik sendiri ataupun eksternal. Pada tahun 2023, perusahaan ini memiliki 9 Unit Pembangkitan (UP) dengan total kapasitas terpasang di 8 pembangkit sebesar 7.054 MW, 1 MW Pembangkit Riset di PLTS Cirata, dan sedang tahap pengembangan konstruksi Add-On Muara Tawar 2-3-4 sebesar 650 MW [4].

Tabel 2.1 Unit Pembangkitan PT PLN Nusantara Power

Unit Pembangkitan	Kapasitas (MW)	Bahan Bakar	Lokasi
PLTA Asahan	2x90	Aliran Air	Toba, Samosir, Sumatera Utara
PLTU Banjasari	2x110	Batubara	Lahat, Sumatera Utara
PLTU Jawa 7 Unit 1	2x1000	Batubara	Serang, Banten
PLTU Cilacap	2x300 1x660 1x1000	Batubara	Cilacap, Jawa Tengah
PLTU Mamuju	2x25	Batubara	Mamuju, Sulawesi Barat
PLTU Jawa 7 Unit 2	2x1000	Batubara	Serang, Banten
PLTA Batang Toru	4x127,5	Aliran Air	Tapanuli Selatan, Sumatera Utara

PLTS Terapung Cirata	1x145	Solar Power	Cirata, Purwakarta, Jawa Barat
PLTU Sumbagsel 1	2x150	Batubara	Muara Enim, Sumatera Selatan

Untuk mendukung keandalan operasi pembangkit tersebut (*unit existing*) PT PLN Nusantara Power mendirikan unit *supporting*. Terdapat tiga unit *supporting* di PT PLN Nusantara Power, yaitu:

1. Unit Pelayanan Pemeliharaan Wilayah Barat (UPHB) di Jakarta, dengan tugas utama memberikan layanan pemeliharaan unit pembangkit di Wilayah Barat, yaitu UP Muara Karang, UP Muara Tawar, dan UP Cirata.
2. Unit Pelayanan Pemeliharaan Wilayah Timur (UPHT) di Gresik dengan tugas utama memberikan layanan pemeliharaan unit pembangkitan yang ada di Wilayah Timur, yaitu: UP Gresik, UP Paiton, dan UP Brantas.
3. Badan Pengelola Waduk Cirata (BPWC) di Purwakarta, dengan tugas utama mengelola waduk Cirata untuk menjaga ketersediaan dan kualitas air untuk PLTA Cirata.

Di samping mengelola pembangkit milik sendiri, PT PLN Nusantara Power juga melaksanakan *operation and maintenance* pembangkit milik PT PLN (Persero). Hingga 2023 terdapat 14 lokasi Jasa O&M yang dioperasikan oleh PT PLN Nusantara Power dengan total daya sebesar 5.822,6 MW yang berlokasi di Jawa atau Luar Jawa sebagai berikut [1].

Tabel 2.2 UBJ O&M PT PLN Nusantara Power

UBJ O&M	Kapasitas (MW)	Bahan Bakar	Lokasi
PLTU Rembang	2x315	Batubara	Rembang, Jawa Tengah
PLTU Indramayu	3x330	Batubara	Indramayu, Jawa Timur
PLTU Pacitan	2x315	Batubara	Pacitan, Jawa Timur
PLTU Paiton 9	1x660	Batubara	Probolinggo, Jawa Timur
PLTU Tanjung Awar-awar	2x350	Batubara	Tuban, Jawa Timur
PLTGU Muara Karang Blok 2	710	Gas	Pluit, Jakarta

PLTGU Muara Tawar Blok 5	242	Gas	Bekasi, Jawa Barat
PLTGU Muara Karang Blok 5	500	Gas	Pluit, Jakarta
PLTA Ampel Gading	2x5	Tenaga air	Malang, Jawa Timur
PLTA Wonorejo	1x6,3	Tenaga air	Tulungagung, Jawa Timur
PLTMG Arun	19x9,7	Gas	Aceh
PLTU Tenayan	2x100	Batubara	Riau
PLTU Teluk Balikpapan	2x110	Batubara	Kalimantan Timur
PLTU Pulang Pisau	2x60	Batubara	Kalimantan Tengah

Jasa O&M yang dikelola PT PLN Nusantara Power melalui Anak Perusahaan PT PJB Services dan berada di luar Jawa sebesar 348 MW tersebar di 13 lokasi sebagai berikut.

Tabel 2.3 UBJ O&M PT PLN Nusantara Power melalui PT PJB Services

UBJ O&M	Kapasitas (MW)	Bahan Bakar	Lokasi
PLTU Ampana	2x3	Batubara	Sulawesi Tengah
PLTU Amurang	2x25	Batubara	Sulawesi Utara
PLTU Anggrek	2x25	Batubara	Gorontalo
PLTU Bangka	2x30	Batubara	Bangka Belitung
PLTU Belitung Baru	2x16,5	Batubara	Bangka Belitung
PLTU Bolok	2x16,5	Batubara	Nusa Tenggara Timur
PLTU Kendari Unit 1 & 2	2x20	Batubara	Sulawesi Tenggara
PLTU Kendari Unit 3	1x10	Batubara	Sulawesi Tenggara
PLTU Ketapang	2x10	Batubara	Kalimantan Barat
PLTU Ropa (Ende)	2x7	Batubara	Nusa Tenggara Timur
PLTU Tembilahan	2x7	Batubara	Riau
PLTU Tidore	2x7	Batubara	Maluku Utara
PLTU KKA	2x18	Batubara	Aceh

2.2 Anak Perusahaan

2.2.1 PT PJB Services

Didirikan tahun 2001 dengan usaha inti pada bidang operasi dan pemeliharaan pembangkit listrik, serta layanan lain yang terkait dengan pembangkit listrik. Kegiatan bisnis meliputi supervisi pemeliharaan, *commissioning*, dan operasi, operasi dan perawatan total, inspeksi dan *overhaul*, pemecahan masalah, inspeksi *bore-scope*, analisa vibrasi, *balancing* dan *alignment*, rekalibrasi alat-alat listrik, dan instrumen kontrol, pembelian dan pembaharuan suku cadang, rehabilitasi pembangkit, relokasi dan instalasi lengkap, serta teknik, pengadaan dan konstruksi.

2.2.2 PT Rekadaya Elektrika

Perusahaan ini bergerak dalam bidang jasa EPC (*Engineering, Procurement & Construction*) untuk industri kelistrikan, kepemilikan saham PLN Nusantara Power dalam perusahaan ini sebesar 98,91 persen, Saham lainnya dimiliki oleh PT Rekayasa Industri, dan YPK PLN.

2.2.3 PT Navigat Innovative Indonesia

PT Navigat Innovative Indonesia (PT NII) berdiri 21 Oktober 2002, memfokuskan diri pada bidang investasi pembangkit berbahan bakar batu bara. PT PLN Nusantara Power masuk menjadi pemegang saham PT NII sejak 12 Januari 2012, dengan cara mengambil alih 72,97 persen saham. Sejak itu PT NII resmi menjadi anak perusahaan PT PLN Nusantara Power.

2.3 Visi, Misi dan Motto Perusahaan

2.3.1 Visi Perusahaan

Visi dari PT PLN Nusantara Power UP Rembang adalah Menjadi perusahaan terpercaya dalam bisnis pembangkitan terintegrasi dengan standar kelas dunia.

2.3.2 Misi Perusahaan

Misi dari PT PLN Nusantara Power UP Rembang yaitu:

- a. Memberi solusi dan nilai tambah dalam bisnis pembangkitan terintegrasi untuk menjaga kedaulatan listrik nasional.
- b. Menjalankan bisnis pembangkitan secara berkualitas, berdaya saing dan ramah lingkungan.
- c. Mengembangkan kompetensi dan produktivitas *Human Capital* untuk pertumbuhan yang berkesinambungan.

2.3.3 Motto Perusahaan

“Produsen Listrik Terpercaya Kini dan Mendatang”

Produsen listrik terpercaya mengandung pengertian bahwa PT PLN Nusantara Power merupakan perusahaan pembangkit tenaga listrik yang andal dengan *EAF* yang tinggi, *EFOR* yang rendah dengan harga produksi sangat kompetitif. Kini dan mendatang mengandung pengertian bahwa pembangkit PT PLN Nusantara Power andal dengan harga produksi yang kompetitif bukan hanya saat ini saja, tetapi selamanya.



Gambar 2.1 Logo PT PLN Nusantara Power

2.4 PT PLN Nusantara Power UP Rembang

2.4.1 Profil Perusahaan

Nama Perusahaan : PT PLN Nusantara Power UP Rembang

Alamat Perusahaan : Jalan Raya Semarang-Surabaya KM. 130 Sluke
Rembang Jawa Tengah.

Jenis Produk : Listrik.



Gambar 2.2 PT PLN Nusantara Power UP Rembang

2.4.2 Sejarah Singkat PT PLN Nusantara Power UP Rembang

Sejarah PT PLN Nusantara Power UP Rembang dimulai dengan adanya pelaksanaan proyek percepatan Pembangkit Tenaga Listrik berbahan bakar batu bara berdasarkan pada Peraturan Presiden RI Nomor 71 Tahun 2006 tanggal 05 Juli 2006 tentang penugasan kepada PT PLN (Persero) untuk melakukan Percepatan Pembangunan Pembangkit Tenaga Listrik berbahan bakar batu bara. Perpres ini menjadi dasar bagi pembangunan 10 unit PLTU di Jawa dan 25 unit PLTU di luar Jawa Bali atau dikenal dengan nama proyek Percepatan Listrik 10.000 MW.

Proyek tersebut bertujuan untuk memenuhi kebutuhan listrik nasional yang diperkirakan mengalami kenaikan permintaan, meningkatkan mutu dan keandalan sistem penyediaan penyaluran, pelayanan listrik kepada konsumen serta mengantisipasi kenaikan harga minyak dunia yang selama ini digunakan sebagai bahan bakar, sehingga dapat menekan biaya produksi listrik.

Dalam pelaksanaan pembangunan proyek PT PLN Nusantara Power UP Rembang dengan kapasitas 2x315 MW ini ditunjuk PT PLN (Persero) Jasa Manajemen Konstruksi untuk melaksanakan supervisi selama periode konstruksi, sesuai surat penugasan Direksi PT PLN (Persero) No.00255/121/DIRKIT/2007. Operasi unit 1 dilaksanakan pada akhir tahun 2010 dan untuk unit 2 mulai beroperasi awal tahun 2011.

2.4.3 Kebijakan LK3 (Lingkungan Keselamatan Kesehatan Kerja)

1. Komitmen Perusahaan
 - a. Meminimalisasi bahaya Kesehatan & Keselamatan Kerja dan dampak Lingkungan dan terus menerus memperbaiki efektivitas sistem kerja.
 - b. Membuat program yang bertujuan untuk pembuangan limbah yang aman dan mengurangi limbah.
 - c. Meminimalisasi penggunaan energi dengan program energi yang efektif.
 - d. Menciptakan lingkungan kerja yang berlandaskan kelompok kerja dan meningkatkan kesadaran terhadap sistem kepada seluruh karyawan.
 - e. Menaati semua perundangan dan peraturan yang berlaku.
2. Target Perusahaan
 - a. *Zero Accident*
 - b. *Green & Clean Power Plant*

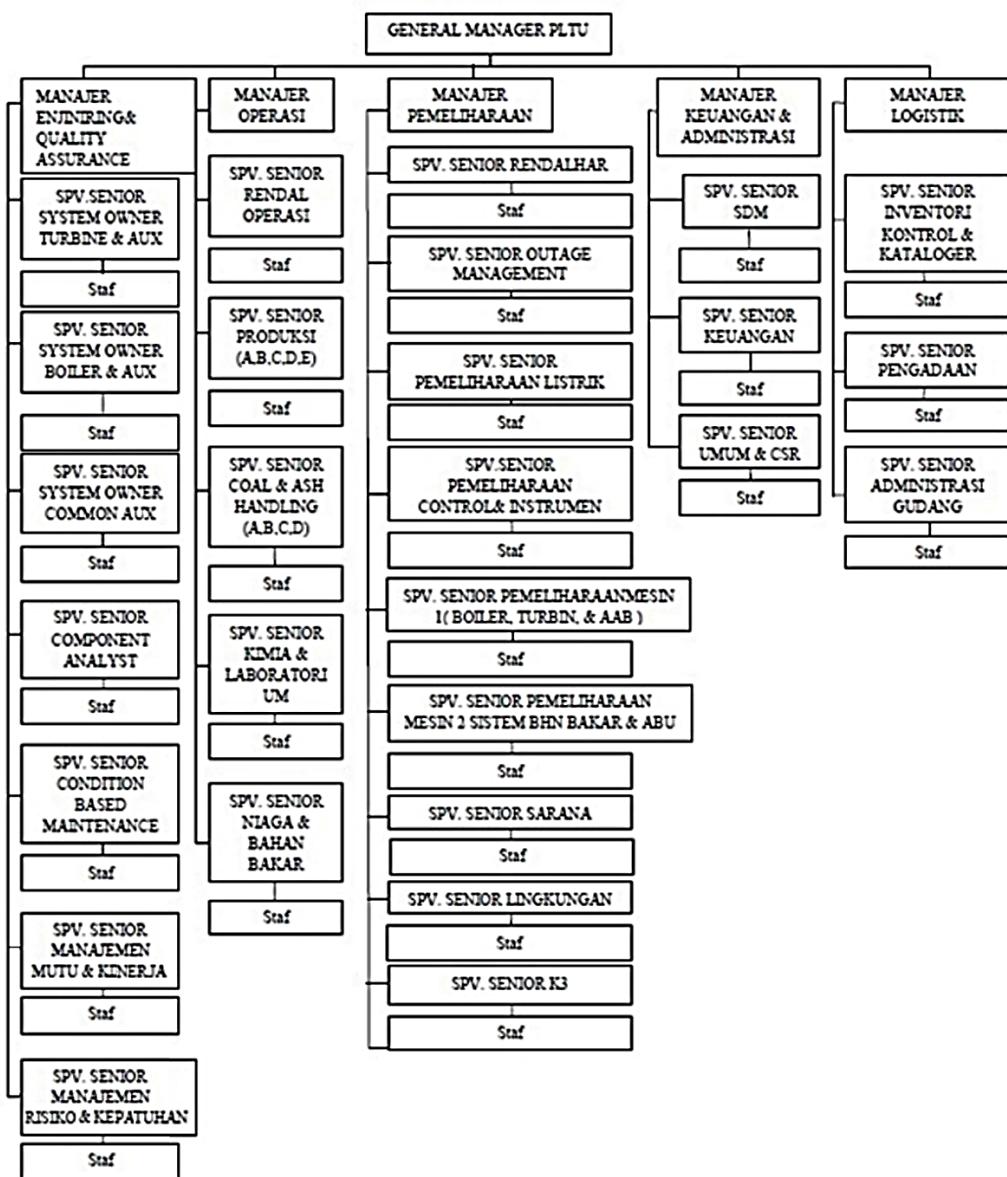
2.4.4 Persyaratan LK3

1. *Safety Induction*

Semua orang (pekerja) yang akan dipekerjakan di lapangan harus mendapatkan Pelatihan Perkenalan.
2. Sistem Manajemen Pengamanan (SMP)
 - a. Tamu yang masuk dan keluar lokasi PLTU Rembang harus tercatat.
 - b. Tamu hanya boleh masuk lokasi PLTU Rembang selama jam kerja.
 - c. Semua Tamu dan Pekerja harus memakai Tanda pengenal (*ID Card*) sebelum memasuki lokasi PLTU Rembang.
 - d. Pemeriksaan apakah barang yang dibawa dan dikeluarkan sesuai dengan surat jalan yang dibawa.
 - e. Pencatatan masuk dan keluar kendaraan tersebut dan diyakinkan kondisi kendaraan tidak akan menyebabkan adanya tumpahan.
 - f. Dalam kegiatan bongkar muat pastikan mesin kendaraan dalam keadaan mati, dan pastikan kegiatan *loading-unloading* tidak menimbulkan potensi bahaya baru di tempat kerja (penempatan material tidak menghalangi akses jalan).

2.4.5 Struktur Organisasi

Dalam tubuh PT PLN Nusantara Power UP Rembang terdapat struktur organisasi perusahaan yang memegang fungsi kendali terhadap seluruh aktivitas perusahaan. Struktur organisasi ini disusun berdasarkan keputusan PLN pusat karena PT PLN Nusantara Power UP Rembang termasuk ke dalam BUMN dan merupakan salah satu *subholding* PLN. Salah satu urgensi dari struktur organisasi dalam PT PLN Nusantara Power UP Rembang adalah merupakan pengambil kebijakan tertinggi dalam internal PT PLN Nusantara Power UP Rembang.

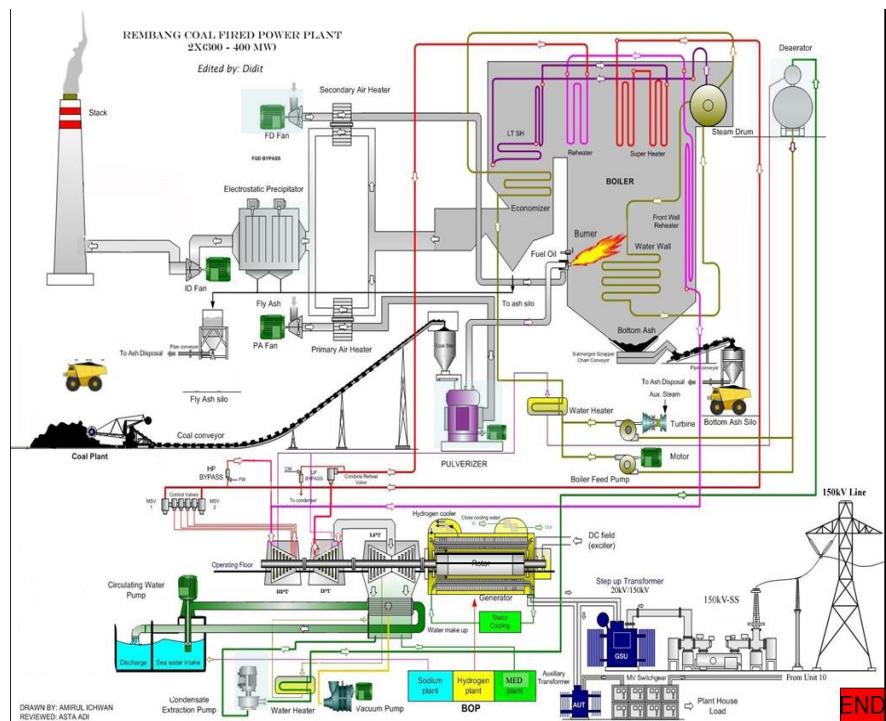


Gambar 2.3 Struktur Organisasi PT PLN Nusantara Power UP Rembang

2.5 Siklus PT PLN Nusantara Power UP Rembang

2.5.1 Siklus Proses Alur Produksi Listrik

PLTU Rembang dirancang untuk memproduksi listrik menggunakan batu bara LRC (*Low Rank Coal*) dan MRC (*Medium Rank Coal*) sebagai bahan bakar utama. HSD (*High Speed Diesel*) sebagai bahan bakar pemantik pada awal penyalaan dengan bantuan udara panas bertekanan. Panas hasil pembakaran digunakan untuk memanaskan air hingga menghasilkan uap yang digunakan untuk memutar turbin. Putaran turbin tersebut digunakan untuk memutar generator hingga menghasilkan listrik. Untuk lebih jelasnya, alur produksi listrik di PLTU Rembang dapat dilihat pada berikut.



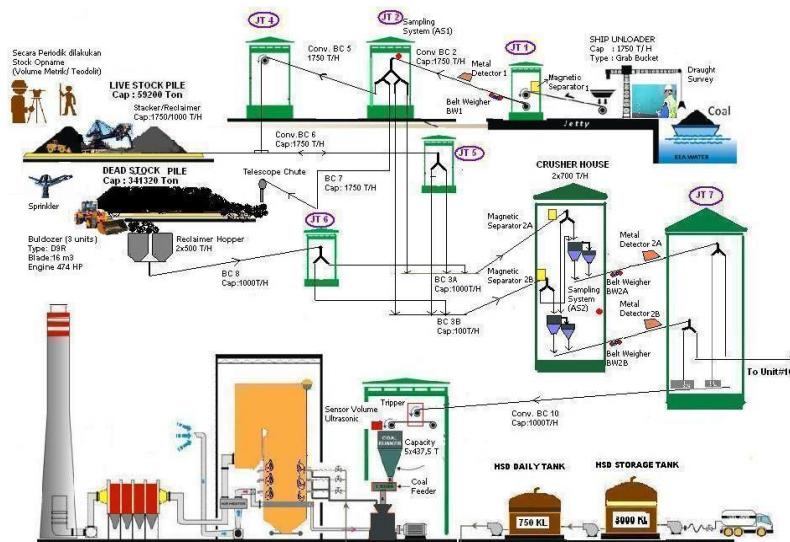
Gambar 2.4 Alur Produksi Listrik PT PLN Nusantara Power UP Rembang

Pada Gambar 2.4 dapat dibagi dalam beberapa siklus di antaranya, yaitu siklus bahan bakar (Batu bara), siklus uap dan air, dan siklus produksi listrik.

2.5.2 Siklus Batu Bara

Agar proses terus berjalan, pasokan batu bara harus selalu tersedia, sedangkan batu bara tersebut didatangkan dari Kalimantan melalui jalur laut. Sehingga dibutuhkan sistem yang dapat menampung batu bara agar pasokannya tetap tersedia untuk pembangkitan. Dalam siklus batu bara terdapat dua proses yang masing-masing memiliki opsi lajur yang dilalui, yaitu :

1. Proses *loading* merupakan proses pengisian.
2. Proses *unloading* merupakan proses pembongkaran.



Gambar 2.5 Siklus Batu Bara PT PLN Nusantara Power UP Rembang

Setiap batu bara yang ada di PLTU Rembang dipasok oleh *supplier* melalui jalur laut menggunakan tongkang. Proses pembongkaran tongkang dilakukan di *jetty* menggunakan *Ship Unloader*. *Ship Unloader* yang ada di PLTU Rembang dilengkapi 4 drum *wire – rope trolley*. Prinsip kerjanya yaitu mengangkut batu bara dari kapal tongkang atau kapal pengangkut batu bara menggunakan *grab* kemudian di unload material batu bara ke *hopper ship unloader* yang kemudian di umpulkan ke *belt conveyor*. *Belt conveyor* pada siklus ini digunakan untuk memindahkan batu bara dari tongkang menuju *coal yard* ataupun menuju unit.

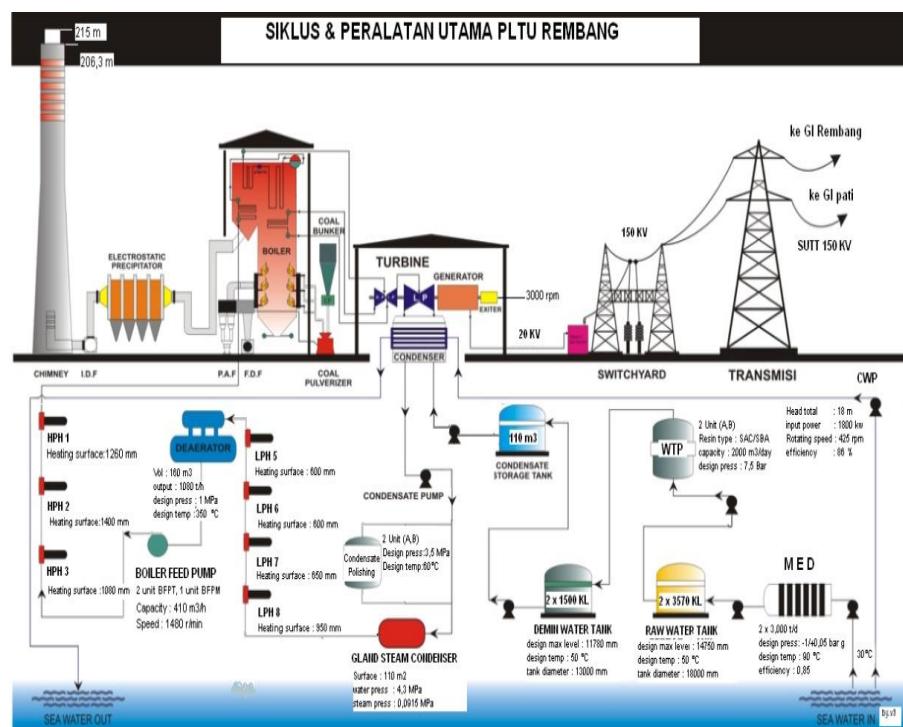
Belt Conveyor (BC) yang ada di PLTU Rembang terdapat 10 *belt conveyor* yang memiliki fungsi masing-masing. BC 1 bekerja pada area *ship unloader* ke *junction tower* 1. *Junction tower* (JT) merupakan bangunan yang berisi komponen

diverter gate yang terdapat pada *two/three way chute* yang berfungsi untuk membagi dan mengubah aliran batubara. BC2 mengalirkan batu bara dari JT1 ke JT2. BC 3 terdapat 2 bagian yaitu BC 3A dan BC 3B yang saling sejajar memiliki *head* di JT5 dan *tail* di *crusher house* yang melewati JT2, dan JT6. BC 5 mengalirkan batu bara dari JT2 ke JT4. BC 6 memiliki *head* di JT5 dan *tail* di *stacker reclaimer* yang melewati JT4. *Stacker reclaimer* merupakan alat yang digunakan untuk memindahkan batubara dari *jetty (ship unloader area)* ke *coal yard* (disebut *stacking*), maupun dari *coal yard* ke *coal bunker* (disebut *reclaiming*) melalui *belt conveyor*. Kapasitas coal yard pada PLTU Rembang adalah untuk *coal yard 1 (live stock pile)* yaitu 59.200 ton dan *coal yard 2 (dead stock pile)* yaitu 341.320 ton. JT7 mengalirkan batu bara dari JT2 ke *thelescopic chute*. *Thelescopic chute* merupakan alat bantu pembongkaran batubara pada *dead stock pile* yang ada di *coal yard*. BC 8 bekerja pada area JT6 dan *reclaimer hopper*. *Reclaimer hopper* merupakan alat yang berfungsi untuk memindahkan batu bara dari *coal yard* ke *coal bunker* yang terletak pada *dead stock pile* dan hanya digunakan sebagai alternatif untuk supply batubara ke *coal bunker* apabila *coal handling* tak dapat menyuplai dari *stacker reclaimer* maupun tak dapat *direct unloading* dari *ship unloader*. BC 9A/9B bekerja pada area *crusher house* ke JT7. *Crusher house* yang ada di PLTU Rembang berisi *crusher* dan *vibrating grizzly feeder*. *Crusher* berfungsi untuk menghancurkan batu bara yang akan masuk kedalam *coal bunker* menjadi lebih kecil sehingga akan memudahkan kerja *mill pulverizer* dalam menggerus batu bara. *Crusher* yang terdapat pada PLTU Rembang memiliki tipe *ring granulator* dengan kapasitas 700 ton/jam mampu menghancurkan batu bara dengan ukuran diatas 3 cm menjadi berukuran dibawah 3cm. Penggerak utama *crusher* memiliki daya 400 KW 1500 Rpm dengan kecepatan putar rotor 720 Rpm. *Vibrating grizzly feeder* berfungsi untuk menyeleksi batu bara. Batu bara yang sudah memiliki ukuran kurang dari 3 cm akan lanjut menuju coal bungker. Untuk ukuran batu bara yang lebih besar dari 3 cm akan diarahkan ke *crusher* untuk dihancurkan menjadi lebih kecil. *Vibrating grizzly feeder* yang ada di PLTU Rembang memiliki kapasitas 1100 TPH dengan kapasitas motor 2 x 22 KW dengan masing – masing 1000 Rpm. BC 10A/10B mengalirkan batu bara dari JT7 ke *tripper*. *Tripper* berfungsi untuk

mengarahkan batu bara ke *coal bunker*. *Coal bunker* merupakan tempat penampungan batu bara sementara sebelum memasuki *mill pulverizer* dengan melalui *coal feeder*. Pada PLTU Rembang, setiap unit terdapat 5 *coal bunker* untuk menyuplai 5 *mill pulverizer*. Setiap *coal bunker* yang dimiliki PLTU Rembang mempunyai kapasitas 8 jam produksi.

2.5.3 Siklus Uap dan Air

Air merupakan bagian yang sangat penting di dalam proses pembangkitan listrik tenaga uap. Fungsi di dalam PLTU adalah sebagai media penyalur energi dari proses pembakaran batu bara untuk menguapkan air menjadi uap kering yang digunakan untuk memutar turbin dan turbin akan memutar generator yang akan menghasilkan listrik. dalam proses air yang digunakan harus memiliki kualitas yang dibutuhkan agar proses pembangkitan listrik agar maksimal. Maka dari hal tersebut terdapat proses pengubahan air laut menjadi air tawar yang sesuai dengan yang dibutuhkan untuk diuapkan di dalam *boiler* menjadi uap panas lanjut. Siklus air di dalam sistem air utama PLTU dibagi menjadi 2 bagian, yaitu air pengisi dan air penambah.



Gambar 2.6 Siklus Uap dan Air PT PLN Nusantara Power UP Rembang

Air penambah merupakan air yang ditambahkan untuk mengisi sistem air tertutup apabila levelnya berkurang. Air penambah berasal dari air laut hasil filtrasi menggunakan saringan pantai, *bar screen*, *trashrack*, dan *traveling bar screen* untuk menghilangkan kotoran yang terbawa ketika dipompa oleh *sea water pump* menuju *clorination*. PLTU rembang memiliki 4 buah *sea water pump*. Di dalam *clorination plant* air laut ditambahkan zat klorin yang bertujuan untuk menghilangkan biota laut yang lolos dari filtrasi. Air laut hasil *clorination* akan masuk ke dalam *destilation plant* untuk menghilangkan kadar garam dari air tersebut menjadi *raw water*. *Destilation plant* PLTU Rembang memiliki kapasitas 2×3000 t/d dengan desain tekanan $-1/+0.05$ bar g dengan temperatur 90°C efisiensi 0.85. selanjutnya *raw water* ditampung ke dalam *raw water tank* yang berkapasitas $2000 \text{ m}^3/\text{h}$ untuk memenuhi kualitas air pengisi *boiler* atau *demin water* yang memiliki *conductivity* $\leq 1\mu\text{C}/\text{cm}$, kadar *Chlorida* (*Cl*) $\leq 100 \text{ ppb}$, *ferro* (*Fe*) $\leq 50 \text{ ppb}$, *SiO₂* $\leq 20 \text{ ppb}$. Untuk air yang tidak memenuhi kualitas, air tersebut akan di tampung di *waste water treatment plant* untuk dinetralkan sebelum dibuang kembali ke laut. Ketika kualitas *demin water* terpenuhi, *demin water* akan disimpan ke dalam *demin water tank* yang berkapasitas $2 \times 1500 \text{ KL}$. Ketika level dari air hasil kondensasi untuk siklus tertutup berkurang, maka akan ditambahkan *demin water*, penambahan tersebut terjadi di dalam *condensate storage tank* yang berkapasitas 110 m^3 .

Siklus air pengisi merupakan siklus air pengisi *steam drum* yang berasal dari sirkulasi tertutup air dan uap. Siklus bermula dari *condensate pump* yang mengalirkan air *condensate* menuju *dearator* dengan melewati beberapa komponen lain. Komponen tersebut di antaranya adalah *condensate polishing* yang berisi resin anion dan kation dengan desain tekanan 3.5 MPa dengan temperatur 60°C untuk menangkap kotoran yang terkandung di dalam air kondensat yang berasal dari korosi jalur air dan kebocoran kondensor. Selanjutnya melalui *gland steam condenser* dengan luasan 110 m^3 digunakan untuk melapisi atau menghalangi uap yang keluar dari celah-celah sudu turbin dengan cara menyemprotkan uap bertekanan 0.0915 MPa yang berasal dari *steam header* kemudian dikondensasikan kondensor dan digunakan untuk memanaskan air kondensat bertekanan 4.3 MPa sebelum masuk ke *deaerator*. Selanjutnya air kondensat akan memasuki pemanasan awal di dalam *low*

pressure heater (LPH) yang berbentuk *shell and tube* dengan memanfaatkan panas dari *bypass uap low pressure turbine*. PLTU Rembang memiliki 4 LPH setiap unitnya, yaitu LPH#8 dengan *heating surface* 950 mm, LPH#7 dengan *heating surface* 650 mm, LPH#6, dan LPH#5 dengan *heating surface* 600 mm. Selanjutnya air kondensat masuk ke dalam *deaerator* untuk menghilangkan O₂ yang ada di dalam dengan cara menyemprotkan uap dari *intermediate pressure turbine*. Deaerator PLTU Rembang memiliki volume 160 m³ dengan kapasitas *output* 1080 t/h *pressure* 1 MPa dengan temperatur 350°C. Selanjutnya air pengisi tersebut masuk *boiler feed pump* (BFP) untuk dipompakan ke dalam *boiler*. BFP PLTU Rembang memiliki 3 buah setiap unitnya. 2 berjenis BFPT yang di gerakkan oleh uap *boiler* dan 1 MBFP yang di gerakkan oleh motor. Semua memiliki kapasitas 410 m³/h kecepatan 1480 r/min. Sebelum masuk boiler air mengalami pemanasan awal lanjut di dalam *high pressure heater* (HPH) dengan memanfaatkan *bypass* uap panas dari *high pressure turbine*. HPH PLTU rembang berjenis *shell and tube* berjumlah 3 buah dengan HPH#3 *heating surface* sebesar 1080 mm, HPH#2 *heating surface* 1400 mm dan pada HPH#1 *heating surface* sebesar 1260 mm. Air yang masuk *boiler* dengan temperatur 277°C bertekanan 17,9 MPa. Selanjutnya air tersebut masuk *steam drum* untuk dipanaskan dan dipisahkan air dengan uap. Air akan turun melewati *down comers* dan dipanaskan lagi di dalam *wall tube* yang berkapasitas 138 m³ dan uap akan dilewatkan ke *superheater*. Di dalam *superheater* uap basah yang berasal dari *steam drum* akan dipanaskan hingga menjadi uap kering. *Superheater* memiliki bagian-bagian untuk meningkatkan suhu uap secara bertahap. Yang pertama adalah *ceiling superheater*, *low temperature superheater* memanaskan dengan temperatur 455°C untuk menghasilkan uap kering dengan temperatur 430.5°C – 431.8°C. Selanjutnya *larger platen superheater* uap kering dipanaskan dengan temperatur 466.5°C – 468.2°C untuk menghasilkan uap panas 432.8°C – 436.7°C. Selanjutnya *real platen superheater* melakukan pemanasan 508.1°C – 508.5°C hingga menghasilkan uap panas 492°C – 497.2°C. dan yang terakhir adalah *high temperature superheater* yang menghasilkan uap kering final 537.7°C – 538.9°C bertekanan 17.4 MPa. Uap panas final tersebut akan diekspansikan ke dalam HP *turbine* yang memiliki 8 tingkat dengan tekanan turun menjadi 40 kg/cm² dengan

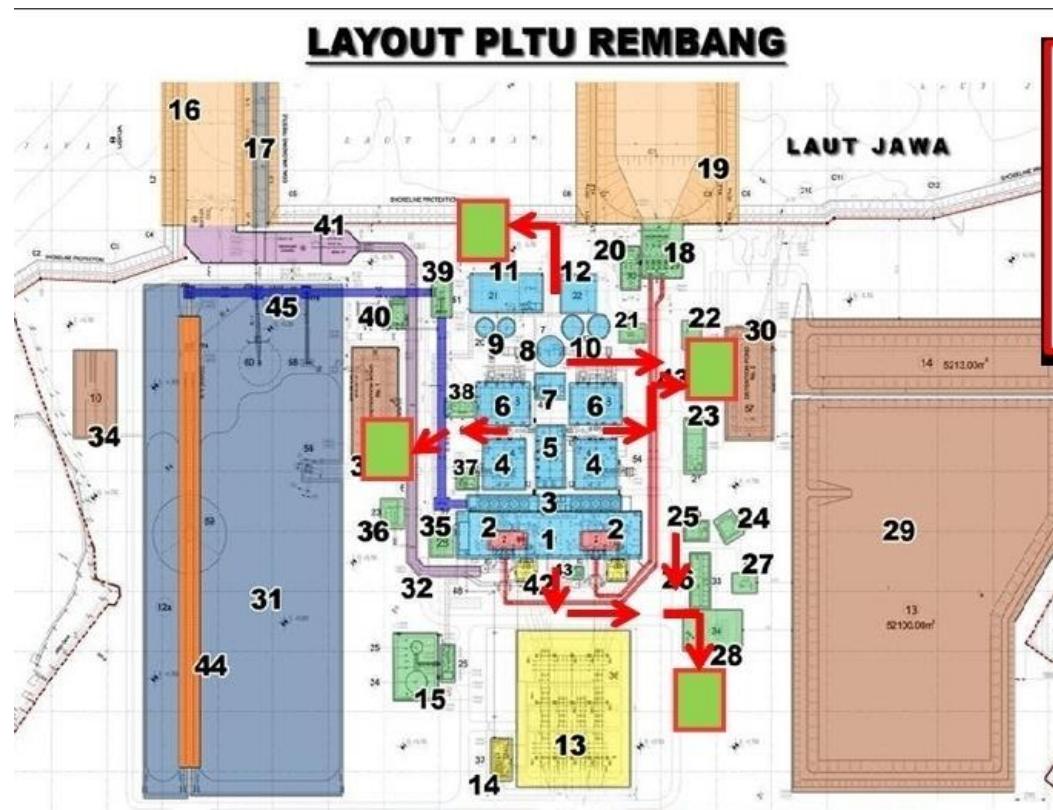
temperatur 300°C. Uap panas hasil HPH harus masuk ke dalam *reheater* di *boiler* kembali untuk menaikkan entalpinya dan temperaturnya agar menjadi 500°C kembali untuk diekspansikan kembali ke dalam *intermediate pressure turbine* menjadi bertekanan 10 kg/cm² dengan temperatur 300°C, dan langsung diekspansikan kembali menjadi bertekanan 56 mmHg (vakum) suhu 40°C. Selanjutnya uap tersebut dikondensasikan di *condenser*. *Condenser* PLTU Rembang memiliki model N-13760 dengan kapasitas *steam flow* 618 ton/h dan *circulating water flow* 48500 m³/h. Proses kondensasi uap tersebut menggunakan sistem pendinginan air terbuka dari temperatur 30°C menjadi 37°C. Sistem air terbuka ini dipompa oleh *circulating water pump* (CWP). CWP PLTU Rembang berjumlah 4 buah untuk 2 buah kondensor pada 2 unit yang masing-masing CWP memiliki kapasitas *head total* 18 m *input power* 1.8 MW. *Rotating speed* 425 RPM dengan efisiensi 86% dengan menggunakan pendinginan bertingkat (oli mendinginkan *bearing*, oli didinginkan oleh *demin water* yang disirkulasi) pada *bearing* CWP akan melewati debris filter untuk proses penyaringan agar air pendingin kondensor tidak merusak tube-tube *condenser*, air pendingin akan dibuang ke laut dengan sisi yang berjauhan dari sisi *inlet* air pendingin, dan untuk uap air dari *low pressure heater* akan berubah menjadi air kondensat lalu masuk ke dalam *condensate storage tank* untuk melanjutkan siklus tertutup.

2.6 Lokasi dan Layout PT PLN Nusantara Power UP Rembang

PT PLN Nusantara Power UP Rembang terletak di Desa Leran dan Desa Trahan, Kec. Sluke, Kab. Rembang pada koordinat 110°-111°30' BT dan 6°30'-7° LS. Lokasi PLTU ini berjarak sekitar 137 Km dari Semarang ke arah timur dan menghadap ke utara Laut Jawa serta berjarak sekitar 600 meter dari jalan utama pantai utara Jawa Tengah bagian timur.



Gambar 2.7 Lokasi PT PLN Nusantara Power UP Rembang



Gambar 2.8 Layout PT PLN Nusantara Power UP Rembang

Keterangan:	
1. <i>Turbine</i>	24. Masjid
2. <i>Generator</i>	25. Kantin
3. <i>Deaerator Bay</i>	26. <i>Administration Building</i>
4. <i>Boiler</i>	27. <i>Domestic Sewage Treatment Plant</i>
5. <i>Central Control Building</i>	28. Parkir
6. ESP	29. <i>Ash Pound</i>
7. <i>ESP Building</i>	30. <i>Deaerator</i>
8. <i>Chimney</i>	31. <i>Coal Yard</i>
9. <i>Demin Tank</i>	32. <i>Discharge Channel</i>
10. <i>Raw Water Tank</i>	33. <i>Dead Coal Yard</i>
11. <i>Water Treatment Plant</i>	34. <i>Coal Run OFF Pound</i>
12. <i>Desalination Plant</i>	35. <i>Auxiliary Boiler</i>
13. <i>Switchyard</i>	36. <i>Hydrogen Plant</i>
14. <i>Substation Control Building</i>	37. <i>Bottom Ash Silo</i>
15. <i>HSD Tank</i>	38. <i>Fly Ash Silo</i>
16. <i>Out Fall</i>	39. <i>Coal Crusher House</i>
17. <i>Jetty</i>	40. <i>Coal Handling Control Building</i>
18. <i>Pump it & Pump House</i>	41. <i>Seal Pit</i>
19. <i>Inlet Channel</i>	42. <i>Generator Transformer</i>
20. <i>Electro Chlorination Plant</i>	43. <i>Intake Pipe</i>
21. <i>Fire Fighting Pump House</i>	44. <i>Shelter</i>
22. <i>Fire Station</i>	45. <i>Junction Tower Conveyor & Belt conveyor</i>
23. <i>Workshop & Storage</i>	

2.7 Sarana Penunjang PT PLN Nusantara Power UP Rembang

1. Gedung Administrasi

Gedung administrasi PT PLN Nusantara Power UP Rembang berada di sebelah timur unit. Gedungnya terdiri dari tiga lantai, lantai pertama digunakan untuk para pegawai, lantai kedua digunakan untuk para pegawai, perpustakaan, ruang server AC dan ruang *driver*.



Gambar 2.9 Gedung Administrasi PT PLN Nusantara Power UP Rembang

2. Masjid

Masjid di PT PLN Nusantara Power UP Rembang ada satu, berada di sebelah utara gedung administrasi. Masjid ini berfungsi dengan baik ditandai dengan adanya sholat berjamaah minimal di waktu Zuhur, Ashar dan Maghrib. Hari Jumat pun diadakan Sholat Jumat di masjid ini.



Gambar 2.10 Masjid PT PLN Nusantara Power UP Rembang

3. Kantin

Kantin di PT PLN Nusantara Power UP Rembang ada satu, tempatnya berada di sebelah barat masjid. Makan siang para pegawai biasanya di kantin ini.



Gambar 2.11 Kantin PT PLN Nusantara Power UP Rembang

4. *Workshop*

Gedung *workshop* di PT PLN Nusantara Power UP Rembang berada di sebelah utara kantin. Di dalamnya berisi alat-alat khusus dan umum yang digunakan untuk proses perawatan pada unit.



Gambar 2.12 Workshop PT PLN Nusantara Power UP Rembang

5. Lapangan Tenis

Lapangan tenis di PT PLN Nusantara Power UP Rembang ada dua. Tempatnya di sebelah utara dari tempat parkir.



Gambar 2.13 Lapangan Tenis PT PLN Nusantara Power UP Rembang

6. Tempat Parkir

Tempat parkir di PT PLN Nusantara Power UP Rembang berada di sebelah timur gedung administrasi. Tempat parkir tersebut terdapat dua area, yaitu area mobil dan area sepeda motor.



Gambar 2.14 Lapangan Parkir PT PLN Nusantara Power UP Rembang

7. Perpustakaan

Perpustakaan di PT PLN Nusantara Power UP Rembang ada satu, tempatnya di dalam gedung administrasi lantai dua. Perpustakaan tersebut mempunyai berbagai buku sumber referensi.



Gambar 2.15 Perpustakaan PT PLN Nusantara Power UP Rembang

8. Unit Kesehatan

Unit kesehatan di PT PLN Nusantara Power UP Rembang ada satu, tempatnya berada di sebelah barat masjid. Klinik tersebut sebagai tempat pelayanan medis bagi karyawan maupun tamu.



Gambar 2.16 Unit Kesehatan PT PLN Nusantara Power UP Rembang

9. Safety Induction (K3)

Safety induction (K3) di PT PLN Nusantara Power UP Rembang terletak di sebelah selatan kantin. Fungsi *safety induction* adalah sebagai pengawasan keselamatan dan kesehatan kerja.



Gambar 2.17 Safety Induction (K3) PT PLN Nusantara Power UP Rembang

BAB III

DASAR TEORI

3.1 Akuisisi Data

Akuisisi data merupakan suatu sistem yang digunakan untuk mengambil, mengumpulkan dan menyiapkan data, kemudian data tersebut diolah lebih lanjut untuk keperluan tertentu [5]. Akuisisi data terdiri dari beberapa bagian, yaitu unit pemrosesan sinyal, sensor, perangkat keras, serta unit komputer. Sensor bertugas untuk melakukan konversi data variabel fisik ke variabel tegangan listrik dengan menggunakan *Analog to Digital Converter* (ADC). Data hasil pembacaan sensor tersebut nantinya akan dikirim ke komputer secara seri atau paralel agar dapat disimpan dan dikaji lebih lanjut.

3.2 Chatbot

Chatbot adalah sebuah program yang dijalankan pada sebuah komputer tertentu dengan maksud untuk menangani kegiatan percakapan kepada manusia secara otomatis. *Chatbot* hanya bertindak sesuai dengan algoritma yang dijalankan, sehingga *chatbot* tidak memiliki emosional seperti halnya manusia. Hal tersebut membuat *chatbot* hanya mengambil tindakan atau keputusan dalam sebuah percakapan dengan manusia tanpa diikuti dengan perasaan. Keuntungan menggunakan *chatbot* yaitu waktu respons yang cepat, selalu ada setiap saat, serta biaya operasional yang terjangkau [6].

Chatbot dapat diklasifikasikan berdasarkan *knowledge domain*, *service provided*, *goals*, dan *input processing and response generation method* sebagai berikut [7].

- *Knowledge Domain*

Klasifikasi ini didasari pada pengetahuan yang dapat diakses oleh *chatbot*.

Knowledge domain terdiri dari open domain yang mana *chatbot* dapat merespons topik secara umum, sedangkan *closed domain* hanya dapat merespons topik tertentu yang sudah ditentukan.

- *Service Provided*

Klasifikasi *service provided* dibedakan berdasarkan interaksi antara pengguna dengan *chatbot* serta layanan yang tersedia. Klasifikasi ini terdiri dari *interpersonal* yang mana *chatbot* hanya meneruskan informasi ke pengguna, *intrapersonal* yang mana *chatbot* menjadi bagian dari pengguna dalam kehidupan sehari-hari, serta *inter-agent* yang merupakan komunikasi antara dua sistem *chatbot* yang berbeda seperti integrasi Alexa-Cortana.

- *Goals*

Klasifikasi ini didasari pada tujuan yang ingin dicapai oleh *chatbot* itu sendiri. *Goals* yang dimaksud yaitu *informative* yang mana *chatbot* hanya memberikan informasi yang sudah ada, *chat based/conversational* yang merespons percakapan layaknya manusia, sedangkan *task based* hanya melakukan pekerjaan yang diberikan.

- *Input Processing and Response Generation Method*

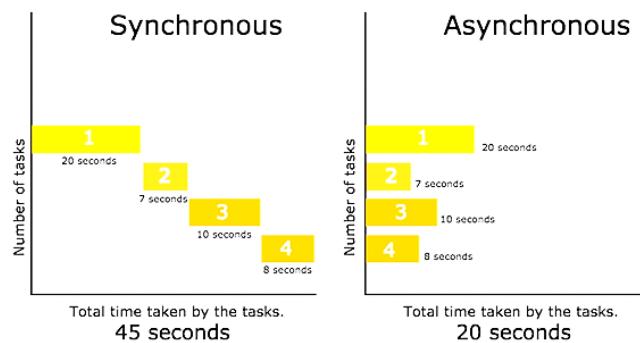
Klasifikasi ini berdasarkan pemrosesan masukan serta keluaran yang dihasilkan. Klasifikasi ini terdiri dari *intelligent system* yang mana *chatbot* akan merespons dengan bahasa alamiah layaknya manusia, *rule based system* yang berdasarkan pola atau alur yang sudah diatur, serta *hybrid system* yang merupakan gabungan dari kedua klasifikasi tersebut.

3.3 Telegram

Telegram merupakan salah satu aplikasi layanan berkirim pesan yang berbasis *cloud* dan tentunya tidak berbayar. Pengguna dapat mengirimkan pesan, foto, video, suara, atau berkas apa saja dengan menggunakan Telegram [8]. Aplikasi ini tentunya sangat aman karena menggunakan enkripsi *end-to-end*. Selain itu, Telegram juga menyediakan bot API yang memungkinkan sebuah program aplikasi untuk menggunakan pesan Telegram sebagai *interface*. Dengan menggunakan bot tersebut, pengguna dapat berinteraksi dengan program aplikasi yang berjalan sehingga dapat mendukung segala jenis tugas yang diinginkan.

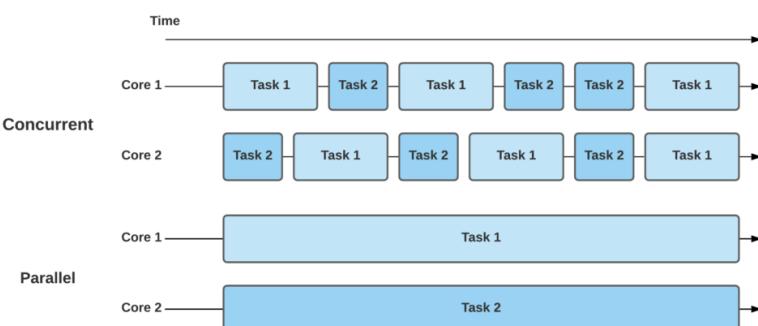
3.4 Asynchronous Programming

Asynchronous programming merupakan salah satu model pemrograman di mana dalam setiap eksekusi baris pada suatu program tidak terikat pada *input-output* atau I/O [9]. Pada model ini, proses eksekusi dilakukan secara independen karena setiap proses tidak terikat pada proses lain, tidak seperti *synchronous*. Hal tersebut membuat semua proses pada *asynchronous* ini dapat dilakukan secara bersamaan sehingga dapat memangkas waktu eksekusi yang dibutuhkan, seperti yang diilustrasikan pada gambar berikut.



Gambar 3.1 Perbedaan *Synchronous* dan *Asynchronous Programming*

Proses eksekusi program pada *asynchronous programming* dibagi menjadi dua, yaitu *concurrent* dan *parallel* [10]. Pada eksekusi *concurrent*, beberapa program akan dieksekusi secara bersamaan dengan saling bergantian dalam suatu waktu. Sebaliknya, pada eksekusi *parallel*, setiap program tersebut akan dieksekusi dalam sebuah proses secara terus menerus, seperti yang diilustrasikan pada gambar berikut.



Gambar 3.2 Perbedaan Eksekusi *Concurrent* dan *Parallel*

3.5 JSON

JSON atau *JavaScript Object Notation* merupakan jenis format yang ringan untuk penyimpanan dan pertukaran data [11]. Selain itu, format ini juga mudah dibaca dan ditulis oleh manusia sehingga mempermudah dalam pengaksesan data. JSON tidak bergantung pada bahasa pemrograman apa pun (independen) karena menggunakan gaya bahasa yang umum digunakan oleh *programmer*. Ekstensi yang digunakan oleh JSON yaitu `.json`.

```
1  {
2    "string": "Hi",
3    "number": 2.5,
4    "boolean": true,
5    "null": null,
6    "object": { "name": "Kyle", "age": 24 },
7    "array": ["Hello", 5, false, null, { "key": "value", "number": 6 }],
8    "arrayOfObjects": [
9      { "name": "Jerry", "age": 28 },
10     { "name": "Sally", "age": 26 }
11   ]
12 }
13
```

Gambar 3.3 Contoh Format JSON

3.6 CSV

Comma Separated Values (CSV) merupakan sebuah format berkas dengan menggunakan teks biasa sebagai media penyimpanan data yang setiap entrinya dibatasi dengan koma [12]. CSV mampu menyimpan data dalam jumlah yang besar karena kesederhanaan yang dimilikinya. Selain itu, tipe berkas ini juga memudahkan dalam proses ekspor maupun impor pada datanya. Oleh karena itu, CSV sering digunakan dalam pertukaran data pada aplikasi yang berbeda. Ekstensi yang digunakan oleh tipe berkas ini yaitu `.csv`.

```
mac,serial,name
00:11:22:33:44:55,1234567890,Demo_name
AA:BB:CC:DD:EE:FF,1234567891,RUT955
00:11:22:33:44:55,1234567892,Name
AA:BB:CC:DD:EE:FF,1234567893,RUT900
00:11:22:33:44:55,1234567894,RUTX11
```

Gambar 3.4 Contoh Format CSV

3.7 OAuth 2.0

OAuth 2.0 atau Open Authorization 2.0 merupakan salah satu protokol keamanan yang memungkinkan sebuah web atau aplikasi dapat mengakses sumber daya yang berada di web atau aplikasi lain [13]. Protokol ini memberikan akses serta membatasi tindakan pada *client* tanpa membagikan kredensial pengguna. Prinsip dari protokol keamanan ini yaitu penggunaan *access token* sebagai otorisasi untuk mengakses sumber daya. Versi 2.0 ini merupakan pengembangan dari versi sebelumnya yaitu OAuth 1.0.

OAuth 2.0 memiliki beberapa *role* yaitu *client*, *resource owner*, *authorization server*, dan *resource server* [14].

- *Client*

Web atau aplikasi yang ingin mengakses sumber daya.

- *Resource Owner*

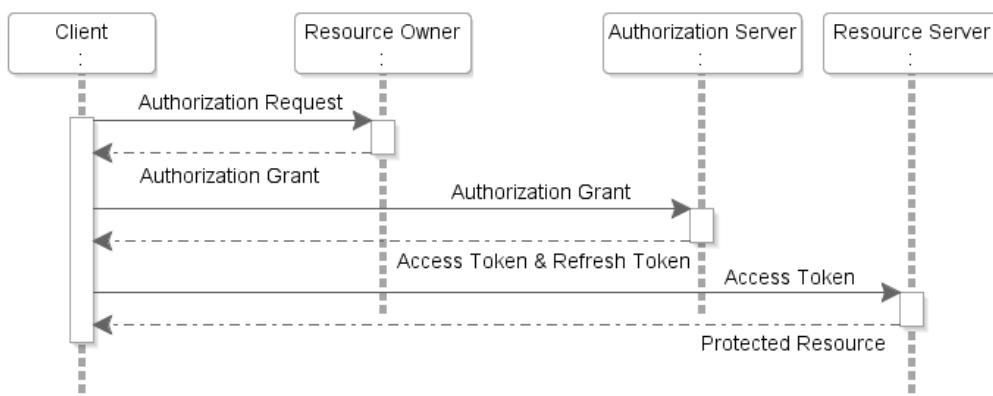
Entitas yang mengizinkan penggunaan sumber daya pada suatu web atau aplikasi dalam hal ini yaitu pemilik akun Google Drive.

- *Authorization Server*

Bertanggung jawab untuk melakukan otorisasi pada web atau aplikasi yang ingin mengakses sumber daya dengan memberikan *access token* dan *refresh token*.

- *Resource Server*

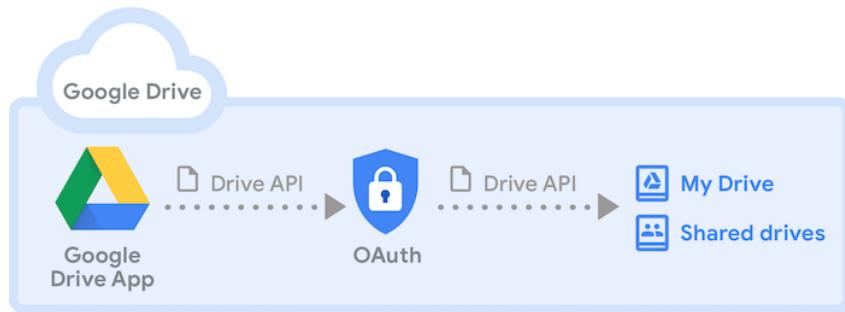
Mengirimkan permintaan sumber daya kepada *client* berdasarkan *scopes* yang ada dengan memvalidasi *access token* yang diterima.



Gambar 3.5 Sequence Diagram dari OAuth 2.0

3.8 Google Drive API

Google Drive API merupakan salah satu antarmuka pemrograman aplikasi yang disediakan oleh Google untuk memudahkan web atau aplikasi dalam mengakses penyimpanan *cloud* Google Drive [15]. Arsitektur yang digunakan pada antarmuka pemrograman aplikasi ini yaitu REST API. Agar dapat mengakses penyimpanan tersebut, web atau aplikasi harus menggunakan protokol otorisasi OAuth 2.0.



Gambar 3.6 Alur Google Drive API

Tidak semua akses diberikan langsung oleh Google Drive API. Akan tetapi, akses tersebut diberikan berdasarkan *scopes* yang ada pada *access token*. Setiap *scopes* pada OAuth 2.0 memiliki hak akses yang berbeda. Berikut merupakan daftar *scopes* yang tersedia pada Google Drive API.

Tabel 3.1 Daftar *Scopes* Google Drive API

Scopes	Penjelasan
https://www.googleapis.com/auth/drive.appdata	Melihat dan mengelola data konfigurasi aplikasi di Google Drive.
https://www.googleapis.com/auth/drive.appfolder	
https://www.googleapis.com/auth/drive.install	Mengizinkan aplikasi muncul sebagai opsi di menu "Open with" atau "New".

<code>https://www.googleapis.com/auth/drive.file</code>	Melihat dan mengelola berkas dan folder Google Drive yang Anda buka atau buat dengan aplikasi.
<code>https://www.googleapis.com/auth/drive.resource</code>	Melihat aplikasi yang diberi otorisasi untuk mengakses Google Drive.
<code>https://www.googleapis.com/auth/drive</code>	Melihat dan mengelola semua berkas dan folder Google Drive.
<code>https://www.googleapis.com/auth/drive.metadata</code>	Melihat dan mengelola metadata berkas di Google Drive.
<code>https://www.googleapis.com/auth/drive.metadata.readonly</code>	Melihat metadata berbagai berkas di Google Drive.
<code>https://www.googleapis.com/auth/drive.readonly.metadata</code>	
<code>https://www.googleapis.com/auth/drive.readonly</code>	Melihat dan mengunduh semua berkas yang ada di Google Drive.
<code>https://www.googleapis.com/auth/drive.scripts</code>	Mengubah <i>behavior</i> dari Google Apps Script.

BAB IV

RANCANG BANGUN CHATBOT

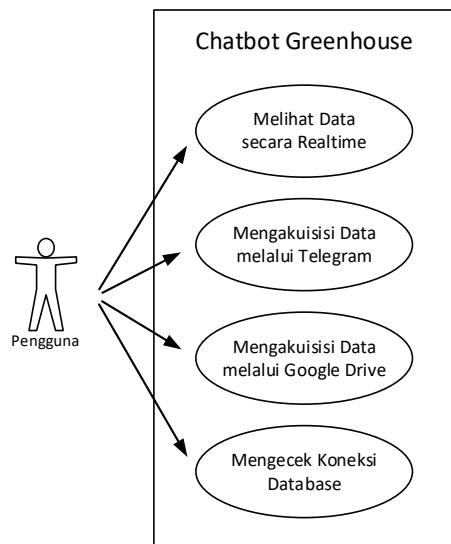
4.1 Analisis Kebutuhan

Perancangan *chatbot* dengan menggunakan Telegram bertujuan untuk memudahkan pengguna dalam proses akuisisi data yang ada pada *greenhouse*. Data yang akan diakuisisi ini tersimpan di dalam *database* lokal. Struktur *database* ini terdiri dari kolom *timestamp* yang diikuti dengan kolom-kolom lainnya. Rentang data yang ingin diakuisisi yaitu berdasarkan tanggal awal dan akhir yang nantinya akan dikirimkan oleh pengguna *chatbot*. Selain itu, pengguna juga membutuhkan layanan untuk memantau koneksi *database* serta data yang terakhir disimpan secara *realtime* untuk memastikan bahwa proses akuisisi data berjalan dengan lancar.

4.2 Perancangan

4.2.1 Use Case Diagram

Pada tahap ini dilakukan pembuatan *use case diagram* untuk menggambarkan interaksi antara pengguna dengan sistem *chatbot*.



Gambar 4.1 Use Case Diagram Chatbot Greenhouse

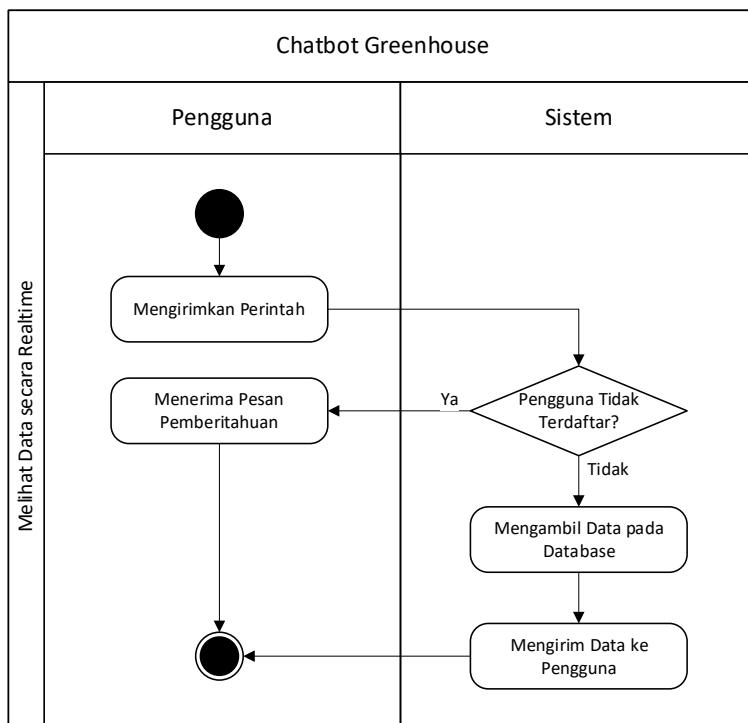
Terlihat pada Gambar 4.1, hanya terdapat satu aktor yaitu pengguna. Pengguna dapat melihat data pembacaan sensor secara *realtime*, mengakuisisi data

melalui Telegram secara langsung, mengakuisisi data melalui tautan Google Drive, serta mengecek koneksi *database* yang ada.

4.2.2 Activity Diagram

Pada tahap ini dilakukan pembuatan *activity diagram* untuk menggambarkan alur saat pengguna menggunakan *chatbot*. Terdapat empat *activity diagram* yang masing-masing menjelaskan interaksi yang ada pada *use case diagram* di atas sebagai berikut.

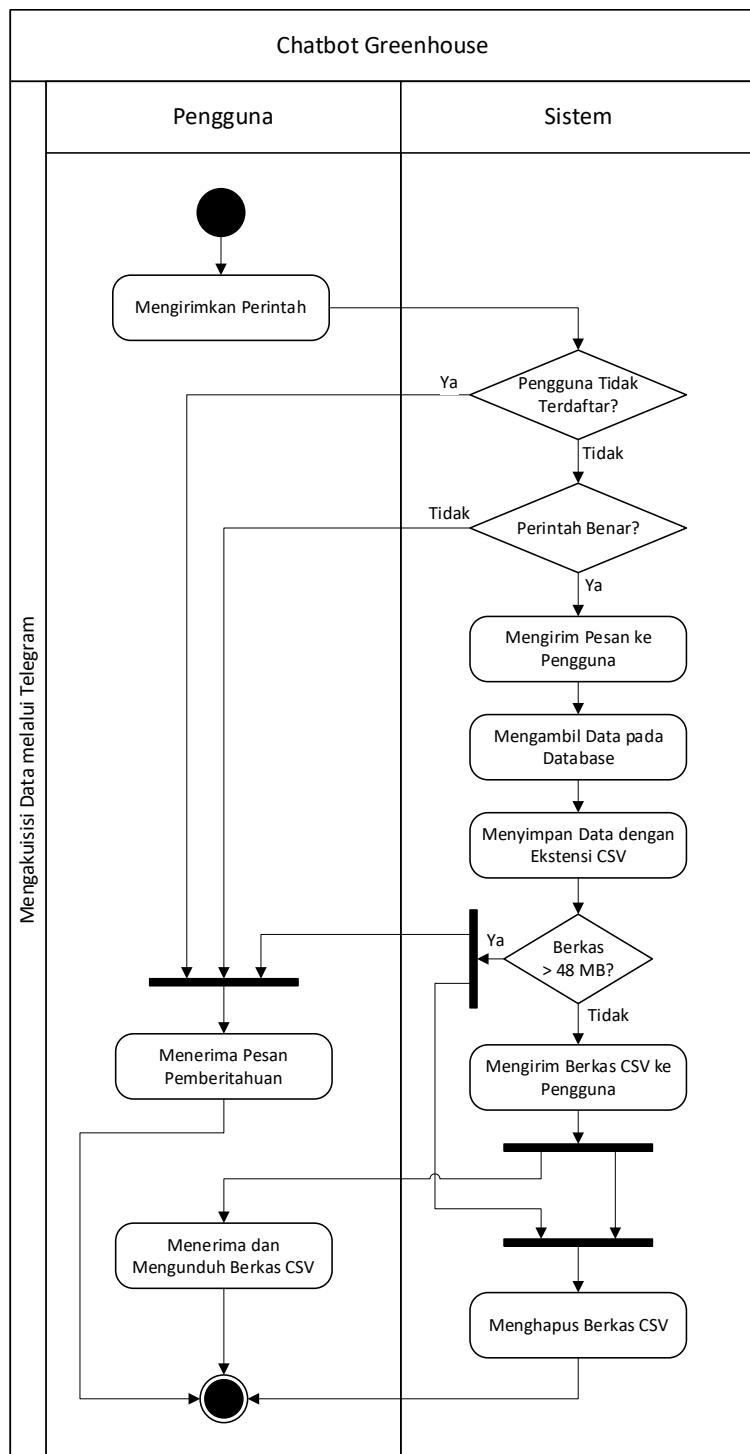
- **Melihat Data secara Realtime**



Gambar 4.2 *Activity Diagram* Melihat Data secara *Realtime*

Terlihat pada Gambar 4.2, terdapat dua *swimlane* yaitu pengguna dan sistem. Pertama, pengguna mengirimkan *command /show_data* ke *chatbot*. Sistem akan melakukan verifikasi terlebih dahulu pada *userid* pengguna. Jika tidak terdaftar, maka pengguna akan mendapatkan pesan pemberitahuan bahwa *userid* pengguna tidak terdaftar. Jika pengguna terdaftar, maka sistem akan mengambil data pada *database* lalu mengirimkannya ke pengguna.

- Mengakuisisi Data melalui Telegram



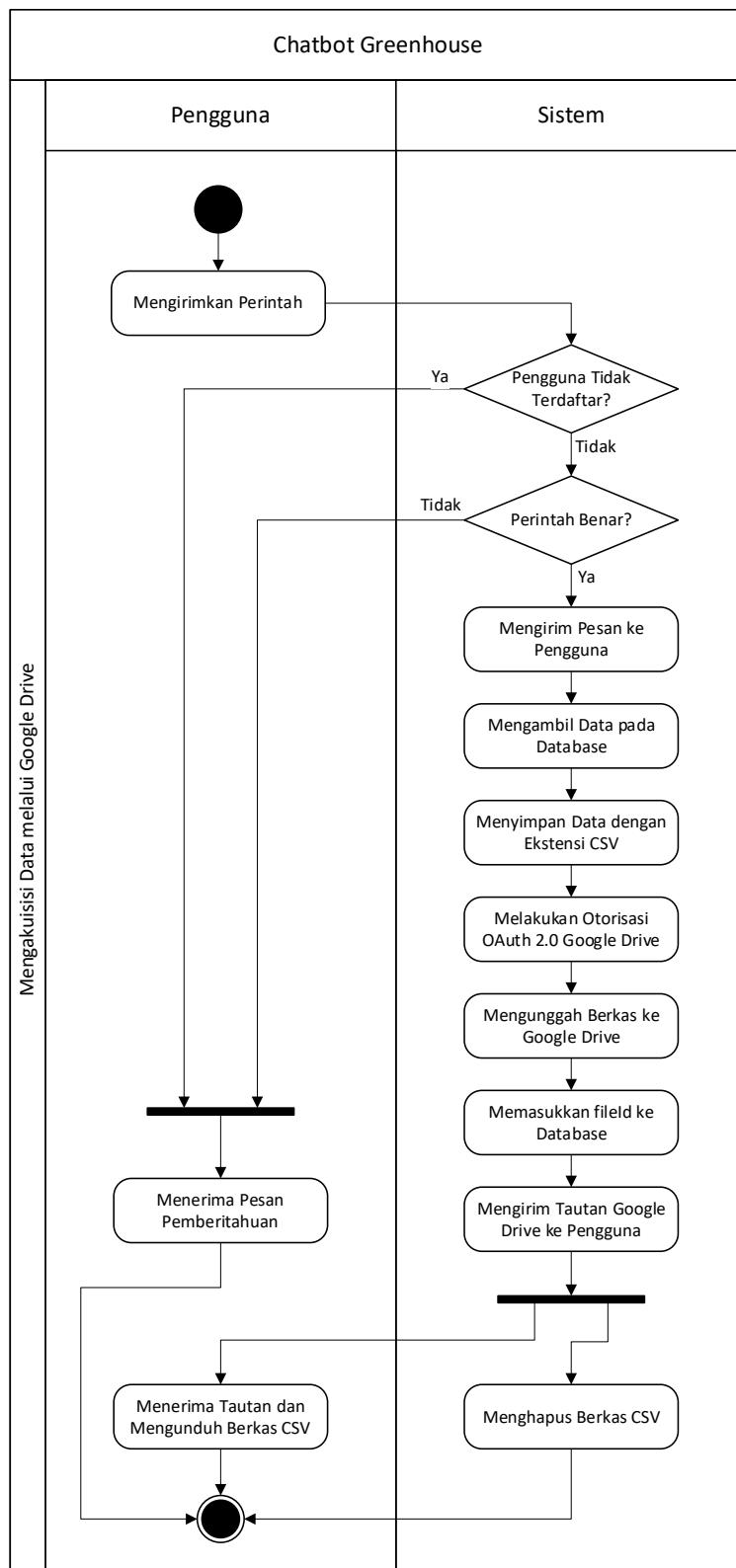
Gambar 4.3 Activity Diagram Mengakuisisi Data melalui Telegram

Terlihat pada Gambar 4.3, terdapat dua *swimlane* yaitu pengguna dan sistem. Pertama, pengguna mengirimkan *command* /get_csv YYYY-MM-DD YYYY-MM-DD ke *chatbot* dengan YYYY-MM-DD merupakan format tanggal. Sistem akan melakukan verifikasi terlebih dahulu pada *userid* pengguna. Jika tidak terdaftar, maka pengguna akan mendapatkan pesan pemberitahuan bahwa *userid* pengguna tidak terdaftar. Jika pengguna terdaftar, maka sistem akan melakukan pengecekan format tanggal.

Jika format tanggal salah, maka pengguna akan mendapatkan pesan pemberitahuan bahwa tanggal yang dimasukkan tidak benar. Jika tidak, maka sistem akan mengirimkan pesan untuk mengonfirmasi *command* yang telah dikirimkan. Setelah itu, sistem akan mengambil data pada *database* sesuai rentang tanggal yang dimasukkan lalu menyimpannya ke dalam ekstensi CSV di penyimpanan lokal sistem.

Jika ukuran berkas CSV melebihi 48 MB, maka pengguna akan mendapatkan pesan pemberitahuan yang menyarankan untuk menggunakan tautan Google Drive. Jika tidak, maka sistem akan mengirimkan berkas tersebut langsung melalui Telegram. Pengguna akan menerima berkas CSV yang dapat diunduh. Selanjutnya, sistem akan menghapus berkas tersebut dari penyimpanan lokal sistem untuk menghemat ruang penyimpanan.

- Mengakuisisi Data melalui Google Drive



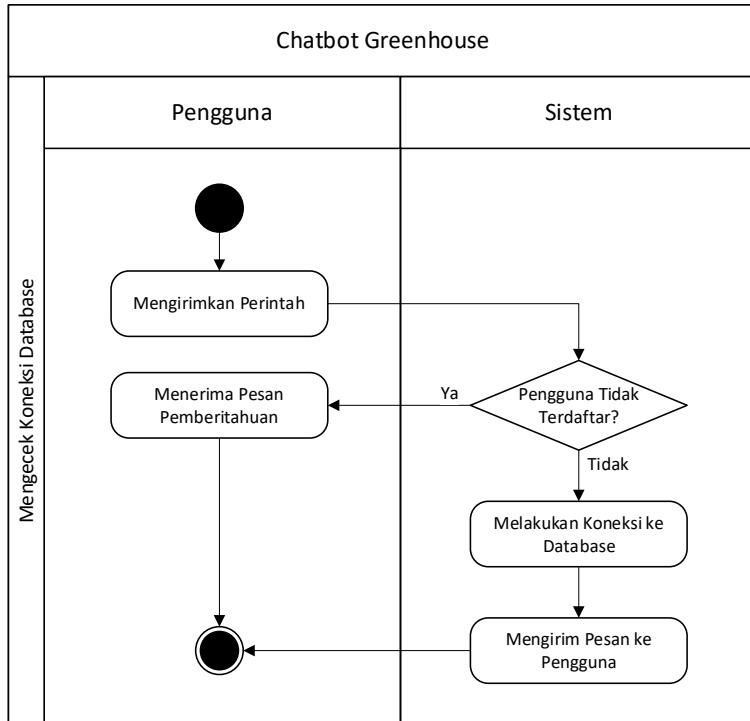
Gambar 4.4 Activity Diagram Mengakuisisi Data melalui Google Drive

Terlihat pada Gambar 4.4, terdapat dua *swimlane* yaitu pengguna dan sistem. Pertama, pengguna mengirimkan *command* /get_drive YYYY-MM-DD YYYY-MM-DD ke *chatbot* dengan YYYY-MM-DD merupakan format tanggal. Sistem akan melakukan verifikasi terlebih dahulu pada *userid* pengguna. Jika tidak terdaftar, maka pengguna akan mendapatkan pesan pemberitahuan bahwa *userid* pengguna tidak terdaftar. Jika pengguna terdaftar, maka sistem akan melakukan pengecekan format tanggal.

Jika format tanggal salah, maka pengguna akan mendapatkan pesan pemberitahuan bahwa tanggal yang dimasukkan tidak benar. Jika tidak, maka sistem akan mengirimkan pesan untuk mengonfirmasi *command* yang telah dikirimkan. Setelah itu, sistem akan mengambil data pada *database* sesuai rentang tanggal yang dimasukkan lalu menyimpannya ke dalam ekstensi CSV di penyimpanan lokal sistem.

Sistem akan melakukan otorisasi OAuth 2.0 untuk dapat mengakses API. Selanjutnya sistem mengunggah berkas CSV ke Google Drive. Setelah proses pengunggahan selesai, sistem akan menyimpan fileId ke *database* serta mengirim tautan Google Drive ke pengguna. Pengguna akan menerima tautan yang dapat diunduh dengan masa aktif satu minggu sebelum berkas dihapus oleh sistem dari Google Drive. Selanjutnya, sistem akan menghapus berkas tersebut dari penyimpanan lokal sistem untuk menghemat ruang penyimpanan.

- **Mengecek Koneksi Database**



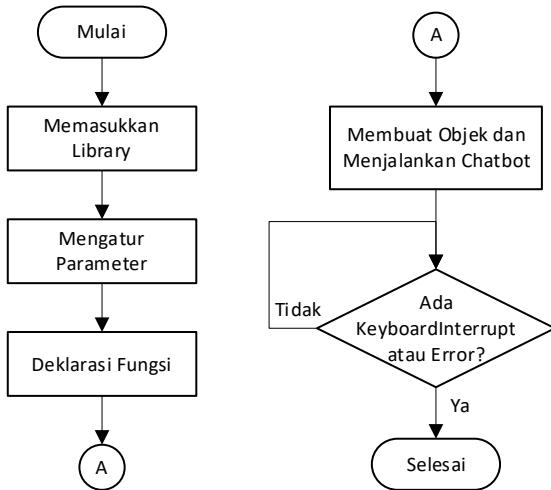
Gambar 4.5 Activity Diagram Mengecek Koneksi Database

Terlihat pada Gambar 4.5, terdapat dua *swimlane* yaitu pengguna dan sistem. Pertama, pengguna mengirimkan *command /check_db* ke *chatbot*. Sistem akan melakukan verifikasi terlebih dahulu pada *userid* pengguna. Jika tidak terdaftar, maka pengguna akan mendapatkan pesan pemberitahuan bahwa *userid* pengguna tidak terdaftar. Jika pengguna terdaftar, maka sistem akan melakukan koneksi ke *database* lalu mengirimkan status koneksi dalam bentuk pesan ke pengguna.

4.3 Pengkodean

4.3.1 Flowchart Program

Penulis menggambarkan alur program yang berjalan dengan menggunakan *flowchart* seperti yang terlihat pada gambar berikut.



Gambar 4.6 Flowchart Program

Alur program dimulai dengan memasukkan beberapa *library* yang dibutuhkan serta mengatur beberapa parameter yang digunakan. Selanjutnya, program melakukan deklarasi fungsi yang bertujuan untuk merespons terhadap setiap *command* yang dikirimkan oleh pengguna. Selain itu, terdapat fungsi lainnya yang dideklarasikan untuk menunjang fungsi utama. Setelah itu, program akan membuat objek dari sebuah *class* yang bertujuan untuk mengatur jalannya fungsi-fungsi yang telah dideklarasikan tersebut. Objek ini nantinya akan menjalankan *chatbot* secara terus menerus hingga terdapat KeyboardInterrupt atau *error* yang muncul.

4.3.2 Memasukkan Library

```
● ● ●  
1 import mysql.connector as sql, os, csv, json, traceback, logging  
2 from telegram.ext import *  
3 from datetime import datetime, timedelta  
4 from googleapiclient.discovery import build  
5 from googleapiclient.http import MediaFileUpload  
6 from google.oauth2.credentials import Credentials
```

Gambar 4.7 Program untuk Memasukkan Library

Library mysql.connector digunakan untuk menghubungkan ke *database* yang ada. Selain itu terdapat *library* os yang digunakan untuk mengelola berkas sistem, csv untuk menangani berkas CSV, serta json untuk menangani berkas JSON. Terdapat juga *library* traceback dan logging yang berfungsi untuk mendeteksi *error* serta menyimpannya ke dalam suatu berkas. *Library* datetime digunakan untuk mengakses dan memanipulasi tanggal dan waktu.

Library telegram.ext merupakan *library* utama yang digunakan untuk menjalankan *chatbot* Telegram. Selain itu, juga terdapat beberapa *library* yang digunakan untuk mengakses Google Drive API yaitu googleapiclient.discovery, googleapiclient.http, serta google.oauth2.credentials.

4.3.3 Mengatur Parameter

```
● ● ●
8 #Logger
9 logging.basicConfig(filename=r"log/bot.log",
10                         format="%(asctime)s in %(funcName)s\n  %(message)s")
11
12 #Google Drive
13 scopes = ["https://www.googleapis.com/auth/drive"]
14 path_token = r"src/token.json"
15 folder = "████████████████████████████████████████"
16
17 #Telegram
18 userid = ["████████"]
19 token = "████████████████████████████████"
20
21 #Database
22 host = "localhost"
23 user = "greenhouse"
24 password = "greenhouse"
25 database = "greenhouse"
```

Gambar 4.8 Program untuk Mengatur Parameter

Parameter yang perlu diatur yaitu folder, nama, serta format penulisan berkas *logger* yang digunakan. Selain itu, terdapat beberapa parameter Google Drive API yaitu *scopes*, *path_token*, serta folder yang digunakan untuk mengunggah berkas. Selanjutnya, parameter utama yaitu untuk *chatbot* Telegram terdiri dari *userid* serta *token*. Parameter terakhir yaitu parameter yang diperlukan untuk melakukan koneksi ke *database* yaitu *host*, *user*, *password*, dan *database* yang digunakan.

4.3.4 Deklarasi Fungsi

- **verify**

Proses verifikasi dilakukan dengan menggunakan *userid* yang dimiliki pengguna. Berkas *userid.json* dimuat ke dalam parameter *userid*. Jika *userid* pengguna terdapat di dalam parameter tersebut, maka fungsi akan mengembalikan nilai *False*. Jika tidak maka *True*.

```
● ● ●  
27 #Verifikasi User  
28 def verify(user):  
29     global userid  
30     #Memuat Data User dari JSON  
31     with open(r"src/userid.json", "r") as file:  
32         try:  
33             userid = json.load(file)  
34         except:  
35             #Error  
36             logging.warning(traceback.format_exc().replace("\n", "\n "))  
37         if str(user) in userid:  
38             return False  
39         else:  
40             return True
```

Gambar 4.9 Fungsi verify

- **show_data**

Sistem akan melakukan verifikasi pada *userid* pengguna. Jika tidak terdaftar, maka pengguna akan mendapatkan pemberitahuan bahwa *userid* tidak terdaftar. Jika pengguna terdaftar, maka sistem akan mengambil data terakhir pada *database* lalu menyimpannya ke parameter *sensor* dalam bentuk *tuple*.

```
● ● ●  
42 #Melihat Data Secara Realtime  
43 async def show_data(update, context):  
44     #Verifikasi User  
45     if verify(update.message.from_user.id):  
46         await update.message.reply_text(  
47             "Mohon maaf, User ID %s milik Anda tidak terdaftar dalam sistem kami."\  
48             % update.message.from_user.id)  
49         return None  
50     try:  
51         #Koneksi ke Database  
52         db = sql.connect(  
53             host = host,  
54             user = user,  
55             password = password,  
56             database = database)  
57         cursor = db.cursor()  
58         #Mengeksekusi Query  
59         cursor.execute("SELECT * FROM monitoring_daya ORDER BY timestamp DESC LIMIT 1")  
60         sensor = tuple(cursor.fetchall()[0])[1:]  
61         db.close()
```

Gambar 4.10 Fungsi show_data Bagian 1

Selanjutnya, sistem akan mengirimkan pesan yang berisi hasil akuisisi data dalam bentuk teks. Apabila terdapat *error* saat menjalankan fungsi ini, pengguna akan mendapatkan pemberitahuan.

```

62     #Mengirim Pesan ke User
63     await update.message.reply_text(
64         "Monitoring Daya Greenhouse\n"\
65         "- Timestamp : %s\n"\
66         "- Voltage PV : %s Volt\n"\
67         "- Current PV : %s Ampere\n"\
68         "- Power PV : %s Watt\n"\
69         "- Voltage VAWT : %s Volt\n"\
70         "- Current VAWT : %s Ampere\n"\
71         "- Power VAWT : %s Watt\n"\
72         "- Anemometer : %s m/s\n"\
73         "- Voltage Battery : %s Volt"\\
74         "% sensor")
75     except Exception as error:
76         #Error
77         logging.warning(traceback.format_exc().replace("\n", "\n "))
78     await update.message.reply_text(
79         "Mohon maaf, terjadi kesalahan sistem saat sedang memproses data. "\\
80         "Silahkan coba beberapa saat lagi.\nError : %s" % error)

```

Gambar 4.11 Fungsi show_data Bagian 2

- **get_csv**

Sistem akan melakukan verifikasi pada *userid* pengguna. Jika tidak terdaftar, maka pengguna akan mendapatkan pemberitahuan bahwa *userid* tidak terdaftar. Jika pengguna terdaftar, maka sistem memeriksa format tanggal yang dikirimkan. Jika salah, maka pengguna akan mendapatkan pemberitahuan bahwa format tanggal yang dimasukkan salah atau tidak benar.



```

82     #Mengakuisisi Data CSV Melalui Telegram
83     async def get_csv(update, context):
84         #Verifikasi User
85         if verify(update.message.from_user.id):
86             await update.message.reply_text(
87                 "Mohon maaf, User ID %s milik Anda tidak terdaftar dalam sistem kami."\\
88                 "% update.message.from_user.id")
89         return None
90     try:
91         #Memeriksa Format Perintah
92         date = str(update.message.text).split("/get_csv ")[1]
93         date_start = date.split(" ")[0]
94         date_end = date.split(" ")[1]
95         datetime.strptime(date_start, "%Y-%m-%d")
96         datetime.strptime(date_end, "%Y-%m-%d")
97     except:
98         #Format Tidak Benar
99         await update.message.reply_text(
100             "Perintah yang dimasukkan tidak benar\nFormat :\n/get_csv YYYY-MM-DD YYYY-MM-DD\n"\\
101             "Contoh :\n/get_csv 2023-01-01 2023-01-20")
102    return None

```

Gambar 4.12 Fungsi get_csv Bagian 1

Jika format tanggal benar, maka sistem akan mengirimkan pesan konfirmasi lalu mengambil data pada *database* berdasarkan rentang tanggal yang dimasukkan. Data tersebut lalu disimpan di dalam parameter `data`. Setelah itu, sistem akan menyimpan data tersebut ke dalam ekstensi CSV di penyimpanan lokal. Jika ukuran berkas tersebut melebihi 48 MB, maka pengguna akan mendapatkan pemberitahuan untuk menggunakan *command /get_drive*. Jika tidak, maka sistem akan mengirimkan berkas tersebut ke pengguna. Apabila terdapat *error* saat menjalankan fungsi ini, pengguna akan mendapatkan pemberitahuan. Setelah semuanya selesai, sistem akan menghapus berkas yang disimpan di dalam penyimpanan lokal untuk menghemat ruang penyimpanan.

```

103     try:
104         await update.message.reply_text("Mohon menunggu, sistem sedang memproses data.")
105         #Koneksi ke Database
106         db = sql.connect(
107             host = host,
108             user = user,
109             password = password,
110             database = database)
111         cursor = db.cursor()
112         #Mengeksekusi Query
113         cursor.execute(
114             "SELECT timestamp, v_pv, i_pv, p_pv, v_vawt, i_vawt, p_vawt, anemo, v_bat \"\
115             \"FROM monitoring_daya WHERE timestamp BETWEEN '%s 00:00:00' AND '%s 00:00:00' \"\
116             \"ORDER BY id ASC" % (date_start, date_end))
117         data = list(cursor.fetchall())
118         db.close()
119         #Menyimpan Data ke CSV
120         column = ["Timestamp", "PV Voltage (Volt)", "PV Current (Ampere)", "PV Power (Watt)", \
121             "VAWT Voltage (Volt)", "VAWT Current (Ampere)", "VAWT Power (Watt)", \
122             "Anemometer (m/s)", "Battery Voltage (Volt)"]
123         with open(r"cache_csv/Data Greenhouse %s to %s.csv" % (date_start, date_end),
124             mode="w", newline="") as file:
125             writer = csv.writer(file)
126             writer.writerow(column)
127             writer.writerows(data)
128             path = os.path.abspath(file.name)
129         #Memeriksa Ukuran Berkas
130         if os.path.getsize(path) >= 48000000:
131             await update.message.reply_text("Mohon maaf, berkas csv melebihi 50 MB. \"\
132                 \"Silahkan menggunakan perintah /get_drive.")
133         else:
134             with open(path, "rb") as csv_file:
135                 await update.message.reply_document(
136                     csv_file, read_timeout=120, write_timeout=120, connect_timeout=120)
137         #Error
138         except Exception as error:
139             logging.warning(traceback.format_exc().replace("\n", "\n "))
140             await update.message.reply_text(
141                 "Mohon maaf, terjadi kesalahan sistem saat sedang memproses data. \"\
142                 \"Silahkan coba beberapa saat lagi.\nError : %s" % error)
143         #Menghapus Berkas Cache
144         try: os.remove(path)
145         except: pass

```

Gambar 4.13 Fungsi `get_csv` Bagian 2

- **get_drive**

Sistem akan melakukan verifikasi pada *userid* pengguna. Jika tidak terdaftar, maka pengguna akan mendapatkan pemberitahuan bahwa *userid* tidak terdaftar. Jika pengguna terdaftar, maka sistem memeriksa format tanggal yang dikirimkan. Jika salah, maka pengguna akan mendapatkan pemberitahuan bahwa format tanggal yang dimasukkan salah atau tidak benar. Jika format tanggal benar, maka sistem akan mengirimkan pesan konfirmasi lalu mengambil data pada *database* berdasarkan rentang tanggal yang dimasukkan. Data tersebut lalu disimpan di dalam parameter *data*. Setelah itu, sistem akan menyimpan data tersebut ke dalam ekstensi CSV di penyimpanan lokal.

```
● ● ●  
147 #Mengakuisisi Data CSV Melalui Google Drive  
148 async def get_drive(update, context):  
149     #Verifikasi User  
150     if verify(update.message.from_user.id):  
151         await update.message.reply_text(  
152             "Mohon maaf, User ID %s milik Anda tidak terdaftar dalam sistem kami."\  
153             % update.message.from_user.id)  
154     return None  
155     try:  
156         #Memeriksa Format Perintah  
157         date = str(update.message.text).split("/get_drive ")[1]  
158         date_start = date.split(" ")[0]  
159         date_end = date.split(" ")[1]  
160         datetime.strptime(date_start, "%Y-%m-%d")  
161         datetime.strptime(date_end, "%Y-%m-%d")  
162     except:  
163         #Format Tidak Benar  
164         await update.message.reply_text(  
165             "Perintah yang dimasukkan tidak benar\nFormat : \n/get_drive YYYY-MM-DD YYYY-MM-DD\n"\  
166             "Contoh : \n/get_drive 2023-01-01 2023-01-20")  
167     return None  
168     try:  
169         await update.message.reply_text("Mohon menunggu, sistem sedang memproses data.")  
170     #Koneksi ke Database  
171     db = sql.connect(  
172         host = host,  
173         user = user,  
174         password = password,  
175         database = database)  
176     cursor = db.cursor()  
177     #Mengeksekusi Query  
178     cursor.execute(  
179         "SELECT timestamp, v_pv, i_pv, p_pv, v_vawt, i_vawt, p_vawt, anemo, v_bat "\\  
180         "FROM monitoring_daya WHERE timestamp BETWEEN '%s 00:00:00' AND '%s 00:00:00' "\\  
181         "ORDER BY id ASC" % (date_start, date_end))  
182     data = list(cursor.fetchall())  
183     #Menyimpan Data ke CSV  
184     column = ["Timestamp", "PV Voltage (Volt)", "PV Current (Ampere)", "PV Power (Watt)", \  
185             "VAWT Voltage (Volt)", "VAWT Current (Ampere)", "VAWT Power (Watt)", \  
186             "Anemometer (m/s)", "Battery Voltage (Volt)"]  
187     with open(r"cache_drive/Data Greenhouse %s to %s.csv" % (date_start, date_end), \  
188             mode="w", newline="") as file:  
189         writer = csv.writer(file)  
190         writer.writerow(column)  
191         writer.writerows(data)  
192         path = os.path.abspath(file.name)
```

Gambar 4.14 Fungsi get_drive Bagian 1

Sistem melakukan protokol otorisasi OAuth 2.0 dengan menggunakan parameter `path_token` yang mengarah ke `token.json` agar dapat mengakses Google Drive API. Setelah mendapatkan otorisasi, sistem akan mengunggah berkas tersebut dalam ekstensi CSV lalu mengubah *permission*-nya menjadi *reader* dan *anyone*. Selanjutnya, sistem akan memasukkan `fileId` ke dalam *database* serta mengirimkan tautan Google Drive ke pengguna yang disertai dengan batas masa berlaku. Apabila terdapat *error* saat menjalankan fungsi ini, pengguna akan mendapatkan pemberitahuan. Setelah semuanya selesai, sistem akan menghapus berkas yang disimpan di dalam penyimpanan lokal untuk menghemat ruang penyimpanan.

```

193     #Otorisasi Google Drive
194     creds = Credentials.from_authorized_user_file(path_token, scopes)
195     service = build("drive", "v3", credentials=creds)
196     #Mengunggah Berkas
197     file_metadata = {
198         "name": "Data Greenhouse %s to %s.csv" % (date_start, date_end),
199         "parents": [folder]}
200     media = MediaFileUpload(
201         path,
202         mimetype="text/csv",
203         resumable=True)
204     file = service.files().create(
205         body=file_metadata,
206         media_body=media,
207         fields="id, webViewLink").execute()
208     media.__del__()
209     #Mengatur Akses Berkas
210     permission = {
211         "role": "reader",
212         "type": "anyone"}
213     service.permissions().create(
214         fileId=file.get("id"),
215         body=permission).execute()
216     #Memasukkan fileId ke Database
217     timestamp = datetime.now()
218     cursor.execute("INSERT INTO google_drive (timestamp, file_id) VALUES ('%s', '%s')"\%
219     % (timestamp.strftime("%Y-%m-%d %H:%M:%S"), file.get("id")))
220     db.commit()
221     #Mengirim Tautan ke Telegram
222     await update.message.reply_text("Berlaku sampai %s\n%s" %
223     ((timestamp + timedelta(days=7)).strftime("%Y-%m-%d %H:%M:%S"),
224     file.get("webViewLink")))
225     #Error
226     except Exception as error:
227         logging.warning(traceback.format_exc().replace("\n", "\n "))
228         await update.message.reply_text(
229             "Mohon maaf, terjadi kesalahan sistem saat sedang memproses data.\n"
230             "Silahkan coba beberapa saat lagi.\nError : %s" % error)
231     #Menghapus Berkas Cache
232     try: os.remove(path)
233     except: pass

```

Gambar 4.15 Fungsi `get_drive` Bagian 2

- **check_db**

Sistem akan melakukan verifikasi pada *userid* pengguna. Jika tidak terdaftar, maka pengguna akan mendapatkan pemberitahuan bahwa *userid* tidak terdaftar. Jika pengguna terdaftar, maka sistem akan melakukan koneksi ke *database* lokal. Hasil dari koneksi tersebut nantinya akan dikirim ke pengguna. Jika terdapat ditemukan *error*, maka pengguna bisa melakukan tindakan lebih lanjut berdasarkan *log file* yang ada.



```
235 #Mengecek Koneksi Database
236 async def check_db(update, context):
237     #Verifikasi User
238     if verify(update.message.from_user.id):
239         await update.message.reply_text(
240             "Mohon maaf, User ID %s milik Anda tidak terdaftar dalam sistem kami.\n"
241             "%s" % update.message.from_user.id)
242     return None
243     try:
244         #Koneksi ke Database
245         db = sql.connect(
246             host = host,
247             user = user,
248             password = password,
249             database = database)
250         db.close()
251         await update.message.reply_text("Local database berhasil terhubung")
252     except Exception as error:
253         #Error
254         logging.warning(traceback.format_exc().replace("\n", "\n "))
255         await update.message.reply_text("Local database tidak terhubung\nError : %s" % error)
```

Gambar 4.16 Fungsi check_db

- **error_handler**

Fungsi ini bertugas untuk menyimpan *traceback error* apabila suatu saat terjadi *error* saat menjalankan program utama.



```
257 #Handler untuk Error
258 async def error_handler(update, context):
259     #Memasukkan Traceback Error ke Log
260     traceback_string = "".join(traceback.format_exception(None,\n
261         context.error, context.error.__traceback__))
262     logging.warning(traceback_string.replace("\n", "\n "))
```

Gambar 4.17 Fungsi error_handler

- **delete_drive**

Fungsi ini bertugas untuk menghapus berkas yang ada di Google Drive apabila sudah melewati 7 hari sejak berkas tersebut diunggah. Sistem akan mengambil data `fileId` beserta `timestamp` yang telah disimpan dalam *database*. Selanjutnya, sistem akan melakukan iterasi sebanyak `fileId` yang ada. Jika `timestamp` berkas sudah lewat 7 hari, maka berkas tersebut akan dihapus dari Google Drive. Selain itu, sistem juga akan menghapus `fileId` tersebut dari *database*. Apabila terjadi *error*, maka *traceback error* tersebut akan disimpan ke dalam *logger*.

```
● ● ●
264 #Menghapus Berkas yang Lebih dari Seminggu
265 async def delete_drive(context):
266     try:
267         #Koneksi ke Database
268         db = sql.connect(
269             host = host,
270             user = user,
271             password = password,
272             database = database
273         )
274         cursor = db.cursor()
275         cursor.execute("SELECT * FROM google_drive")
276         data = cursor.fetchall()
277         #Iterasi fileId
278         for datum in data:
279             now = datetime.now()
280             past = datum[1]
281             #Mengecek Selisih Waktu Berkas
282             if (now - past).days >= 7:
283                 try:
284                     #Menghapus Berkas di Google Drive
285                     creds = Credentials.from_authorized_user_file(path_token, scopes)
286                     service = build("drive", "v3", credentials=creds)
287                     service.files().delete(fileId=str(datum[2])).execute()
288
289                     #Menghapus Baris di Database
290                     cursor.execute("DELETE FROM google_drive WHERE id = %s" % datum[0])
291                     db.commit()
292                 except:
293                     #Error
294                     logging.warning(traceback.format_exc().replace("\n", "\n "))
295                     db.close()
296                 #Error
297             except:
298                 logging.warning(traceback.format_exc().replace("\n", "\n "))


```

Gambar 4.18 Fungsi `delete_drive`

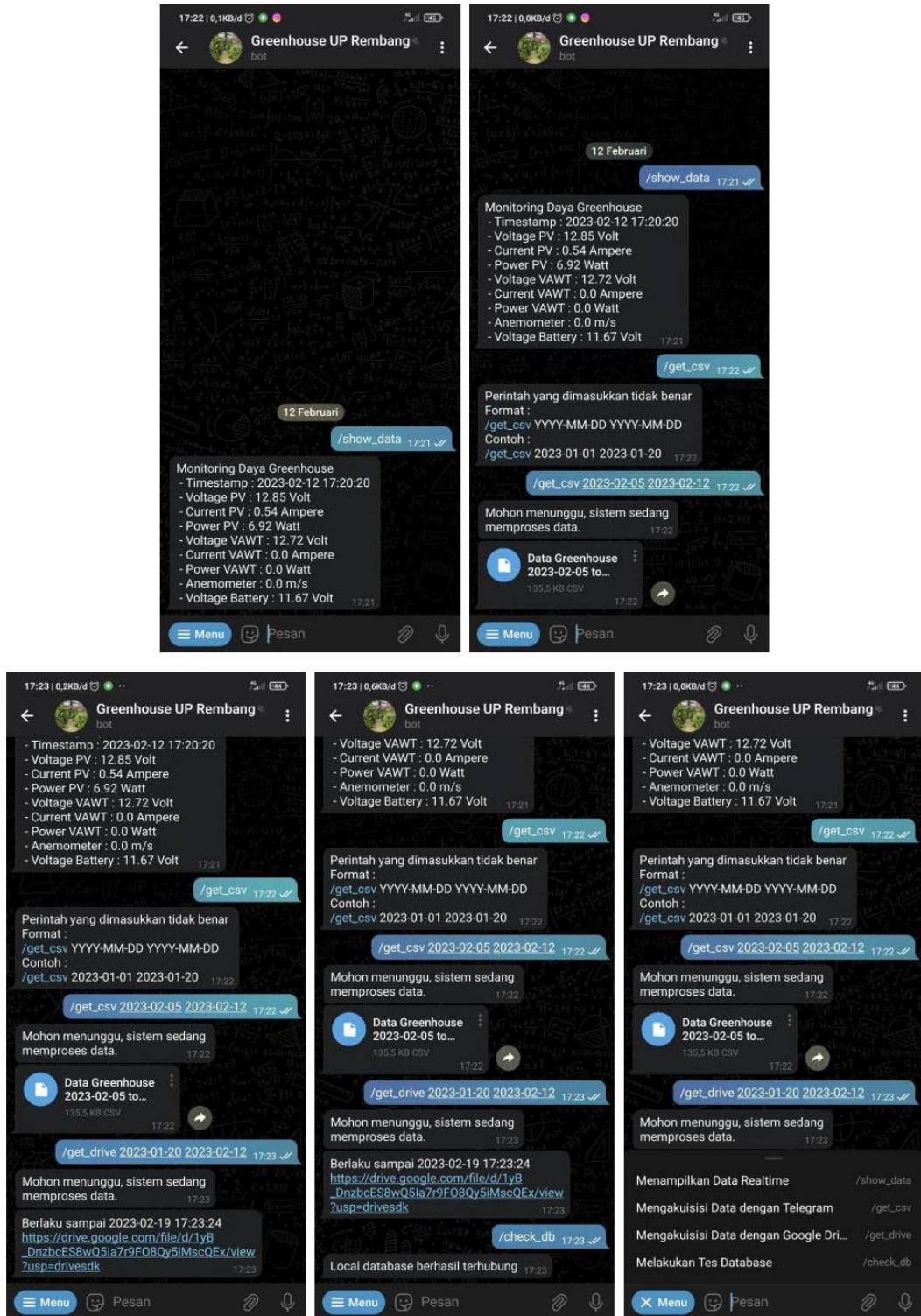
4.3.5 Membuat Objek dan Menjalankan Chatbot

```
300 if __name__ == "__main__":
301     #Membuat Objek dari class Application
302     application = Application.builder().token(token).build()
303     #Membuat Handler untuk Perintah
304     application.add_handler(CommandHandler("check_db", check_db))
305     application.add_handler(CommandHandler("show_data", show_data))
306     application.add_handler(CommandHandler("get_csv", get_csv))
307     application.add_handler(CommandHandler("get_drive", get_drive))
308     #Membuat Handler untuk Error
309     application.add_error_handler(error_handler)
310     #Membuat Job Queue untuk Menghapus Berkas
311     application.job_queue.run_repeating(delete_drive, interval=3600)
312     #Menjalankan Bot
313     application.run_polling()
```

Gambar 4.19 Program untuk Membuat Objek dan Menjalankan *Chatbot*

Program akan membuat objek dengan menggunakan parameter *token* dari Telegram yang telah tersedia dengan nama *application*. Objek ini nantinya akan mengatur jalannya fungsi *check_db*, *show_data*, *get_csv*, serta *get_drive* saat *chatbot* mendapatkan *trigger command* yang biasanya diawali dengan garis miring. Objek *application* juga akan menjalankan fungsi *error_handler* jika saat menjalankan *chatbot* terjadi *error* atau kegagalan sistem. Selain itu, objek ini juga menjalankan *job_queue* pada fungsi *delete_drive* dengan interval tiap perulangannya sebesar 3600 detik. Yang terakhir, objek akan menjalankan *method run_polling* untuk menjalankan *chatbot* secara terus menerus hingga terdapat *exception KeyboardInterrupt* atau *error*.

4.4 Hasil Chatbot



Gambar 4.20 Hasil Chatbot

BAB V

PENUTUP

5.1 Kesimpulan

Selama melaksanakan kegiatan kerja praktik, penulis mendapatkan kesimpulan sebagai berikut.

1. Telegram bot dapat digunakan sebagai *interface* yang menyediakan kecepatan dan kemudahan dalam proses akuisisi data.
2. *Concurrent* pada *asynchronous programming* lebih sering digunakan untuk mengeksekusi program yang bersifat *non-blocking* yang berhubungan dengan jaringan seperti pada program *chatbot* Telegram.
3. Pada program *chatbot* digunakan beberapa *library* seperti telegram.ext, mysql.connector, datetime, googleapiclient, google.oauth2, os, csv, json, traceback, serta logging.
4. Terdapat empat *command* yang dapat dikirim ke *chatbot*, yaitu /show_data, /get_csv, /get_drive, dan /check_db. Setiap *command* memiliki fungsi yang berbeda-beda.
5. Ekstensi CSV mempermudah proses akuisisi data dalam jumlah yang besar dengan perangkat atau aplikasi yang berbeda.
6. Ukuran berkas yang dapat dikirim melalui Telegram secara langsung maksimal 48 MB karena batasan dari API bot sebesar 50 MB.
7. Semakin jauh rentang tanggal yang diinginkan, semakin lama juga waktu yang dibutuhkan untuk memproses data tersebut.
8. Penggunaan ekstensi JSON dapat memudahkan proses penyimpanan dan pertukaran data yang dilakukan oleh sistem serta memudahkan manusia dalam melakukan penyuntingan.
9. Pengguna dengan *userid* yang tidak terdaftar dalam sistem tidak akan bisa menggunakan *command* pada *chatbot*.
10. Protokol OAuth 2.0 memudahkan sistem dalam proses otorisasi saat menggunakan API tanpa harus membagikan kredensial pengguna.

11. Google Drive API dapat digunakan untuk mengunggah, menyunting, maupun menghapus berkas secara otomatis.

5.2 Saran

Setelah penulis melaksanakan kegiatan kerja praktik, penulis masih menemui beberapa kekurangan dalam *chatbot* Telegram yang dibangun. *Chatbot* sesekali tidak merespons saat pengguna mengirimkan *command*. Hal tersebut disebabkan karena dua hal, yaitu koneksi internet tidak stabil serta mini PC mati karena kekurangan daya. Kekurangan ini dapat ditangani dengan mengganti penyedia layanan internet yang memiliki koneksi stabil serta menyediakan kebutuhan daya yang cukup selama 24 jam penuh. Selain itu, pengiriman berkas juga mengalami kendala apabila ukuran berkas terlalu besar. Hal tersebut dapat terjadi karena proses pengunggahan membutuhkan waktu yang cukup lama, sedangkan koneksi internet tidak mendukung sehingga sering terjadi *timeout* atau kehabisan waktu. Kekurangan ini dapat disiasati dengan memilih rentang waktu yang tidak terlalu jauh agar ukuran berkas tidak terlalu besar.

DAFTAR PUSTAKA

- [1] PT PLN Nusantara Power, “Operation and Maintenance,” 2023. <https://www.plnnusantarapower.co.id/operation-and-maintenance/> (accessed Mar. 01, 2023).
- [2] R. Pressman, *Software Engineering: A Practitioner’s Approach*. Boston: McGraw Hill, 2010.
- [3] PT PLN Nusantara Power, “Tentang Kami,” 2023. <https://www.plnnusantarapower.co.id/tentang-kami/> (accessed Mar. 01, 2023).
- [4] PT PLN Nusantara Power, “Unit Pembangkitan,” 2023. <https://www.plnnusantarapower.co.id/unit-pembangkitan/> (accessed Mar. 01, 2023).
- [5] Rachmad Setiawan, *Teknik Akuisisi Data*. Yogyakarta: Graha Ilmu, 2008.
- [6] P. Dewonoto, L. Santoso, I. Riski, N. Kholik, and M. R. Akbar, “Penerapan Artificial Intelligence dalam Aplikasi Chatbot sebagai Media Informasi dan Pembelajaran mengenai Kebudayaan Bangsa,” vol. 6, no. 3, pp. 579–589, 2021.
- [7] K. Nimavat and T. Champaneria, “Chatbots: An Overview Types, Architecture, Tools and Future Possibilities,” *Int. J. Sci. Res. Dev.*, vol. 5, no. 7, pp. 1019–1026, 2017.
- [8] Telegram, “Telegram FAQ,” 2023. <https://telegram.org/faq?setIn=en> (accessed Feb. 05, 2023).
- [9] Gusti Pangestu, “Asynchronous vs Synchronous Programming,” 2023. <https://binus.ac.id/malang/2022/05/asynchronous-vs-synchronous-programming/> (accessed Feb. 04, 2023).
- [10] Minh-Phuc Tran, “Sync vs. Async vs. Concurrent vs. Parallel,” 2019. <https://betterprogramming.pub-sync-vs-async-vs-concurrent-vs-parallel-5754cdb60f66> (accessed Feb. 04, 2023).
- [11] JSON, “Introducing JSON,” 2023. <https://www.json.org/json-en.html> (accessed Feb. 09, 2023).

- [12] Chris Hoffman, “What Is a CSV File, and How Do I Open It?,” 2022. <https://www.howtogeek.com/348960/what-is-a-csv-file-and-how-do-i-open-it/> (accessed Feb. 06, 2023).
- [13] Okta, “What is OAuth 2.0?,” 2023. <https://auth0.com/intro-to-iam/what-is-oauth-2> (accessed Feb. 04, 2023).
- [14] Pavan Ramchandani, “What’s the difference between OAuth 1.0 and OAuth 2.0?,” 2018. <https://hub.packtpub.com/what-is-the-difference-between-oauth-1-0-and-2-0/> (accessed Feb. 04, 2023).
- [15] Google Developers, “Introduction to Google Drive API,” 2022. <https://developers.google.com/drive/api/guides/about-sdk> (accessed Feb. 04, 2023).

BIODATA PENULIS



Naufal Yafi Susanto (21060120120011). Lahir di Kabupaten Batang pada tanggal 28 Mei 2002. Telah menempuh pendidikan mulai dari TK Aisyiyah selama 2 tahun, SD Negeri Proyonangan 12 selama 6 tahun, SMP Negeri 3 Batang selama 3 tahun, serta SMA Negeri 1 Batang selama 3 tahun. Saat ini penulis sedang menempuh pendidikan sarjana S1 Teknik Elektro di Fakultas Teknik Universitas Diponegoro Semarang dengan konsentrasi Teknologi Informasi angkatan 2020.

LAMPIRAN

1. Tampilan pengguna saat mengirimkan command /show_data ke chatbot.



2. Tampilan pengguna saat mengirimkan command /get_csv 2023-02-05 2023-02-12 ke chatbot.



3. Tampilan pengguna saat mengirimkan command `/get_drive` 2023-02-05 2023-02-12 ke chatbot.



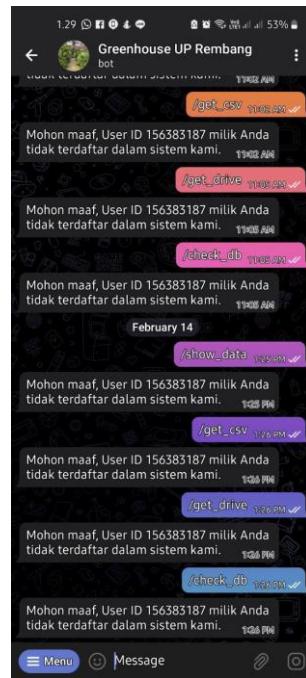
4. Tampilan pengguna saat mengirimkan command `/check_db` ke chatbot.



5. Tampilan pilihan *command* yang tersedia pada *chatbot*.



6. Tampilan pengguna yang tidak terdaftar saat mengirimkan *command*.



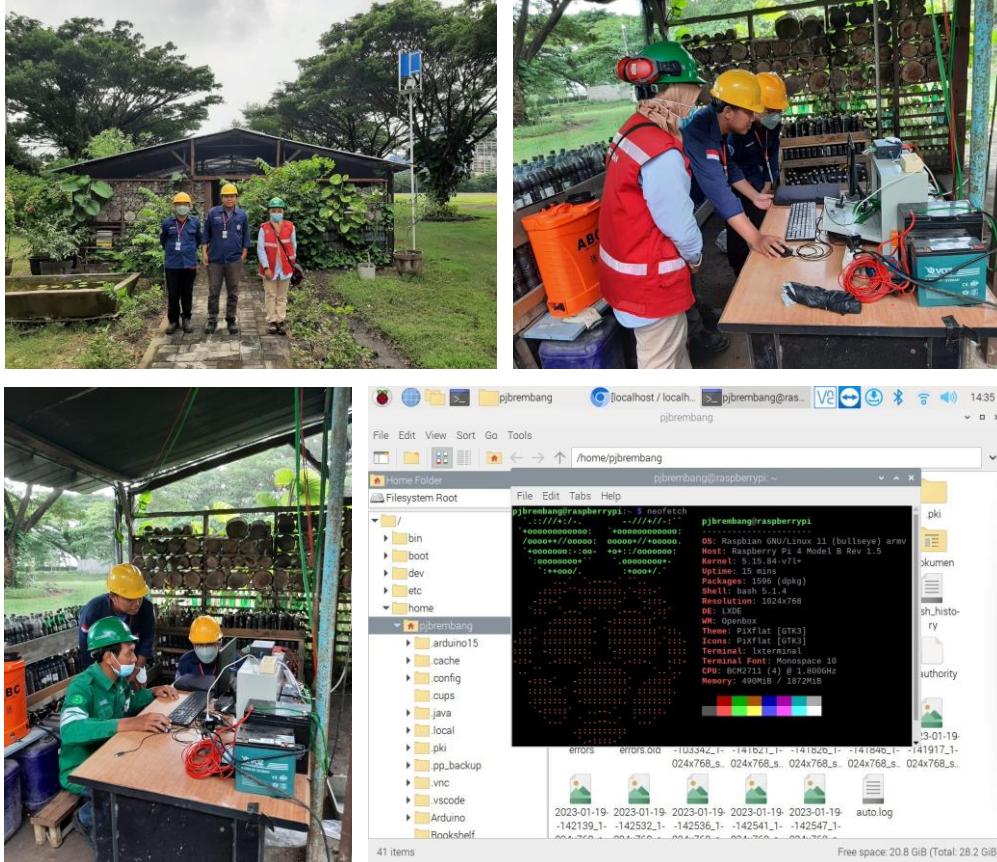
7. Tampilan hasil data yang diunduh.

```

Data Greenhouse 2023-02-05 to 2023-02-12.csv - Notepad
File Edit Format View Help
Timestamp,PV Voltage (Volt),PV Current (Ampere),PV Power (Watt),VAWT Voltage (Volt),VAWT Current (Ampere),VAWT Power (Watt),Anemometer (m/s),Battery Voltage (Volt)
2023-02-07 10:01:27,13.95,2.73,38.13,74.0,0.0,0.0,0.0,13.08
2023-02-07 10:08:54,13.99,2.82,39.47,13.79,0.0,0.0,0.0,13.15
2023-02-07 10:33:54,13.94,2.61,36.38,13.76,0.0,0.0,0.0,13.07
2023-02-07 10:38:55,13.98,2.69,37.67,13.8,0.0,0.0,0.0,13.09
2023-02-07 10:43:55,13.98,2.69,37.67,13.8,0.0,0.0,0.0,13.11
2023-02-07 10:48:56,14.05,3.15,44.28,13.85,0.0,0.0,0.0,13.18
2023-02-07 10:53:57,14.11,3.28,46.22,13.91,0.0,0.0,0.0,13.26
2023-02-07 10:58:58,14.1,3.28,46.28,13.92,0.0,0.0,0.0,13.28
2023-02-07 11:03:59,14.09,3.53,49.72,13.9,0.0,0.0,0.0,13.24
2023-02-07 11:08:59,14.14,3.84,54.23,13.94,0.0,0.0,0.0,13.33
2023-02-07 11:13:00,14.15,3.94,55.72,13.95,0.0,0.0,0.0,13.34
2023-02-07 11:18:00,14.15,3.94,55.72,13.95,0.0,0.0,0.0,13.38
2023-02-07 11:23:01,14.2,4.07,57.75,14.0,0.0,0.0,0.0,13.4
2023-02-07 11:29:02,14.22,4.18,59.49,14.01,0.0,0.0,0.0,13.42
2023-02-07 11:34:03,14.31,4.54,64.93,14.1,0.0,0.0,0.0,13.53
2023-02-07 11:39:04,14.4,4.64,66.81,14.19,0.0,0.0,0.0,13.61
2023-02-07 11:44:04,14.48,4.95,71.66,14.26,0.0,0.0,0.0,13.72
2023-02-07 11:49:05,14.61,5.18,75.69,14.39,0.0,0.0,0.0,13.88
2023-02-07 12:04:06,14.66,5.24,76.66,14.45,0.0,0.0,0.0,13.94
2023-02-07 12:19:07,14.59,4.59,66.88,14.35,0.0,0.0,0.0,13.81
2023-02-07 12:24:07,14.49,4.68,67.9,14.28,0.0,0.0,0.0,13.71
2023-02-07 12:29:08,14.54,4.73,68.75,14.32,0.0,0.0,0.0,13.77
2023-02-07 12:34:09,14.54,4.68,68.02,14.33,0.0,0.0,0.0,13.79
2023-02-07 12:39:09,14.59,4.92,71.8,14.37,0.0,0.0,0.0,13.82
2023-02-07 12:44:10,14.69,5.13,77.0,14.42,0.0,0.0,0.0,13.95
2023-02-07 12:49:11,14.74,5.28,78.1,14.48,0.0,0.0,0.0,13.97
2023-02-07 12:54:12,14.76,5.44,80.31,14.54,0.0,0.0,0.0,14.03
2023-02-07 12:59:12,14.72,4.08,69.09,14.49,0.0,0.0,0.0,13.95
2023-02-07 12:44:13,14.74,5.47,80.68,14.53,0.0,0.0,0.0,13.99
2023-02-07 12:49:14,14.98,5.01,75.03,14.74,0.0,0.0,0.0,14.16
2023-02-07 12:54:15,14.65,3.74,54.83,14.42,0.0,0.0,0.0,13.83
2023-02-07 12:59:15,14.27,3.12,44.54,14.07,0.0,0.0,0.0,13.38
2023-02-07 13:04:16,14.22,3.56,50.65,14.02,0.0,0.0,0.0,13.35
2023-02-07 13:09:17,14.22,3.56,50.65,14.02,0.0,0.0,0.0,13.35
2023-02-07 13:14:17,14.48,4.21,68.14,28.0,0.0,0.0,0.0,13.67
2023-02-07 13:19:18,14.34,2.09,29.91,14.14,0.0,0.0,0.0,13.48
2023-02-07 13:24:19,13.72,1.89,25.92,13.55,0.0,0.0,0.0,12.78
2023-02-07 13:29:20,13.78,2.67,36.83,13.62,0.0,0.0,0.0,12.81
2023-02-07 13:34:20,13.95,2.47,34.46,13.77,0.0,0.0,0.0,13.0

```

8. Dokumentasi lapangan

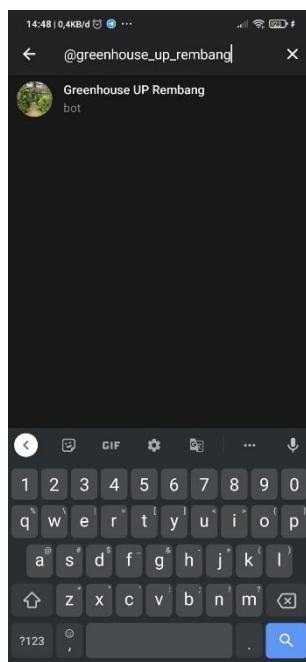


9. Prosedur penggunaan *chatbot*

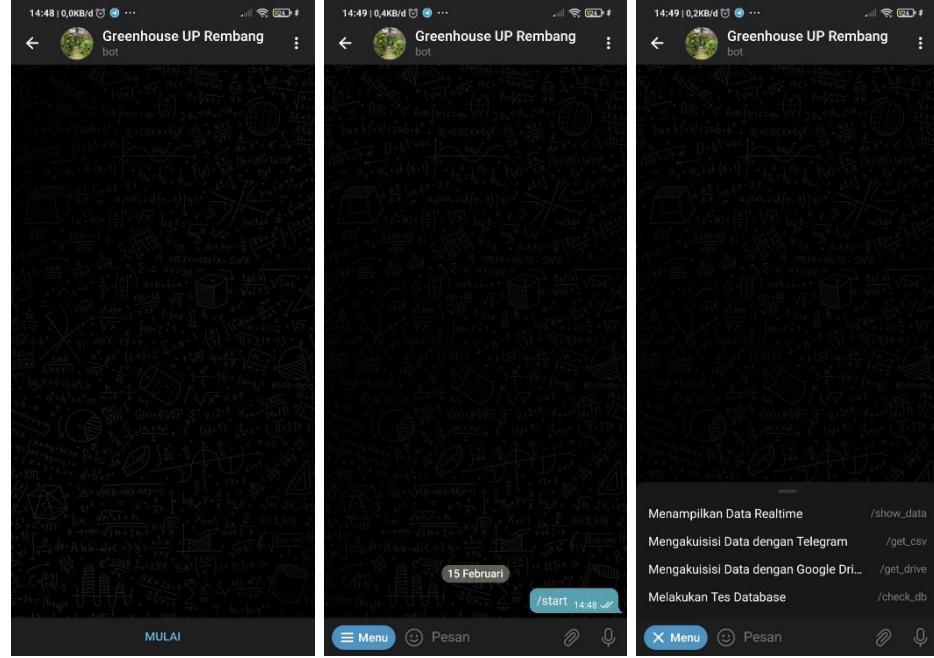
- Membuka Google Playstore lalu mengunduh aplikasi Telegram.



- Membuka aplikasi Telegram dan masuk ke menu pencarian. Ketik “@greenhouse_up_rembang_bot” di menu pencarian.



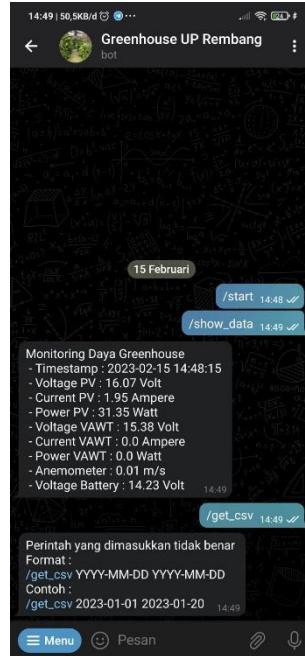
c. Memulai percakapan dengan *chatbot*. Terdapat pilihan menu yang menampilkan *command* yang tersedia.



d. Menggunakan *command* /show_data.



- e. Jika *command* yang dikirimkan salah, maka akan pengguna akan mendapatkan pesan pemberitahuan.



- f. Mengirimkan *command* yang lainnya.

