

SKRIPSI

ANALISIS KEBERLANJUTAN PENGGUNA JALA MENGUNAKAN *FACTOR ANALYSIS*



Disusun Oleh:

N a m a : Andri Wahyu Ahmad Ruslam

NIM : 17523221

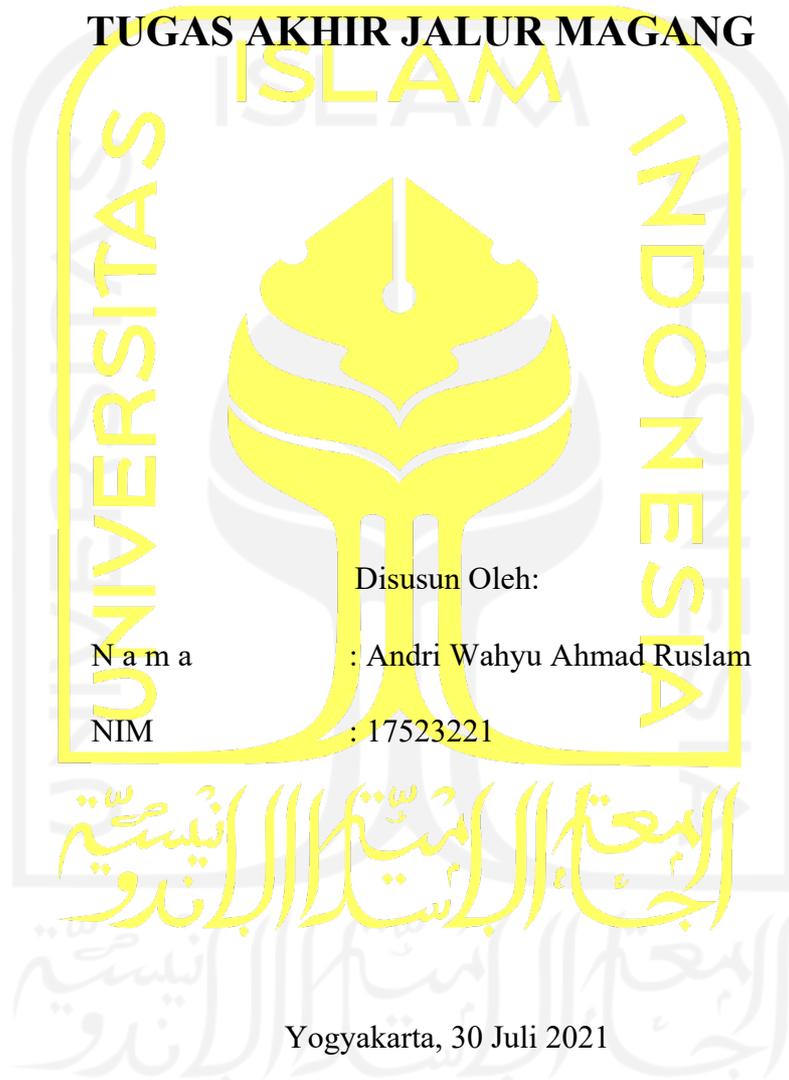
**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2021

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**ANALISIS KEBERLANJUTAN PENGGUNA JALA
MENGUNAKAN *FACTOR ANALYSIS***

TUGAS AKHIR JALUR MAGANG



Disusun Oleh:

N a m a : Andri Wahyu Ahmad Ruslam

NIM : 17523221

Yogyakarta, 30 Juli 2021

Pembimbing,

(Erika Ramadhani, S.T., M.Eng.)

HALAMAN PENGESAHAN DOSEN PENGUJI

ANALISIS KEBERLANJUTAN PENGGUNA JALA

MENGGUNAKAN *FACTOR ANALYSIS*

TUGAS AKHIR JALUR MAGANG

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 30 Juli 2021

Tim Penguji

Ketua Penguji

Erika Ramadhani, S.T., M.Eng.

Anggota 1

Sri Mulyati, S.Kom., M.Kom.

Anggota 2

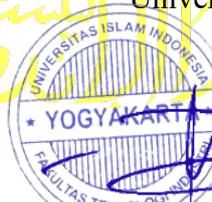
Fietyata Yudha, S.Kom., M.Kom.

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia



(Dr. Raden Teduh Dirgahayu, S.T., M.Sc.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Andri Wahyu Ahmad Ruslam

NIM : 17523221

Tugas akhir dengan judul:

ANALISIS KEBERLANJUTAN PENGGUNA JALA MENGUNAKAN *FACTOR ANALYSIS*

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 30 Juli 2021



(Andri Wahyu Ahmad Ruslam)

HALAMAN PERSEMBAHAN

Laporan penjaluran magang ini, dipersembahkan terkhusus untuk kedua orang tua yang senantiasa mendoakan dan mendukung baik materiel maupun morel. Persembahan penulisan ini juga ditujukan untuk Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi Republik Indonesia yang telah membantu dari sisi materiel berupa pemberian beasiswa selama menjalani studi di Universitas Islam Indonesia. Selain itu, persembahan juga ditujukan bagi seluruh pembaca yang ingin menjadikan tulisan ini sebagai bahan referensi untuk penjaluran magang.



HALAMAN MOTO

“Maka sesungguhnya di balik kesulitan pasti ada kemudahan. Sesungguhnya di balik kesulitan itu ada kemudahan” - (Q.S. Al-Insyirah: 5-6)

“Jika kamu tidak sanggup menahan lelahnya belajar, maka kamu harus bersabar menahan perihnya kebodohan” - Imam Syafi'i

“We are what we repeatedly do. Excellence, then, is not an act, but a habit” - Aristotle



KATA PENGANTAR

Segala puji bagi Allah *subhanahu wata'ala* yang telah memberikan kemudahan, rahmat, serta hidayah sehingga penulisan Laporan Akhir Magang ini dapat diselesaikan tepat waktu. Selawat serta salam dihaturkan kepada nabi Muhammad *Shallallahu 'alaihi wasallam* sebagai idola dan panutan dalam menjalani kehidupan sehari-hari dan berharap pertolongannya di hari penghisaban kelak.

Laporan Akhir Magang ini ditujukan sebagai laporan aktivitas selama menjalani program magang dan menjadi salah satu persyaratan untuk menyelesaikan studi Sarjana di Program Studi Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia.

Tentunya dalam menjalani aktivitas magang dan menyusun laporan ini terdapat banyak pihak yang terlibat. Oleh karena itu, ucapan terima kasih dihaturkan kepada seluruh keluarga, dosen, supervisor, rekan kerja, dan sahabat yang telah terlibat. Beberapa diantaranya adalah sebagai berikut.

1. Kedua orang tua, kakak, dan seluruh keluarga yang telah memberikan semangat berupa dorongan moral dan doa agar penyusunan laporan ini dapat terselesaikan.
2. Ibu Erika Ramadhani, S.T., M.Eng., selaku dosen pembimbing tugas akhir yang telah bersedia untuk meluangkan waktu dalam membimbing dan mengarahkan selama proses program magang dilaksanakan.
3. Bapak Syauqy Aziz selaku CTO (*Chief Technology Officer*) yang sudah memberikan kesempatan untuk melaksanakan program magang di tim *software* Jala.
4. Bapak Luqman Hakim selaku *Lead Data Scientist* yang telah membimbing dalam melaksanakan tugas dan tanggung jawab selama magang sebagai *data scientist*.
5. Ibu Aridhayanti Arifin. S.T., M.Cs., dan seluruh dosen penanggung jawab mahasiswa magang. Bapak/Ibu dosen senantiasa memberikan arahan, panduan, dan nasehat selama menjalankan program magang agar mampu menjadi karyawan yang dibutuhkan perusahaan.
6. Teman-teman kontrakan *utsman bin affan* yang senantiasa berbagi semangat maupun informasi seputar penugasan akhir sehingga laporan magang bisa selesai tepat waktu.

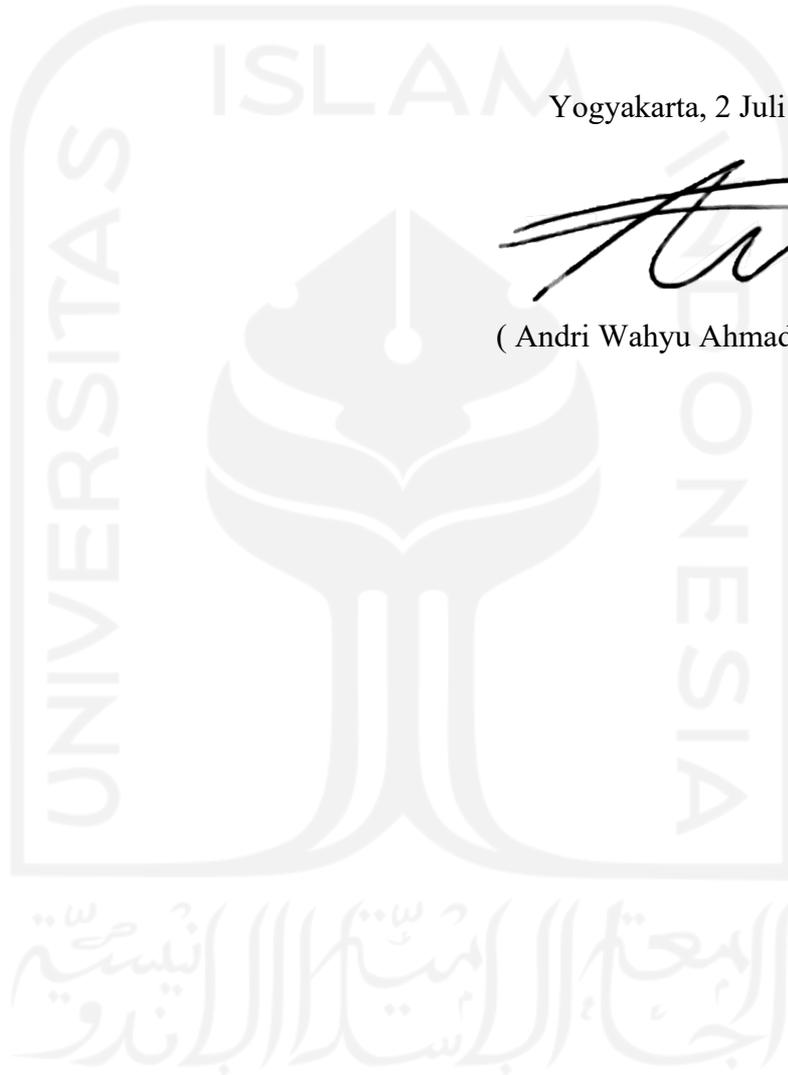
7. Seluruh teman-teman seperjuangan penjaluran magang yang tidak bisa disebutkan satu per satu.

Tentunya dalam penulisan laporan ini terdapat banyak kekurangan sehingga kritik dan masukan sangat diperlukan agar laporan ini bisa jadi lebih baik. Semoga laporan ini bisa bermanfaat dan dapat digunakan sebaik-baiknya bagi pembaca.

Yogyakarta, 2 Juli 2021



(Andri Wahyu Ahmad Ruslam)



SARI

Jala merupakan sebuah perusahaan yang bergerak di bidang akuakultur, khususnya dalam pembudidayaan udang. Jala memiliki visi untuk memberdayakan para petambak dengan menyediakan produk teknologi yang dapat membantu kegiatan kulturisasi. Terdapat dua jenis produk yang disediakan oleh Jala, yaitu *hardware* dan *software*. Banyaknya produk yang ditawarkan Jala yang sesuai dengan kebutuhan para petambak, mendorong tingginya angka pengguna Jala. Tentu saja ini sangat menguntungkan bagi perusahaan. Akan tetapi, setelah dilakukan riset internal aktivitas penggunaan *software*, ditemukan bahwa jumlah pengguna yang terdaftar di sistem dibandingkan dengan jumlah pengguna aktif berbeda sangat signifikan. Hal ini menunjukkan terdapat banyak pengguna yang *churn* dan tidak menggunakan produk *software*.

Inisiasi proyek Analisis Keberlanjutan Pengguna menggunakan *Factor Analysis* bertujuan mencari tahu faktor apa saja yang mempengaruhi pengguna untuk lanjut atau tidak menggunakan Jala sejak satu bulan pertama menggunakannya. Dalam menerapkan *Factor Analysis*, terdapat beberapa langkah yang harus dilakukan, yaitu ekstraksi data, *preprocessing*, *clustering* menggunakan *K-Means*, dan *modeling* menggunakan algoritma *Random Forest*. Dari model yang telah dihasilkan, kemudian diekstraksi nilai *feature importances* atau nilai signifikansi dari setiap variabel. Nilai tersebut yang dijadikan dasar untuk melakukan *Factor Analysis* dan menentukan variabel apa yang paling berpengaruh terhadap keberlanjutan pengguna ke bulan-bulan berikutnya.

Dari hasil yang didapatkan bahwa fitur pencatatan pakan, *sampling*, dan kualitas air merupakan fitur yang paling berpengaruh sehingga tingginya intensitas penggunaan di satu bulan pertama akan meningkatkan persentase keberlanjutan pengguna. Jika dibandingkan dengan fitur media yang dibuat khusus untuk mengedukasi pengguna dan membuatnya tetap menggunakan Jala, ternyata belum berpengaruh signifikan terhadap keberlanjutan pengguna. Dengan demikian diperlukan optimasi fitur ini agar dapat berperan sesuai dengan peruntukannya.

Kata kunci: *Factor Analysis*, ekstraksi data, *preprocessing*, *clustering*, *Random Forest*, *K-Means*

GLOSARIUM

| | |
|----------------------|---|
| <i>IoT</i> | IoT (<i>Internet of Things</i>) adalah teknologi yang menghubungkan informasi dari suatu alat, berupa hasil pengukuran maupun perhitungan, dengan sistem lainnya melalui media internet. |
| Siklus | aktivitas budi daya, dimulai dari penebaran benih dan diakhiri dengan panen, yang dilakukan dalam periode waktu tertentu. |
| Sampling | kegiatan pengambilan sampel udang untuk dilakukan pengukuran terhadap pertumbuhan panjang dan beratnya dalam setiap periode waktu tertentu, biasanya seminggu sekali selama proses budi daya. |
| <i>Preprocessing</i> | tahapan pengolahan data menjadi format tertentu dengan tujuan agar data dapat terstandarisasi dan memiliki struktur yang konsisten. |
| <i>Pandas</i> | <i>library python</i> yang memiliki kemampuan pengolahan data dalam bentuk tabel (baris-kolom) dan kalkulasi statistik. |
| <i>Clustering</i> | suatu proses pengelompokan data berdasarkan kriteria tertentu baik yang terlihat secara eksplisit maupun implisit |
| <i>K-Means</i> | algoritma yang mengelompokkan data numerik, bersifat <i>unsupervised</i> , dan terus melakukan perulangan hingga mencapai kondisi tertentu. |
| <i>Modeling</i> | pembuatan model yang dihasilkan dari penerapan algoritma <i>Machine Learning</i> yang dapat digunakan untuk melakukan prediksi pada data baru. |
| <i>Random Forest</i> | salah satu algoritma yang dapat digunakan untuk menyelesaikan dua kasus sekaligus, baik regresi maupun klasifikasi. |
| <i>PCA</i> | <i>PCA (Principal Component Analysis)</i> adalah teknik menganalisis multivariabel (<i>multivariate technique</i>) yang bertujuan untuk memperkecil dimensi dari suatu data. |
| <i>Elbow Method</i> | metode yang berfungsi membantu menentukan jumlah K yang optimal untuk digunakan pada algoritma <i>K-Means</i> sesuai dengan data yang dimiliki. |

DAFTAR ISI

| | |
|---|----|
| HALAMAN JUDUL | 1 |
| HALAMAN PENGESAHAN DOSEN PEMBIMBING | 2 |
| HALAMAN PENGESAHAN DOSEN PENGUJI | 3 |
| HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR | 4 |
| HALAMAN PERSEMBAHAN | 5 |
| HALAMAN MOTO | 6 |
| KATA PENGANTAR | 7 |
| SARI | 9 |
| GLOSARIUM | 10 |
| DAFTAR ISI | 11 |
| DAFTAR TABEL | 13 |
| DAFTAR GAMBAR | 14 |
| BAB I PENDAHULUAN | 16 |
| 1.1 Latar Belakang | 16 |
| 1.2 Ruang Lingkup Magang | 19 |
| 1.3 Tujuan | 20 |
| 1.4 Manfaat | 20 |
| 1.5 Sistematika Penulisan | 21 |
| BAB II DASAR TEORI | 22 |
| 2.1 Aplikasi Budi Daya | 22 |
| 2.2 Ekstraksi Data | 26 |
| 2.3 <i>Preprocessing</i> | 27 |
| 2.4 <i>Clustering</i> | 28 |
| 2.5 <i>Random Forest</i> | 30 |
| 2.6 <i>Factor Analysis</i> | 31 |
| BAB III PELAKSANAAN MAGANG | 32 |
| 3.1 <i>Data Science Roadmap</i> | 32 |
| 3.1.1 <i>Planning</i> | 32 |
| 3.1.2 <i>Execution</i> | 34 |
| 3.1.3 <i>Review With Dev</i> | 35 |

| | | |
|--|--|----|
| 3.1.4 | <i>Deployment/Presentation</i> | 36 |
| 3.2 | Implementasi Analisis Keberlanjutan Pengguna | 37 |
| 3.2.1 | Kriteria data | 37 |
| 3.2.2 | Ekstraksi Data | 39 |
| 3.2.3 | <i>Preprocessing</i> | 44 |
| 3.2.4 | <i>Clustering</i> | 49 |
| 3.2.5 | <i>Modeling</i> | 51 |
| BAB IV REFLEKSI PELAKSANAAN MAGANG | | 55 |
| 4.1 | Teknis | 55 |
| 4.1.1 | Analisis Fitur Berpengaruh | 55 |
| 4.1.2 | Analisis Fitur Media | 56 |
| 4.1.3 | Pengalaman Melakukan <i>Preprocessing</i> | 57 |
| 4.1.4 | Pengalaman Menggunakan Algoritma K-Means | 58 |
| 4.2 | Non Teknis | 59 |
| 4.2.1 | Manfaat Magang | 59 |
| 4.2.2 | Hambatan dan Tantangan Magang | 60 |
| 4.2.3 | Kontribusi Selama Magang | 60 |
| BAB V KESIMPULAN DAN SARAN | | 61 |
| 5.1 | Kesimpulan | 61 |
| 5.2 | Saran | 61 |
| DAFTAR PUSTAKA | | 62 |
| LAMPIRAN | | 65 |

DAFTAR TABEL

| | |
|---|----|
| Tabel 3.1 Rangkuman <i>timeline</i> dari PRD | 34 |
| Tabel 3.2 Variabel dari fitur-fitur <i>website</i> | 37 |
| Tabel 3.3 Variabel hasil interaksi pengguna dengan <i>website</i> | 38 |
| Tabel 3.4 Kriteria data yang digunakan..... | 39 |



DAFTAR GAMBAR

| | |
|--|----|
| Gambar 1.1 Data tambak dengan siklus aktif | 18 |
| Gambar 2.1 Beranda aplikasi budi daya | 22 |
| Gambar 2.2 Fitur pencatatan pakan | 23 |
| Gambar 2.3 Fitur pencatatan sampling | 23 |
| Gambar 2.4 Fitur pencatatan kualitas air | 24 |
| Gambar 2.5 Sub fitur kabar udang | 25 |
| Gambar 2.6 Sub fitur harga udang | 25 |
| Gambar 2.7 Sub fitur penyakit udang | 25 |
| Gambar 2.8 Proses pengolahan data menjadi informasi | 26 |
| Gambar 2.9 Tahapan dalam <i>preprocessing</i> | 27 |
| Gambar 2.10 <i>Flowchart</i> algoritma <i>K-Means</i> | 29 |
| Gambar 3.1 <i>Engineering Roadmap Data Scientist</i> Jala | 32 |
| Gambar 3.2 Notion untuk penulisan PRD | 33 |
| Gambar 3.3 Alur pengerjaan proyek | 35 |
| Gambar 3.4 Proses review bersama <i>lead of data scientist</i> | 36 |
| Gambar 3.5 Proses presentasi kepada stakeholder | 36 |
| Gambar 3.6 Penerapan <i>with clause</i> | 40 |
| Gambar 3.7 Penerapan <i>join clause</i> | 40 |
| Gambar 3.8 Cuplikan <i>query</i> seluruh variabel | 40 |
| Gambar 3.9 Hasil <i>query</i> deskripsi siklus | 41 |
| Gambar 3.10 Cuplikan <i>query</i> deskripsi tambak | 42 |
| Gambar 3.11 Hasil <i>query</i> deskripsi tambak | 42 |

| | |
|---|----|
| Gambar 3.12 Cuplikan <i>query</i> keberlanjutan pengguna..... | 43 |
| Gambar 3.13 Hasil <i>query</i> keberlanjutan pengguna..... | 43 |
| Gambar 3.14 Tahapan <i>preprocessing</i> yang dilakukan..... | 44 |
| Gambar 3.15 Relasi tabel siklus, kolom, dan tambak..... | 45 |
| Gambar 3.16 Agregasi data berdasarkan kolom..... | 45 |
| Gambar 3.17 Cuplikan <i>method</i> mengisi <i>missing value</i> | 46 |
| Gambar 3.18 Cuplikan pemisahan data numerik dan kategorik..... | 46 |
| Gambar 3.19 Cuplikan <i>method</i> penyaringan <i>quantile</i> | 47 |
| Gambar 3.20 Hasil penerapan <i>quantile</i> | 47 |
| Gambar 3.21 Hasil standarisasi data..... | 48 |
| Gambar 3.22 Ekstraksi nilai <i>variance</i> dari setiap variabel..... | 49 |
| Gambar 3.23 Visualisasi <i>elbow method</i> | 50 |
| Gambar 3.24 Cuplikan penerapan <i>K-Means</i> | 50 |
| Gambar 3.25 Hasil <i>clustering</i> | 51 |
| Gambar 3.26 Alur <i>Modeling</i> | 51 |
| Gambar 3.27 Hasil pembagian data..... | 52 |
| Gambar 3.28 Inisialisasi <i>hyperparameter</i> | 53 |
| Gambar 3.29 <i>Hyperparameter</i> paling optimum..... | 53 |
| Gambar 3.30 Cuplikan <i>method</i> menyimpan dan memuat model..... | 53 |
| Gambar 3.31 Ekstraksi nilai signifikansi variabel..... | 54 |
| Gambar 4.1 Analisis fitur yang paling berpengaruh..... | 55 |
| Gambar 4.2 Analisis fitur media..... | 57 |
| Gambar 4.3 Persentase durasi waktu yang digunakan <i>data scientist</i> | 58 |

BAB I PENDAHULUAN

1.1 Latar Belakang

Sebagai negara maritim dengan gelar negara kepulauan terbesar di dunia, Indonesia memiliki beragam kekayaan sumber daya alam. Dari kekayaan hutan hingga lautan ada di Indonesia. Oleh karena itu, Indonesia dikelompokkan sebagai salah satu negara yang memiliki kekayaan alam yang sangat luar biasa atau yang diistilahkan sebagai *mega-biodiversity*. Memiliki wilayah laut seluas 5,8 juta km² dan panjang garis pantai mencapai 81.000 km, menjadikan wilayah laut Indonesia sebagai tempat yang nyaman untuk tinggal bagi beragam hayati laut (Arafat, 2018). Kelebihan-kelebihan inilah yang menjadikan pengelolaan sumber daya alam Indonesia, khususnya sumber daya lautan, menjadi salah satu sumber penghasilan yang menguntungkan bagi pebisnis dan negara. Hal ini dibuktikan dari pendapatan hasil ekspor sumber daya lautan Indonesia pada tahun 2013, mencapai angka US\$ 4,2 milyar. Fakta ini menjadikan Indonesia sebagai 10 negara pengekspor hasil laut terbesar (Nasution, 2017).

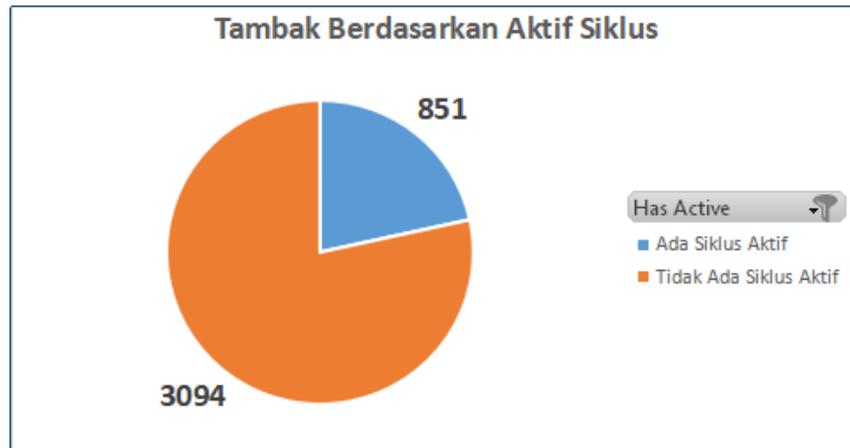
Salah satu komoditas yang menarik mata dunia adalah udang. Dari data yang pernah tercatat, ekspor komoditas udang dapat meraup pendapatan sebesar US\$ 1.280 juta (Pulungan et al., 2015). Hal ini menjadikan budi daya udang menjadi peluang yang menguntungkan sehingga banyak orang mencoba membuka tambak dan memulai bisnis ini. Akan tetapi, ternyata dalam menjalani budi daya udang terdapat beberapa masalah yang dihadapi para petambak. Salah satunya adalah perubahan kualitas air yang dapat terjadi secara signifikan dengan waktu yang sangat cepat. Perubahan kualitas air yang tidak terkontrol dapat menyebabkan rendahnya tingkat kelulushidupan udang dan mendorong terjadinya gagal panen. Selain itu, proses budi daya yang tidak terkontrol, seperti pemberian pakan yang berlebihan dapat menyebabkan tingginya biaya produksi yang berpengaruh terhadap angka kerugian.

Berdasarkan permasalahan-permasalahan tersebut, Jala hadir sebagai sebuah perusahaan yang bergerak di bidang akuakultur, khususnya dalam pembudidayaan udang. Visi dari Jala adalah membantu petambak dalam mengelola tambaknya dengan menghadirkan produk-produk berbasis teknologi. Tujuan dari produk-produk ini adalah untuk mengefisienkan tambak serta membantu mengontrol tambak secara otomatis dan *real-time*. Terdapat dua jenis produk Jala, yaitu produk berbasis *hardware* dan *software*.

Baruno adalah salah satu produk *hardware* yang ditawarkan oleh Jala kepada penggunanya. Baruno dibuat dengan menggunakan komponen-komponen elektronik tertentu yang memungkinkannya untuk mengukur kualitas air, seperti salinitas, pH, temperatur, dan *Dissolved Oxygen* (DO) -banyaknya oksigen yang terkandung di air-. Dengan menerapkan teknologi *IoT* (*Internet of Things*), Baruno dapat menyediakan informasi kualitas air kepada pengguna secara *real-time* yang dapat membantu untuk mengontrol stabilitas kualitas air di tambak. *IoT* adalah teknologi yang menghubungkan informasi dari suatu alat, berupa hasil pengukuran maupun perhitungan, dengan sistem lainnya melalui media internet (Hussein, 2019). Dengan demikian, Baruno dapat menjadi solusi petambak agar dapat mengontrol kualitas air tambaknya sehingga dapat meningkatkan hasil produksi udang.

Selain baruno, Jala menghadirkan produk berbentuk *software*, yaitu aplikasi budi daya yang tersedia dalam bentuk *website* dan *mobile*. Aplikasi budi daya bertujuan untuk membantu petambak dalam mengelola dan mencatat seluruh aktivitas dalam budi daya. Aktivitas budi daya yang dicatat oleh *website* ini meliputi deskripsi tambak, jumlah kolam, pembuatan siklus, pencatatan pakan, sampling, kualitas air, dan panen. Berdasarkan data-data tersebut, aplikasi ini juga mampu memprediksi tingkat kelulushidupan udang sehingga dapat membantu pengguna untuk mengoptimalkan tambaknya. Aplikasi juga dilengkapi dengan fitur media yang berisi informasi seputar tips & trik budi daya, harga penjualan udang, dan penyakit dapat dialami oleh udang. Salah satu tujuan dari fitur media adalah mengedukasi pengguna agar dapat mengetahui informasi seputar budi daya dan dapat terus menggunakan Jala dalam jangka panjang.

Beragamnya produk yang ditawarkan Jala, berdampak pada tingginya angka pengguna yang menggunakan Jala. Untuk produk *software*, tercatat 9000 pengguna yang terdaftar pada sistem Jala yang tersebar di enam negara. Tingginya jumlah pengguna ini tentu sangat membanggakan bagi perusahaan. Akan tetapi, ketika dilihat lebih detail terdapat fakta yang cukup mengejutkan. Ternyata jumlah pengguna yang terdaftar, tidak berbanding lurus dengan jumlah siklus aktif yang dicatatkan pengguna. Siklus aktif adalah aktivitas budi daya, dimulai dari penebaran benih dan diakhiri dengan panen, yang dilakukan dalam periode waktu tertentu.



Gambar 1.1 Data tambak dengan siklus aktif

Hal ini mengindikasikan bahwa terdapat pengguna yang hanya mendaftar ke sistem Jala, menggunakannya dalam beberapa periode waktu tertentu, lalu berhenti. Berdasarkan data internal perusahaan tahun 2018-2020 seperti yang terlihat pada Gambar 1.1, terdapat 3945 tambak yang terdaftar menggunakan Jala. Dari angka tersebut, hanya 851 tambak yang masih mencatatkan siklus aktifnya ke Jala. Artinya, hanya sekitar 22% pengguna yang masih menggunakan Jala dan 78% sisanya telah *churn*. *Churn* adalah kondisi pengguna yang sudah tidak berlangganan produk dari suatu perusahaan dan berpindah ke kompetitor (Govindaraju et al., 2008).

Tinggi rendahnya angka *churn* sangat berpengaruh terhadap performa perusahaan. Apabila angka *churn* pengguna terus meningkat, dapat menyebabkan kerugian bagi perusahaan disebabkan biaya yang banyak telah dikeluarkan untuk mempertahankan pengguna (Govindaraju et al., 2008). Dengan demikian, penting untuk mengetahui faktor yang dapat mempengaruhi keberlanjutan pengguna dalam jangka panjang. Mengetahui hal ini dapat membantu perusahaan dalam mengoptimisasi produk serta memperbaiki teknik *marketing* agar pengguna tetap loyal dengan produk yang ditawarkan. Oleh karena itu, diinisiasi sebuah proyek yang diberi nama Analisis Keberlanjutan Pengguna menggunakan teknik *Factor Analysis*.

Factor Analysis adalah teknik yang digunakan untuk melihat pengaruh antar setiap variabel terhadap objek tertentu -dalam hal ini, yaitu keberlanjutan pengguna- (Amruthnath & Gupta, 2019). Terdapat beberapa teknik yang digunakan untuk dapat melakukan *Factor Analysis*, yaitu ekstraksi data, *preprocessing*, *clustering* menggunakan algoritma *K-Means*, dan *modeling* menggunakan *Random Forest*. Hasil dari *modeling* tersebut akan menjadi dasar untuk mengukur nilai signifikansi dari setiap variabel. Nilai inilah yang akan digunakan sebagai dasar dalam *Factor Analysis*.

1.2 Ruang Lingkup Magang

Program magang berlangsung selama enam bulan dengan *role* sebagai seorang *data scientist* di perusahaan Jala. Selama magang sebagai *data scientist*, terdapat tiga proyek yang dilaksanakan, yaitu sebagai berikut.

a. Analisis Efisiensi Tambak

Analisis efisiensi tambak bertujuan untuk memetakan tingkat efisiensi dari setiap tambak yang menggunakan Jala berdasarkan data yang dimasukkan oleh pengguna ke sistem. Dalam proyek ini data yang digunakan adalah seluruh data budi daya pengguna dengan kelengkapan pencatatan pakan di atas 85% sebagai indikator bahwa tambak tersebut secara aktif menggunakan Jala. Hasil akhir dari proyek ini berbentuk dashboard *business intelligence* yang dapat digunakan oleh stakeholder Jala -pimpinan, *marketing*, bisnis manajer, dan seluruh karyawan Jala- untuk menentukan strategi ke depan.

b. Analisis Akurasi Model Prediksi

Dalam proyek ini, fokus pengerjaan adalah menganalisis ketepatan model prediksi yang dimiliki oleh Jala. Analisis dilakukan dengan menghitung MAPE (*Mean Absolute Percentage Error*) dari setiap variabel yang diprediksi. Hasil akhirnya berupa dashboard *business Intelligence* yang memetakan hasil perhitungan MAPE dari setiap variabel. Dashboard ini kemudian dijadikan sebagai salah satu KPI perusahaan. KPI atau *Key Performance Indicator* merupakan objek dari suatu produk yang dimiliki oleh perusahaan yang dapat diukur untuk menilai kualitas dari perusahaan tersebut sehingga dapat dibandingkan dengan perusahaan lain (Ishaq Bhatti, Awan, & Razaq, 2013).

c. Analisis Keberlanjutan Pengguna

Tujuan dari proyek ini adalah mencari tahu faktor-faktor yang mempengaruhi keberlanjutan pengguna sejak satu bulan pertama menggunakan Jala. Pengambilan satu bulan sebagai parameter keberlanjutan dikarenakan durasi budi daya dalam satu siklus rata-rata minimal 30 hari. Analisis dilakukan dengan menggunakan seluruh data budi daya yang tercatat di dalam sistem dan data interaksi pengguna dengan fitur-fitur yang ada di *website*. Proses analisis akan difokuskan dengan membandingkan antara fitur-fitur pencatatan budi daya dan fitur media. Selain itu, dalam prosesnya, proyek ini menggunakan pengelompokan data yang bertujuan sebagai pembandingan hasil antara data tanpa pengelompokan dan dengan pengelompokan.

Dalam penulisan tugas akhir ini, fokus penjelasan adalah proyek pada poin c, yaitu Analisis Keberlanjutan Pengguna menggunakan *Factor Analysis*. Pengambilan proyek ini sebagai laporan tugas akhir karena pengetahuan yang didapatkan dari hasil pengerjaan dapat membantu perusahaan dalam mengetahui faktor keberlanjutan pengguna. Selain itu, proyek ini sangat berpeluang besar untuk dilanjutkan sehingga dokumentasi dalam bentuk laporan akan dapat membantu pengerjaan proyek sejenis ke depannya. Penjelasan detail akan dijabarkan pada bab-bab selanjutnya.

1.3 Tujuan

Tujuan dari penerapan teknik *Factor Analysis* untuk menganalisis keberlanjutan pengguna antara lain, yaitu:

- a. Menganalisis faktor-faktor yang mempengaruhi keberlanjutan pengguna berdasarkan fitur *website* yang tersedia.
- b. Melihat performa fitur media terhadap keberlanjutan pengguna. Performa tersebut dilihat dengan menganalisis tiga sub fitur dari fitur media, yaitu sub fitur kabar udang, sub fitur harga udang, dan sub fitur penyakit udang.
- c. Membandingkan hasil *Factor Analysis* antara data yang dikelompokkan dengan tanpa pengelompokkan.
- d. Sebagai langkah inisialisasi proyek-proyek *Factor Analysis* lainnya dengan variabel yang berbeda-beda

1.4 Manfaat

Terdapat beberapa manfaat yang didapatkan dari hasil implementasi proyek ini, yaitu:

- a. Mengetahui fitur-fitur yang harus dioptimalkan agar pengguna dapat menggunakan Jala dalam jangka panjang.
- b. Hasil dari penerapan proyek ini, dapat dijadikan referensi untuk melakukan strategi *marketing* ke depan.
- c. Mendapatkan pengetahuan mengenai signifikansi penggunaan pengelompokkan data dalam kasus *Factor Analysis*.
- d. Mendapatkan ide-ide segar untuk mengerjakan proyek lain yang sejenis, seperti analisis keberlanjutan pengguna berdasarkan perbandingan antara efektifitas aplikasi *website* dan *mobile*.

1.5 Sistematika Penulisan

Sistematika Penulisan adalah rangkaian urutan penulisan dari suatu dokumen. Adapun penulisan tugas akhir ini disusun dengan sistematika penulisan sebagai berikut:

- a. BAB I berfokus pada pendahuluan yang meliputi latar belakang, ruang lingkup, tujuan, dan manfaat dari Analisis Keberlanjutan Pengguna
- b. BAB II akan membahas mengenai detail dari teori-teori yang berkaitan dengan proyek yang dikerjakan.
- c. BAB III akan menjelaskan pelaksanaan magang, mulai dari tahapan persiapan hingga implementasi dari teori-teori yang telah dituliskan di bab sebelumnya.
- d. BAB IV berfokus pada refleksi selama magang yang dilihat dari dua sudut pandang, yaitu sudut pandang teknis dan non teknis.
- e. BAB V akan membahas mengenai kesimpulan yang didapatkan dari hasil pengerjaan proyek. Selain itu bab ini juga akan berisi saran dari penulis yang dirasakan selama magang sehingga dapat dijadikan pembelajaran bagi pembaca.

BAB II

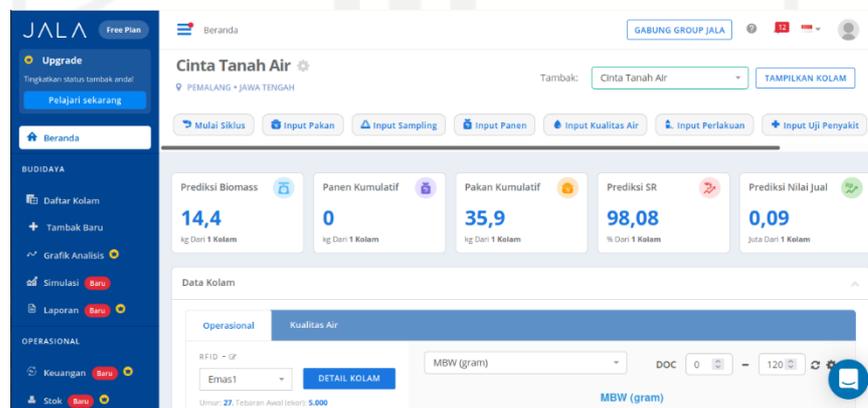
DASAR TEORI

2.1 Aplikasi Website

Aplikasi website adalah sebuah aplikasi yang berjalan pada web server baik bersifat lokal maupun daring yang dapat diakses perangkat lain menggunakan web browser seperti Chrome, Firefox, dan Safari. Aplikasi website berisi halaman-halaman berbasis HTML (*Hyper Text Markup Language*) yang dibungkus dalam domain atau subdomain sehingga ketika mengakses domain tersebut, halaman-halaman aplikasi website dapat ditampilkan dan digunakan (Trimarsiah & Arafat, 2017).

2.1.1 Aplikasi Budi Daya

Aplikasi budi daya merupakan aplikasi yang membantu petambak dalam mencatat seluruh aktivitas yang berkaitan dengan budi daya, mulai dari pencatatan pakan, sampling, kualitas air, panen, hingga laporan keuangan. Saat ini aplikasi ini tersedia dalam dua basis utama, yaitu berbasis *website* dan *android*. Fokus pada pembahasan laporan akhir ini adalah aplikasi budidaya berbasis *website*.



Gambar 2.1 Beranda aplikasi budi daya

Gambar 2.1 menunjukkan halaman beranda dari aplikasi budi daya. Terdapat banyak fitur yang ditawarkan oleh Jala, beberapa di antaranya adalah fitur pencatatan pakan, sampling, kualitas air, dan panen.



Gambar 2.2 Fitur pencatatan pakan

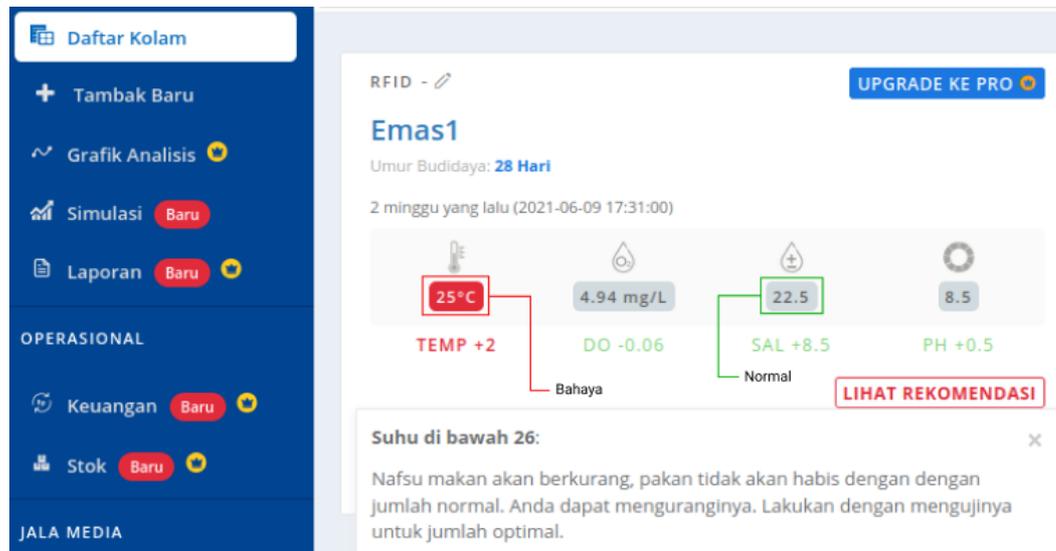
Fitur pakan ditujukan untuk membantu pengguna dalam mengelola pemberian pakan. Pada fitur ini, pengguna dapat memasukkan jumlah pakan yang diberikan setiap harinya. Selain itu, terdapat informasi grafik mengenai catatan pemberian pakan yang telah diberikan seperti yang terdapat pada Gambar 2.2. Harapannya dengan menggunakan fitur ini para petambak akan terbantu dalam mengelola pemberian pakan sehingga pakan yang diberikan ideal sesuai dengan kebutuhan udang. Apabila pakan diberikan berlebihan dapat mengubah kualitas air sehingga air dapat menjadi beracun. Sementara jika pakan yang diberikan kurang, dapat menghambat tumbuh kembang udang.



Gambar 2.3 Fitur pencatatan sampling

Sampling adalah kegiatan pengambilan sampel udang untuk dilakukan pengukuran terhadap pertumbuhan panjang dan beratnya dalam setiap periode waktu tertentu, biasanya seminggu sekali selama proses budi daya (Witoko et al., 2018). Sampling bertujuan untuk mengevaluasi pertumbuhan udang agar dapat dikontrol dan diketahui kondisinya. Oleh karena itu, selain mengukur panjang dan berat udang, pada saat sampling juga dilakukan pengecekan terhadap

kesehatan udang untuk mengidentifikasi penyakit udang secara dini. Jala menyediakan fitur pencatatan sampling yang berfungsi untuk membantu petambak dalam melakukan pencatatan terhadap hasil sampling. Selain itu, fitur ini juga dapat membantu memprediksi nilai MBW (*Mean Body Average*) -rata-rata berat udang dalam gram- dan ADG (*Average Daily Growth*) - pertumbuhan harian udang dalam gram-. Gambar 2.3 menunjukkan fitur sampling.

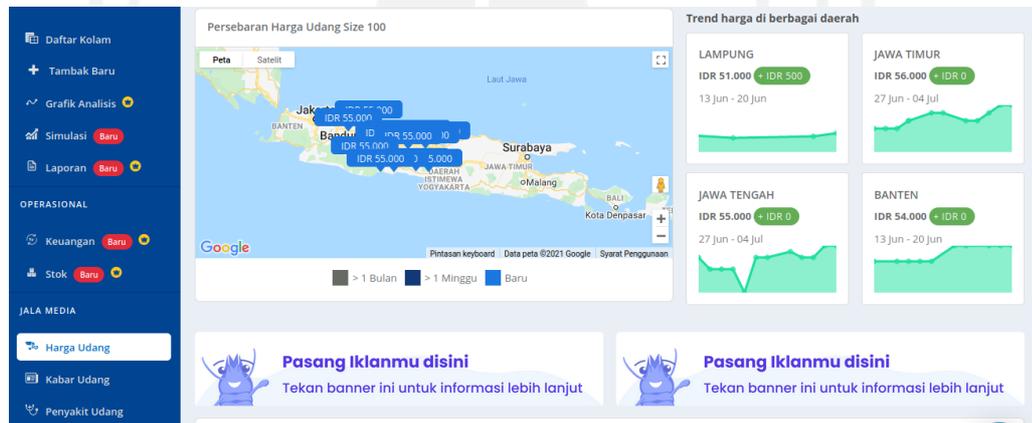


Gambar 2.4 Fitur pencatatan kualitas air

Fitur pencatatan kualitas air merupakan salah satu fitur yang sangat membantu pengguna dalam mengontrol kondisi air pada tambak. Bagusnya kualitas air sangat berpengaruh terhadap kualitas udang. Selain membantu mencatat parameter utama kualitas air, meliputi pH, temperatur, salinitas, dan oksigen terlarut (DO atau *Dissolved Oxygen*), fitur ini dapat mendeteksi dan memberikan saran mengenai kualitas air. Proses pencatatan kualitas air dapat dilakukan dengan dua cara, yaitu dengan memasukkan secara manual dari hasil sampling air atau dengan memanfaatkan penyimpanan awan hasil dari pembacaan Baruno. Gambar 2.4 menunjukkan fitur kualitas air.



Gambar 2.5 Sub fitur kabar udang



Gambar 2.6 Sub fitur harga udang

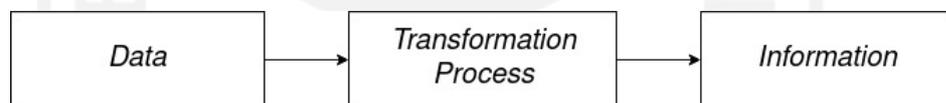


Gambar 2.7 Sub fitur penyakit udang

Fitur media adalah salah satu fitur yang ditawarkan Jala yang bertujuan untuk mendukung pengguna seputar Jala dan seputar budi daya udang. Dari fitur ini terdapat tiga sub fitur, yaitu kabar udang, harga udang, dan penyakit udang. Sub fitur kabar udang merupakan fitur yang menyediakan informasi seputar tips dan trik untuk merawat kolam serta pembudidayaan udang. Gambar 2.5 menunjukkan sub fitur kabar udang. Sub fitur harga udang merupakan fitur yang memberikan informasi kepada pengguna berkaitan dengan harga penjualan udang di wilayah-wilayah Indonesia. Gambar 2.6 menunjukkan sub fitur harga udang. Sementara sub fitur penyakit udang merupakan fitur yang menyediakan informasi seputar penyakit udang, mulai dari cara identifikasi hingga melakukan penanganan terhadap penyakit tersebut. Gambar 2.7 menunjukkan sub fitur penyakit udang. Harapannya dengan fitur-fitur media ini pengguna dapat teredukasi menggunakan Jala dan tertarik untuk terus menggunakannya dalam jangka panjang.

2.2 Ekstraksi Data

Ekstraksi merupakan sebuah proses pengambilan data dari sumber penyimpanan / *database* dengan tujuan untuk menyediakan data agar dapat diolah. Data perlu diolah menjadi informasi agar dapat mudah dipahami oleh para stakeholder -pimpinan, *marketing*, bisnis manajer, dan seluruh karyawan Jala- yang membutuhkan informasi tersebut.



Gambar 2.8 Proses pengolahan data menjadi informasi

Sumber: (Satoto et al., 2017)

Gambar 2.8 menunjukkan proses perubahan dari data menjadi informasi. Ekstraksi data merupakan tahapan awal dalam *Transformation Process*. Detailnya proses ekstraksi data yang dilakukan akan sangat membantu mempermudah proses-proses selanjutnya dalam mengubah data menjadi informasi.

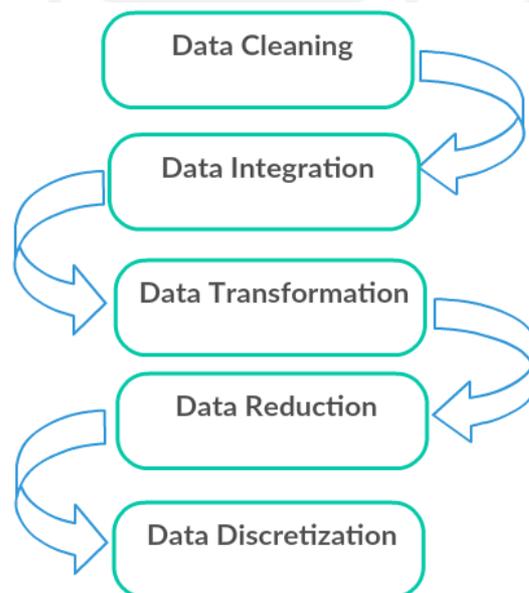
Pada *software*, data disimpan dan dikelola menggunakan *database*. *Database* adalah koleksi data yang disimpan dengan struktur tertentu pada komputer sehingga dapat diakses dan digunakan oleh perangkat komputer tersebut. Untuk mengelola data yang berada di *database* diperlukan sebuah kakas yang sering disebut dengan DBMS (*Database Management System*) (Satoto et al., 2017). MySQL merupakan salah satu DBMS dengan konsep *relational database* yang paling populer yang banyak digunakan oleh *software engineer* dunia. Hal ini karena MySQL bersifat

open-source dan menggunakan konsep relational (tabel-kolom) sehingga mudah untuk dipahami dan dikembangkan (Satoto et al., 2017).

MySQL mengelola data dengan menyimpannya dalam baris dan kolom yang memiliki berbagai macam tipe data. Tipe data yang dapat dikelola oleh MySQL meliputi tipe data angka, tanggal dan jam, serta karakter. Setiap kali pembuatan *database* baru diperlukan inisialisasi tipe data dari setiap kolom sehingga pada saat memasukkan data ke kolom tertentu harus menyesuaikan dengan tipe datanya masing-masing.

2.3 Preprocessing

Preprocessing adalah tahapan pengolahan data menjadi format tertentu dengan tujuan agar data dapat terstandarisasi dan memiliki struktur yang konsisten. Teknik ini perlu diterapkan karena pada dasarnya data yang kita miliki tidak terstruktur, banyak duplikasi, dan inkonsisten (Agarwal, 2015).



Gambar 2.9 Tahapan dalam *preprocessing*

Sumber: (Agarwal, 2015)

Dalam *data science* proses *preprocessing* meliputi beberapa tahapan. Gambar 2.9 menunjukkan tahapan-tahapan dalam *preprocessing*.

- a. *Data cleaning* merupakan tahapan pembersihan dengan mengidentifikasi data yang rusak atau tidak sesuai dengan data yang berada dari sumber tabel. Pada tahapan ini fokus

dari pengelolaan data adalah mengidentifikasi data yang tidak konsisten, data yang rusak, dan tidak akurat lalu melakukan modifikasi pada data tersebut atau menghapusnya.

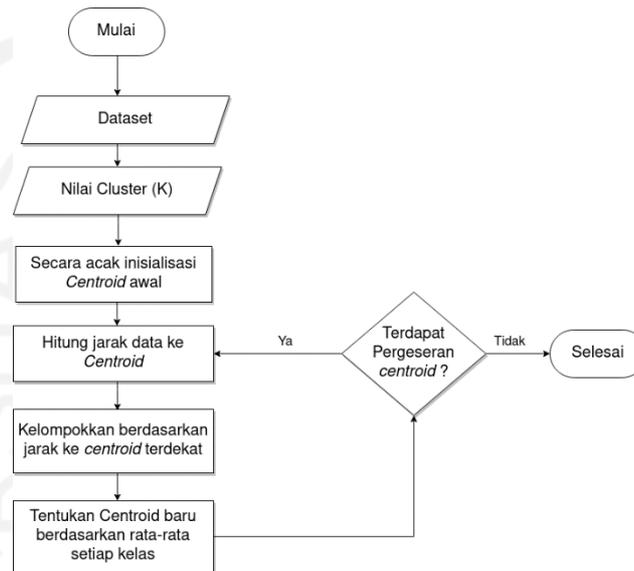
- b. *Data Integration* adalah tahapan penggabungan data dari berbagai sumber menjadi satu kesatuan. Dalam prosesnya, penggabungan ini akan memunculkan *error* karena data yang tersedia pasti memiliki struktur yang berbeda-beda. Dengan demikian, ketelitian dalam mencari kesamaan dari setiap data yang tersedia lalu menggabungkannya merupakan kemampuan yang sangat diperlukan dalam proses ini.
- c. *Data Transformation* merupakan proses perubahan data dari format yang sulit dipahami menjadi format yang mudah untuk dimengerti. Pada tahap ini dilakukan normalisasi, agregasi, dan generalisasi terhadap data yang dimiliki. Normalisasi meliputi pengolaan data yang redundan sehingga dapat memperkecil ukuran dari data yang dimiliki. Agregasi berfungsi untuk menggabungkan data berdasarkan variabel tertentu sehingga penyerapan informasi dapat dilakukan dengan lebih mudah. Generalisasi adalah proses untuk melakukan rangkuman terhadap data yang dimiliki untuk mengevaluasi kualitas dari data yang diolah.
- d. *Data Reduction* merupakan tahapan pengurangan dimensi yang bertujuan untuk mempermudah komputasi dari pengelolaan data sehingga proses pengelolaan data yang besar menjadi lebih cepat.
- e. *Data Discretization* adalah tahapan untuk mengubah data yang bersifat kontinu menjadi data diskret.

Salah satu *tools* untuk melakukan *preprocessing* pada *python* adalah *Pandas*. *Pandas* merupakan *library python* yang memiliki kemampuan pengolahan data dalam bentuk tabel (baris-kolom) dan kalkulasi statistik. *Pandas* dibuat untuk memenuhi kebutuhan akademis dan industri *data science* dalam *preprocessing* data menggunakan *python* (Agarwal, 2015). Sebagai *tools preprocessing*, *Pandas* memiliki berbagai kemampuan pengolahan data, seperti membentuk struktur dari sebuah data menjadi bentuk baris dan kolom. Salah satu data yang paling sering diolah adalah data dari API yang berbentuk JSON (Agarwal, 2015).

2.4 Clustering

Clustering dalam *data science* merupakan suatu proses pengelompokkan data berdasarkan kriteria tertentu baik yang terlihat secara eksplisit maupun implisit (Shi et al., 2010). Dalam proses *Clustering*, algoritma akan melakukan pencarian pola dari suatu data kemudian akan mengelompokkannya berdasarkan pola tersebut. Salah satu algoritma yang terkenal dalam

Clustering data adalah algoritma *K-Means*. *K-Means* adalah algoritma yang mengelompokkan data numerik, bersifat *unsupervised*, dan terus melakukan perulangan hingga mencapai kondisi tertentu. *K-Means* memiliki tingkat efektifitas yang tinggi dan telah terbukti dalam banyak implementasi dengan hasil klasifikasi yang memuaskan (Shi et al., 2010). Penggunaan algoritma *K-Means* dalam penelitian ini dikarenakan penerapannya yang mudah, banyak digunakan, dan secara umum telah terbukti efektifitasnya.



Gambar 2.10 *Flowchart* algoritma *K-Means*

Gambar 2.10 menunjukkan flowchart *K-Means*. Algoritma *K-Means* akan dieksekusi dalam dua fase. Pada fase pertama, *K-Means* akan mengambil sampel secara acak sejumlah nilai K yang ditentukan sebagai *center* dari masing-masing kelas. Pada fase selanjutnya, algoritma akan menghitung jarak masing-masing data relatif terhadap *center*. Data yang terdekat dengan *center* akan menjadi anggota kelas dari *center* tersebut. Berikut adalah penjelasan dari setiap tahapan algoritma *K-Means*.

- Input yang diperlukan pada algoritma ini adalah *dataset* dan jumlah kelas (K) yang diinginkan.
- Algoritma kemudian akan melakukan inisialisasi secara acak *centroid* sejumlah K yang dimasukkan.
- Kemudian, setiap data akan dihitung jaraknya relatif terhadap masing-masing *centroid* menggunakan formula *Euclidean Distance*. Rumus (2.1) adalah formula *Euclidean Distance*.

$$d(x_i, y_i) = \left[\sum_{i=1}^n (x_i - y_i)^2 \right]^{\frac{1}{2}} \quad (2.1)$$

x_i dan y_i merupakan objek x dan y ke- i . Sementara n adalah jumlah objek

- d. Data yang terdekat dengan salah satu *centroid* kelas akan menjadi anggota dari kelas tersebut.
- e. Setelah semua data telah dikelompokkan, kemudian akan diambil *centroid* baru yang dihitung dari rata-rata nilai setiap kelas.
- f. Apabila terdapat perubahan nilai *centroid*, algoritma akan mengulang tahap c hingga e sampai tidak terdapat perubahan nilai *centroid* lagi.

2.5 Random Forest

Salah satu algoritma yang dapat digunakan untuk dua kasus sekaligus, baik regresi maupun klasifikasi adalah algoritma *Random Forest* (Amruthnath & Gupta, 2019). Algoritma ini memanfaatkan *decision tree* untuk melakukan pembelajaran. *Decision tree* yang digunakan memiliki jumlah yang banyak. Hal ini karena penggunaan *single decision tree* memiliki akurasi yang rendah jika mendapatkan masukkan baru yang berbeda dari *dataset training*. Berikut adalah langkah-langkah saat menerapkan algoritma *Random Forest* (Amruthnath & Gupta, 2019).

- a. Melakukan *bootstrap* sebanyak n_{tree} . n_{tree} didapatkan berdasarkan masukkan yang diterima oleh algoritma. *Bootstrap* adalah proses *resampling dataset* dengan mengambil sejumlah baris dan kolom secara acak dengan metode *row sampling with replacement*. Artinya antara satu *bootstrap* dan *bootstrap* yang lain akan berbeda.
- b. Dari setiap *bootstrap* akan dikembangkan *decision tree*. Pada umumnya ketika membentuk *decision tree*, *node* diambil dari nilai variabel terbaik yang dihitung menggunakan metode tertentu, salah satunya (*Gini Impurity*) dengan mempertimbangkan seluruh variabel. Akan tetapi, pada *Random Forest node* diambil secara acak dengan mengambil sejumlah m_{txy} variabel, bukan secara keseluruhan.
- c. Prediksi data dengan melakukan agregasi dari setiap *decision tree* yang telah terbentuk (pada klasifikasi menggunakan *majority vote* sementara pada kasus regresi menggunakan *average*)

Akurasi dari *Random Forest* dihitung dengan menggunakan *Out-of-Bag* (OOB). OOB merupakan data yang ada di dalam *dataset* tetapi tidak terdapat dalam *bootstrap*. Dengan menguji

data tersebut ke setiap *decision tree* akan terlihat persentase data prediksi yang benar dengan mengakumulasi dari setiap *decision tree*. Sebaliknya, *error rate* diketahui dengan menghitung persentase prediksi yang salah dari agregasi seluruh *decision tree*.

Random Forest juga dapat melihat nilai signifikansi dari setiap variabel. Nilai signifikansi ini dapat dihitung dari seberapa besar *prediction error* meningkat ketika data pada variabel tersebut diacak sementara variabel yang lain tetap sama (Amruthnath & Gupta, 2019).

2.6 Factor Analysis

Factor Analysis adalah teknik yang digunakan untuk mempertimbangkan setiap variabel yang berpengaruh signifikan terhadap kelas tertentu. *Factor Analysis* banyak digunakan untuk menyelesaikan masalah-masalah dalam klasifikasi dan pengurangan dimensi dari suatu data (Amruthnath & Gupta, 2019). Pengurangan dimensi dari suatu data merupakan salah satu implementasi *factor analysis* yang banyak digunakan.

Konsep dalam pengurangan dimensi menggunakan *factor analysis* adalah dengan menentukan “*Factor*” dari seluruh variabel yang tersedia. “*Factor*” adalah suatu variabel yang dapat mewakili variabel lain dengan syarat memiliki karakteristik yang sama. Dengan mendapatkan “*factor*”, variabel lain yang telah terwakilkan oleh “*factor*” tersebut dapat dihilangkan. Tujuan dari pengurangan dimensi ini adalah memperkecil jumlah variabel yang akan diolah sehingga dapat meringankan komputasi pengelolaan data (Ghatak, 2021).

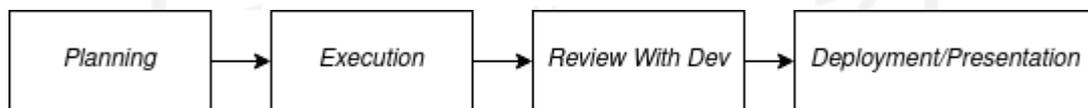
Dalam proses pengurangan dimensi terdapat satu persyaratan utama dalam menggunakan teknik *Factor Analysis*, yaitu data yang digunakan harus bersifat linear. Apabila variabel-variabel yang dimiliki tidak bersifat linear, penggunaan metode pengurangan dimensi dengan *factor analysis* tidak berpengaruh signifikan. Hal ini karena ketika data yang digunakan tidak linear, jumlah “*factor*” yang dihasilkan akan sama dengan total variabel yang digunakan (Ghatak, 2021).

Dalam dunia industri konsep *factor analysis* berkembang sangat pesat dan banyak digunakan. Salah satu penggunaannya adalah untuk mencari tahu faktor utama yang menyebabkan kegagalan dari pengoperasian mesin industri (Amruthnath & Gupta, 2019). Selain di bidang industri, terdapat banyak bidang yang sudah menerapkan *factor analysis*, diantaranya yaitu segmentasi pengguna, riset penyakit kanker, riset medis, dan riset masyarakat/lingkungan.

BAB III PELAKSANAAN MAGANG

3.1 *Data Science Roadmap*

Dalam pengerjaan proyek perangkat lunak dikenal sebuah istilah *engineering roadmap*. *Engineering roadmap* merupakan kumpulan seluruh tanggung jawab yang harus dilakukan oleh *engineer* -dalam industri software, yaitu *software engineer*- dalam proses pengerjaan proyek. Setiap perusahaan dan masing-masing divisi memiliki *engineering roadmap* yang berbeda-beda yang disesuaikan dengan budaya kerja dari perusahaan dan divisi tersebut.



Gambar 3.1 *Engineering Roadmap Data Scientist Jala*

Data scientist di Jala memiliki alur pengerjaan proyek tersendiri yang berbeda dengan *role* lain pada divisi *software*. Beberapa tahapan tersebut, yaitu *Planning*, *Execution*, *Review With Dev*, *Deployment/Presentation*. Gambar 3.1 menunjukkan alur *data science roadmap* yang ada di Jala.

3.1.1 *Planning*

Pada tahap ini pekerjaan difokuskan untuk menggali latar belakang dari proyek yang akan dikerjakan. Bagi *data scientist*, salah satu sumber untuk menggali latar belakang tersebut adalah dengan menganalisis data yang dimiliki. Setelah mendapatkan landasan pentingnya pengerjaan proyek, langkah selanjutnya adalah menulis PRD (*Product Requirement Document*). PRD adalah dokumen yang berisikan spesifikasi dari pengembangan proyek, mulai dari tujuan pengerjaan proyek, masalah yang akan diselesaikan, hasil akhir dari proyek, fitur-fitur yang akan dikerjakan hingga *timeline* pengerjaan proyek (Yue, 2016).



Gambar 3.2 Notion untuk penulisan PRD

. Penulisan PRD di Jala menggunakan *Notion*. *Notion* adalah platform yang menyediakan layanan untuk pengelolaan proyek, seperti menulis perencanaan, memasang target & *deadline*, serta kolaborasi dalam manajemen proyek (Yue, 2016). Gambar 3.2 menunjukkan penulisan PRD menggunakan Nation. PRD menjadi acuan dan dasar dalam pengerjaan proyek serta menjadi basis pertanggungjawaban kepada pimpinan. Di dalam PRD juga terdapat *timeline* pengerjaan proyek yang disertai dengan daftar tugas yang harus diselesaikan. Tabel 3.1 merupakan gambaran umum *timeline* pengerjaan proyek. Pengerjaan proyek ini ditargetkan selesai dalam 30 hari dengan fokus pengerjaan banyak dilakukan pada tahap ekstraksi dan *preprocessing*. Hal ini penting dilakukan agar data yang disajikan pada saat *clustering* dan *modeling* merupakan data yang berkualitas sehingga hasilnya dapat menunjukkan akurasi yang baik.

Tabel 3.1 Rangkuman *timeline* dari PRD

| No | Aktivitas | Durasi |
|----|--|---------|
| 1. | Ekstraksi data dan <i>preprocessing</i> . Ekstraksi dilakukan dengan mengambil data yang berada di <i>database</i> , yaitu data deskripsi tambak, deskripsi siklus, pencatatan kualitas air, dan data hasil panen. Ekstraksi juga dilakukan dengan mengambil seluruh aktivitas pengguna dalam <i>website</i> . Setelah itu, data yang sudah diekstraksi kemudian dibersihkan pada tahap <i>preprocessing</i> , salah satunya adalah mengelola <i>missing value / outlier data</i> lalu distandarisasi dengan format yang sama. | 14 hari |
| 2. | <i>Clustering</i> . Pada tahap <i>clustering</i> data banyak dilakukan riset dan uji coba metode dan jumlah kelompok yang tepat. Terdapat dua metode yang diuji coba, menggunakan <i>cohort</i> (bulan dan tahun pendaftaran tambak) dan menggunakan <i>k-means</i> . Setelah dilakukan uji coba, <i>k-means</i> menjadi pilihan sebagai algoritma <i>clustering</i> data. | 5 hari |
| 3. | <i>Modeling</i> . <i>Modeling</i> dilakukan dengan menerapkan algoritma <i>random forest</i> . | 9 hari |

3.1.2 Execution

Tahap selanjutnya adalah mengeksekusi seluruh persyaratan dan fitur yang telah dicantumkan di dalam PRD. Terdapat beberapa tahapan yang dilakukan dalam proses pengerjaan proyek yang meliputi, penentuan kriteria data yang akan digunakan, ekstraksi data, *preprocessing*, *clustering* menggunakan *K-Means*, *modeling* menggunakan algoritma *Random Forest*, dan mengekstrak nilai signifikansi setiap variabel sebagai luaran untuk melakukan *Factor Analysis*.

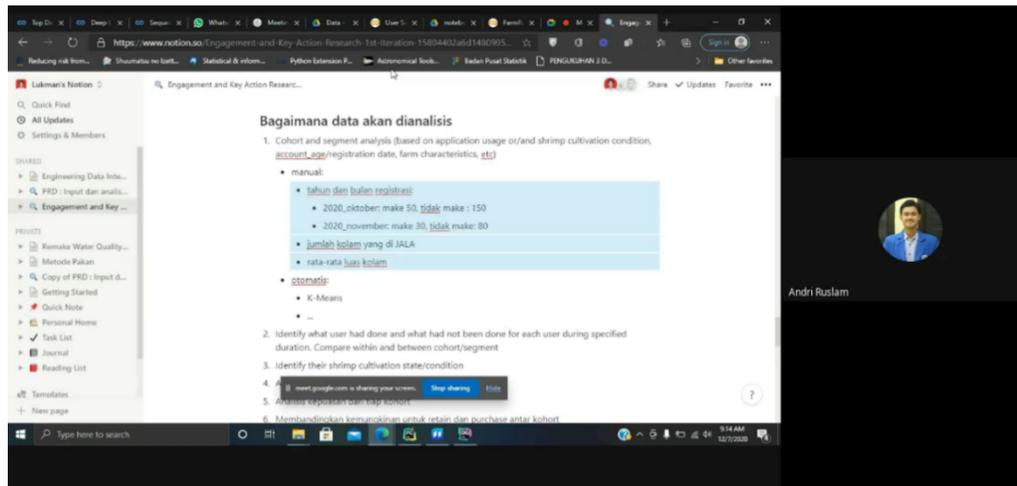


Gambar 3.3 Alur pengerjaan proyek

Dalam implementasinya tahapan ekstraksi hingga *modeling* merupakan proses yang berulang agar dapat mengoptimasi nilai dari setiap variabel sehingga mencapai hasil yang maksimal. Gambar 3.3 menunjukkan tahapan dalam pengerjaan dan deskripsi data yang digunakan dalam proyek ini.

3.1.3 Review With Dev

Review With Dev adalah tahapan evaluasi yang dilakukan oleh senior *data scientist* dan pemimpin proyek untuk mengecek pekerjaan berdasarkan PRD. Proses evaluasi dilakukan secara daring selama menjalani program magang dengan konsep WFH (*Work From Home*).

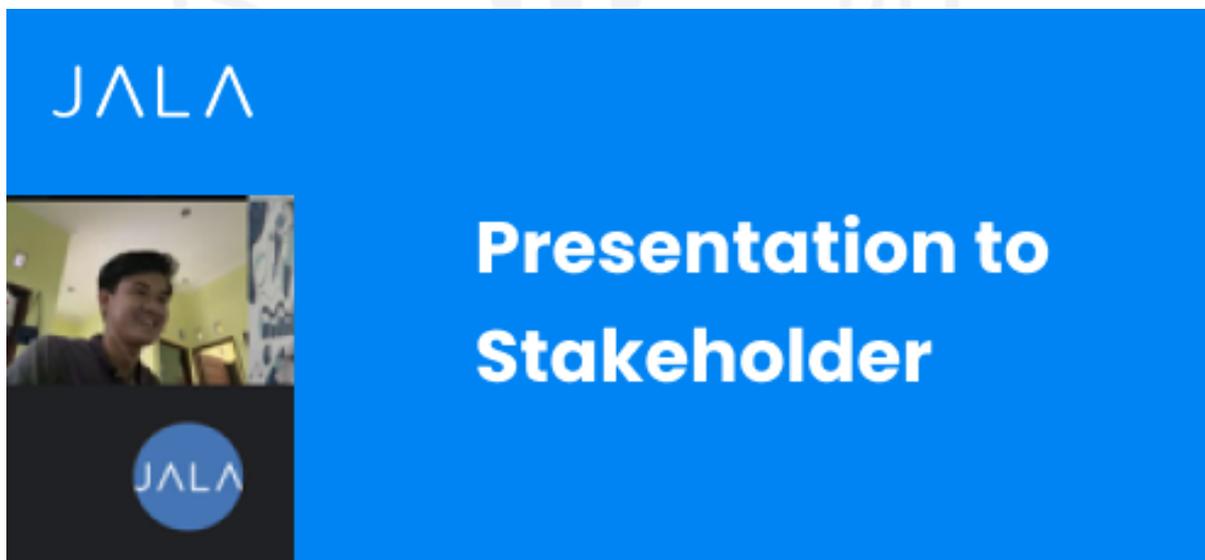


Gambar 3.4 Proses review bersama *lead of data scientist*

Gambar 3.4 menunjukkan proses evaluasi proyek. Dari hasil evaluasi yang dilakukan, proses implementasi biasanya akan kembali ke fase ekstraksi atau *preprocessing* untuk memperbaiki kualitas dari data.

3.1.4 *Deployment/Presentation*

Proyek yang telah dievaluasi dan diperbaiki kemudian disiapkan untuk *deployment*. Hasil dari proyek Analisis Keberlanjutan Pengguna belum masuk ke tahap *deployment*.



Gambar 3.5 Proses presentasi kepada stakeholder

Hasil akhir proyek ini difinalisasi dengan mempresentasikannya ke hadapan seluruh stakeholder Jala agar dapat menjadi pengetahuan dan mendapatkan umpan balik dari informasi

tersebut. Gambar 3.5 menunjukkan momen ketika mempresentasikan hasil proyek ke seluruh stakeholder Jala.

3.2 Implementasi Analisis Keberlanjutan Pengguna

Implementasi merupakan tahap pengerjaan proyek ke dalam bentuk kode program dengan menerapkan tahapan-tahapan yang terdapat pada Gambar 3.3. Tahapan-tahapan tersebut, yaitu persiapan data, ekstraksi, *preprocessing*, *clustering*, dan *modeling*. Penjelasan setiap tahap tersebut adalah sebagai berikut.

3.2.1 Kriteria data

Terdapat dua data berjenis teks yang akan diekstraksi dan digunakan dalam proyek ini. Data pertama adalah data aktivitas budi daya yang dimasukkan pengguna ke dalam *website*. Dari data ini, terdapat 46 kolom atau variabel yang diambil dari seluruh fitur yang berkaitan dengan budi daya, yaitu fitur sampling, pencatatan pakan, kualitas air, panen, deskripsi tambak, kolam, dan siklus yang tercatat di sistem. Tabel 3.2 menunjukkan sampel dari variabel-variabel yang digunakan.

Tabel 3.2 Variabel dari fitur-fitur *website*

| No | Nama variabel | Makna Variabel |
|----|-----------------------|--|
| 1. | farm_age | Usia tambak |
| 2. | user_count | Total pengguna yang terdaftar dalam satu tambak |
| 3. | total_pond | Total kolam dalam satu tambak |
| 4. | seed_average | Rata-rata jumlah benih yang telah ditebar |
| 5. | total_cycle | Total siklus |
| 6. | ph_completion_mean | Persentase kelengkapan data pH |
| 7. | shrimp_current_weight | Rata-rata berat udang per ekor saat melakukan sampling |
| 8. | sampling_data_count | Total data sampling |

Data kedua merupakan data hasil interaksi setiap pengguna dengan *website*. Dengan menggunakan data ini, akan terlihat fitur-fitur apa saja yang sering digunakan oleh pengguna dan bagaimana pengaruhnya terhadap keberlanjutan pengguna. Efektivitas fitur media juga akan

dianalisis berdasarkan data ini. Fitur media memiliki tiga sub fitur, yaitu sub fitur kabar udang, harga udang, dan penyakit udang.

Terdapat 159 variabel yang dianalisis dari data ini. Variabel-variabel tersebut berupa intensitas kunjungan atau interaksi yang dilakukan oleh pengguna, seperti jumlah kunjungan ke informasi kolam, intensitas pembuatan *form*, dan intensitas *login* dari pengguna. Tabel 3.3 menunjukkan sampel variabel / kolom interaksi dari pengguna.

Tabel 3.3 Variabel hasil interaksi pengguna dengan *website*

| No | Nama variabel | Makna Variabel |
|----|---------------------------------|---|
| 1. | user_id | ID pengguna |
| 2. | user_name | Nama pengguna |
| 3. | created_year | Tahun pengguna terdaftar di sistem |
| 4. | event_Create Farm: Filling Form | Banyak pengisian form pembuatan tambak |
| 5. | event_Visit Trend Shrimp Price | Banyak kunjungan ke fitur trend harga udang |
| 6. | event_Visit Ponds Detail | Banyak kunjungan ke detail kolam |
| 7. | event_Report: Export | Banyak export laporan tambak |
| 8. | farm_id | ID tambak |

Untuk data yang dimasukkan oleh pengguna, terdapat beberapa kriteria dan batasan tambahan. Tabel 3.4 menunjukkan kriteria dan batasan dari data yang digunakan. Data yang digunakan adalah seluruh data yang tercatat dari tahun 2018-2020 bulan November. Rentang pengambilan data, yaitu 30 hari sejak pengguna pertama kali mendaftarkan tambaknya.

Tabel 3.4 Kriteria data yang digunakan

| No. | Kriteria |
|-----|---|
| 1. | Seluruh data siklus yang tercatat hingga 22-11-2020 |
| 2. | Seluruh data dalam rentang 0-30 hari sejak tambak dibuat. |
| 3. | Data diagregasikan berdasarkan tambak. |
| 4. | Identifikasi lanjut atau berhenti menggunakan Jala ditentukan dari persentase kelengkapan data pakan dalam satu bulan setelah bulan pertama menggunakan Jala. Kelengkapan data di atas 30% diasumsikan lanjut menggunakan Jala. |

3.2.2 Ekstraksi Data

DBMS yang digunakan Jala adalah MySQL sehingga proses ekstraksi data akan dilakukan dengan bahasa SQL. Terdapat tiga data berjenis teks yang diekstraksi dari *database*, yaitu deskripsi siklus, deskripsi tambak, dan persentase kelanjutan menggunakan Jala. Penjelasan dari setiap data tersebut adalah sebagai berikut.

Deskripsi Siklus

Data ini merupakan data gabungan dari tabel tambak, kolam, sampling, pakan, dan kualitas air. Hasilnya berupa daftar siklus beserta informasi tambahan yang ada di dalamnya seperti total benih yang ditebar pada saat memulai siklus tersebut, tanggal mulai siklus, dan tanggal siklus berakhir. Pada budi daya udang, satu kolam bisa memiliki banyak siklus.

Dalam implementasinya, pembuatan *query* banyak menggunakan *with* dan *join clause* untuk menggabungkan beberapa tabel sekaligus. *With clause* adalah *query* pada SQL yang berguna sebagai *subquery* dengan tujuan untuk membangun relasi antar tabel sementara (*temporary relation*). Hasil dari relasi tersebut dapat digunakan berkali-kali untuk berbagai keperluan, salah satunya penggabungan tabel (Pedamkar, 2020).

```

with cycles_desc as (
  select
    cycles.id as cycle_id,
    ponds.farm_id,
    cycles.pond_id,
    cycles.started_at,
    coalesce(cycles.finished_at, '2020-11-23') as finished_at,
    datediff(coalesce(cycles.finished_at, '2020-11-23'), cycles.started_at) as doc,
    cycles.area,
    cycles.total_seed,
    if(cycles.finished_at is null, if(cycles.pro is null, 0, cycles.pro), null) as ongoing_pro,
    if(cycles.finished_at is null, if(cycles.pro is null, 1, 0), null) as ongoing_free,
    samplings.survival_rate
  from cycles
  inner join ponds
    on ponds.id = cycles.pond_id
  left join samplings
    on samplings.cycle_id = cycles.id
  order by cycles.id
),

```

Gambar 3.6 Penerapan *with clause*

```

inner join (
  select farms.id as farm_id,
    date_format(farms.created_at, '%Y-%m-%d') as registered_at
  from farms
) as farm_detail on farm_detail.farm_id = cycles_desc.farm_id

```

Gambar 3.7 Penerapan *join clause*

```

select
  cycles_desc.cycle_id,
  cycles_desc.farm_id,
  cycles_desc.pond_id,
  cycles_desc.started_at,
  cycles_desc.finished_at,
  cycles_desc.doc,
  feedings.feed_data_count,
  (feedings.feed_data_count / cycles_desc.doc) * 100 as feed_completion,
  feed_val.feed_total,
  feed_val.feed_average,
  cycles_desc.area,
  cycles_desc.total_seed,
  avg(if(cycles_desc.survival_rate is null, sr.survival_rate, cycles_desc.survival_rate))
  as sr_mean,
  stddev(if(cycles_desc.survival_rate is null, sr.survival_rate, cycles_desc.survival_rate))
  as sr_std,
  max(if(cycles_desc.survival_rate is null, sr.survival_rate, cycles_desc.survival_rate))
  as sr_max,
  min(if(cycles_desc.survival_rate is null, sr.survival_rate, cycles_desc.survival_rate))
  as sr_min,
  temperature.temp_data_count,

```

Gambar 3.8 Cuplikan *query* seluruh variabel

| cycle_id | farm_id | pond_id | started_at | finished_at | doc | feed_data_count |
|----------|---------|---------|------------|-------------|-----|-----------------|
| 1 | 3 | 1 | 2017-12-04 | 2018-07-20 | 228 | <null> |
| 2 | 3 | 2 | 2017-12-04 | 2018-07-20 | 228 | <null> |
| 3 | 3 | 3 | 2017-12-04 | 2018-07-20 | 228 | <null> |
| 4 | 3 | 4 | 2017-12-04 | 2018-07-20 | 228 | <null> |
| 6 | 40 | 2086 | 2018-04-17 | 2018-08-15 | 120 | <null> |
| 9 | 41 | 2108 | 2018-05-04 | 2018-09-01 | 120 | <null> |
| 10 | 41 | 2109 | 2018-05-04 | 2018-09-01 | 120 | <null> |
| 11 | 41 | 2105 | 2018-04-22 | 2018-08-20 | 120 | <null> |
| 12 | 41 | 2110 | 2018-04-22 | 2018-08-20 | 120 | <null> |
| 13 | 41 | 2104 | 2018-03-30 | 2018-07-28 | 120 | <null> |
| 14 | 41 | 2102 | 2018-04-03 | 2018-08-01 | 120 | <null> |
| 20 | 56 | 2202 | 2018-05-22 | 2018-09-19 | 120 | <null> |

Gambar 3.9 Hasil *query* deskripsi siklus

Gambar 3.6 menunjukkan penerapan dari *with* clause pada ekstraksi deskripsi siklus. *Query* tersebut bertujuan untuk membuat relasi tabel sementara antara tabel siklus, kolam, dan sampling. Hasil dari relasi tersebut kemudian digunakan pada kode SQL selanjutnya untuk melakukan penggabungan dengan tabel yang lain.

Join clause bertujuan untuk menggabungkan dua atau lebih tabel menjadi satu kesatuan (Mahajan, 2019). Gambar 3.7 menunjukkan cuplikan dari salah satu *join clause*. Pada cuplikan tersebut, *query join* digunakan untuk menggabungkan data tambak dengan hasil dari *query with*. Tujuannya adalah agar tabel yang dihasilkan memiliki data tanggal pembuatan tambak sehingga dapat dilakukan penyaringan terhadap data yang berada pada rentang 30 hari pertama sejak tambak dibuat. Gambar 3.8 menunjukkan cuplikan dari keseluruhan variabel yang diambil dari *database*. *Query* tersebut berguna untuk mendaftar seluruh kolom yang akan ditampilkan sebagai hasil dari ekstraksi deskripsi siklus. Sementara Gambar 3.9 menunjukkan luaran dari deskripsi siklus.

Deskripsi Tambak

Deskripsi tambak berisi informasi detail dari sebuah tambak, mulai dari tanggal inisiasi tambak hingga lokasi tambak. Data ini dihasilkan dari gabungan beberapa tabel, yaitu tabel tambak, kolam, wilayah, dan tabel pengguna. Tujuan dari data deskripsi tambak adalah untuk menarik informasi tentang detail dari tambak tersebut, salah satunya adalah tanggal pembuatan tambak.

```

with farm_information as (
  select farms.id as farm_id,
         farms.name as farm_name,
         datediff('2020-11-23', farms.created_at) as farm_age,
         if(length(districts.id) = 4, districts.name, null) as district,
         if(length(proviences.id) = 2, proviences.name, null) as province,
         year(farms.created_at) as register_year,
         month(farms.created_at) as register_month
  #         (sum(if(ponds.id is not null, 1, 0))) as total_pond
  from farms
  inner join ponds
    on ponds.farm_id = farms.id
  left join regions districts
    on left(farms.region_id, 4) = districts.id
  left join regions proviences
    on left(farms.region_id, 2) = proviences.id

  where farms.created_at < '2020-11-23'
  group by farms.id, districts.id, proviences.id
)

select farm_information.*,
       farm_users.user_count

```

Gambar 3.10 Cuplikan *query* deskripsi tambak

| farm_id | farm_name | farm_age | district | province | register_year |
|---------|-------------------------|----------|-------------|---------------|---------------|
| 12 | SDB Blok B | 992 | CIANJUR | JAWA BARAT | 2018 |
| 9 | Tambak Tegalsari | 992 | SUBANG | JAWA BARAT | 2018 |
| 8 | supm tegal | 992 | TEGAL | JAWA TENGAH | 2018 |
| 11 | Tambak Dinas DKP Congot | 992 | KULON PROGO | DI YOGYAKARTA | 2018 |
| 19 | RAS Kerapu | 992 | SLEMAN | DI YOGYAKARTA | 2018 |
| 14 | Safaraz Berdaya Farm | 992 | PURWOREJO | JAWA TENGAH | 2018 |
| 248 | NRT | 679 | PURWOREJO | JAWA TENGAH | 2019 |
| 1008 | SBF | 494 | PURWOREJO | JAWA TENGAH | 2019 |
| 3 | PW I | 992 | PURWOREJO | JAWA TENGAH | 2018 |
| 18 | Tambak RAS Udang | 992 | SLEMAN | DI YOGYAKARTA | 2018 |

Gambar 3.11 Hasil *query* deskripsi tambak

Gambar 3.10 menunjukkan cuplikan dari *query* deskripsi tambak. Salah satu fokus pada *query* tersebut adalah mengekstraksi lokasi kabupaten dan provinsi tambak dengan menggabungkan tabel tambak dan tabel wilayah. Hal ini karena setiap tambak memiliki *id* wilayah yang berisi penggabungan antara *id* kabupaten dan provinsi sehingga perlu dilakukan pengolahan terlebih dahulu untuk mendapatkan informasi tersebut. Hasil dari *query* ditunjukkan pada Gambar 3.11.

Kelanjutan Menggunakan Jala

Parameter keberlanjutan menggunakan Jala dihitung dari kelengkapan data pakan. Proses perhitungan dilakukan dengan mengkalkulasikan data pakan pada 30 hari selanjutnya setelah satu bulan pertama siklus berjalan.

```
with cycles_desc as (
  select
    cycles.id as cycle_id,
    ponds.farm_id,
    cycles.pond_id,
    cycles.started_at,
    coalesce(cycles.finished_at, '2020-11-23') as finished_at,
    datediff(coalesce(cycles.finished_at, '2020-11-23'), cycles.started_at) as doc
  from cycles
  inner join ponds
    on ponds.id = cycles.pond_id
  order by cycles.id
)
select
  cycles_desc.cycle_id,
  cycles_desc.farm_id,
  cycles_desc.pond_id,
  cycles_desc.started_at,
  feedings.day_31,
  feedings.day_after_30,
  cycles_desc.finished_at,
```

Gambar 3.12 Cuplikan *query* keberlanjutan pengguna

| cycle_id | farm_id | pond_id | started_at | day_31 | day_after_30 | finished_at |
|----------|---------|---------|------------|------------|--------------|-------------|
| 107 | 104 | 2476 | 2018-09-28 | 2018-10-28 | 2018-11-27 | 2019-01-26 |
| 111 | 108 | 2497 | 2018-07-25 | 2018-08-24 | 2018-09-23 | 2018-10-12 |
| 112 | 108 | 2498 | 2018-07-25 | 2018-08-24 | 2018-09-23 | 2018-10-15 |
| 139 | 118 | 2545 | 2018-09-25 | 2018-10-25 | 2018-11-24 | 2019-01-23 |
| 140 | 118 | 2552 | 2018-09-25 | 2018-10-25 | 2018-11-24 | 2019-01-23 |
| 141 | 118 | 2546 | 2018-09-25 | 2018-10-25 | 2018-11-24 | 2019-01-23 |
| 142 | 118 | 2547 | 2018-09-25 | 2018-10-25 | 2018-11-24 | 2019-01-23 |
| 143 | 118 | 2548 | 2018-09-25 | 2018-10-25 | 2018-11-24 | 2019-01-23 |
| 144 | 118 | 2549 | 2018-09-25 | 2018-10-25 | 2018-11-24 | 2019-01-23 |
| 145 | 118 | 2550 | 2018-09-25 | 2018-10-25 | 2018-11-24 | 2019-01-23 |
| 146 | 118 | 2551 | 2018-09-25 | 2018-10-25 | 2018-11-24 | 2019-01-23 |
| 154 | 135 | 2652 | 2018-08-20 | 2018-09-19 | 2018-10-19 | 2018-12-08 |
| 156 | 124 | 2622 | 2018-10-31 | 2018-11-30 | 2018-12-30 | 2019-01-25 |

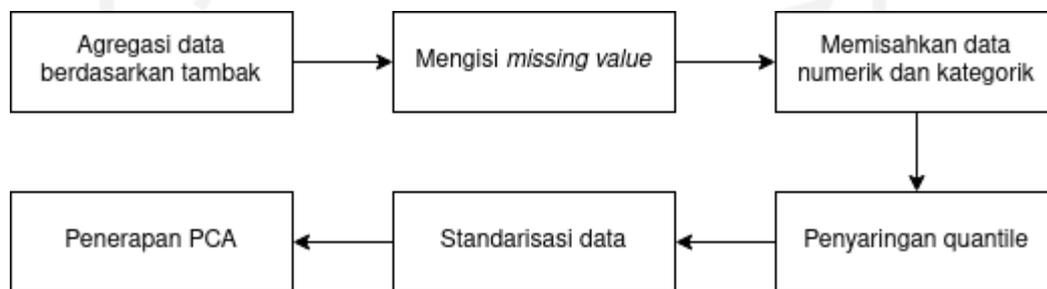
Gambar 3.13 Hasil *query* keberlanjutan pengguna

Data yang memiliki persentase sama dengan atau lebih dari 30% dikategorikan sebagai pengguna yang lanjut menggunakan Jala. Sementara pengguna yang memiliki kelengkapan data pakan kurang dari 30% dikategorikan tidak lanjut menggunakan Jala. Gambar 3.12 dan Gambar 3.13 menunjukkan *query* dari data kelanjutan pengguna dan hasilnya.

Data yang telah diekstrak dari *database* kemudian disimpan dalam bentuk *.csv*. CSV atau *Comma Separated Values* merupakan format data yang menyimpan kumpulan data dalam bentuk teks dengan menggunakan koma sebagai pemisah antara teks (Hoffman, 2018). CSV digunakan karena sifatnya yang ringan, simpel, dan mudah dibaca oleh mesin (Carvalho et al., 2015).

3.2.3 Preprocessing

Preprocessing adalah salah satu tahapan yang akan dilalui oleh *data scientist* untuk mengelolah data mentah yang dimiliki menjadi data yang terstruktur dan berkualitas. Salah satu indikasi dari data yang berkualitas adalah data tersebut mengandung variabel berkaitan dengan analisis yang akan dilakukan serta memiliki struktur yang konsisten.

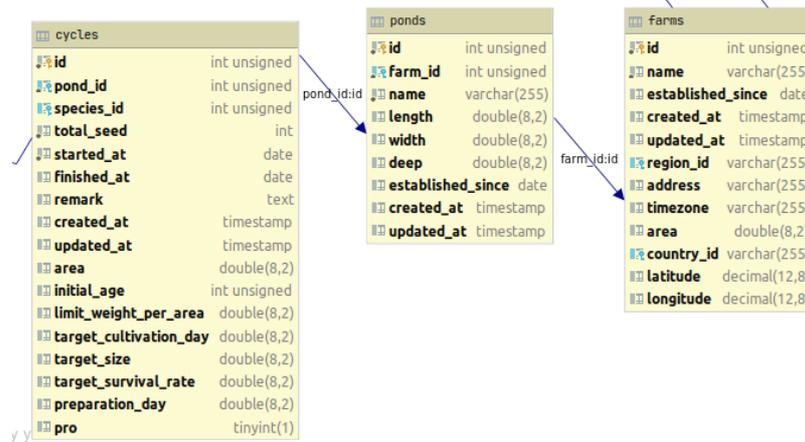


Gambar 3.14 Tahapan *preprocessing* yang dilakukan

Preprocessing dilakukan dengan menggunakan *python* versi 3.8.5 dengan *library Pandas*. Tahapan-tahapan yang dilakukan selama proses *preprocessing* adalah agregasi data berdasarkan tambak, mengisi *missing value*, memisahkan data numerik dan kategorik, penyaringan *quantile*, standarisasi data, dan penerapan PCA (*Principal Component Analysis*) seperti yang ditunjukkan pada Gambar 3.14.

Agregasi Data Berdasarkan Tambak

Terdapat tiga data yang dihasilkan dari tahap ekstraksi, yaitu data deskripsi siklus, tambak, dan kelanjutan pengguna. Ketiga data ini masih terpisah satu sama lain sehingga diperlukan *preprocessing* untuk menggabungkan data tersebut. Penggabungan dilakukan dengan mengagregasi data berdasarkan data tambak. Hal ini dilakukan karena proses analisis akan dilakukan per tambak.



Gambar 3.15 Relasi tabel siklus, kolam, dan tambak

Group by pond_id from cycles

```
1 pond_sum_cycle = cycle_detail.groupby('pond_id').size().reset_index()
2 pond_sum_cycle = pond_sum_cycle.rename(columns={0 : "total_cycle"})
```

```
1 pond_detail_max = cycle_detail.groupby('pond_id')[['sr_max']].max().reset_index()
2 pond_detail_min = cycle_detail.groupby('pond_id')[['sr_min']].min().reset_index()
```

```
1 pond_detail_std = cycle_detail.groupby('pond_id')[['sr_std']].std().reset_index()
```

```
1 pond_detail_mean = cycle_detail.groupby('pond_id')[['feed_average', 'sr_mean',
2               'temp_mean', 'do_mean', 'ph_mean',
3               'salinity_mean', 'shrimp_current_weight',
4               'shrimp_growth_average', 'sampling_count',
5               'total_seed', 'area', 'farm_id', 'feed_completion',
6               'temp_completion', 'do_completion',
7               'ph_completion', 'salinity_completion']].mean().reset_index()
8
9 pond_detail_mean = pond_detail_mean.rename(columns={"sampling_count" : "sampling_data_average",
10            'total_seed' : "seed_average",
11            'feed_completion' : 'feed_completion_mean',
12            'temp_completion' : 'temp_completion_mean',
13            'do_completion' : 'do_completion_mean',
14            'ph_completion' : 'ph_completion_mean',
15            'salinity_completion' : 'salinity_completion_mean'})
```

Gambar 3.16 Agregasi data berdasarkan kolom

Gambar 3.15 menunjukkan skema relasi dari setiap tabel. Berdasarkan gambar tersebut, terlihat bahwa siklus dimiliki oleh kolam dan kolam dimiliki oleh tambak. Dengan demikian untuk mengagregasi data siklus diperlukan penggabungan berdasarkan kolam terlebih dahulu, lalu diagregasikan berdasarkan tambak. Gambar 3.16 menunjukkan cuplikan kode agregasi data berdasarkan kolom menggunakan *Pandas*. Setelah agregasi data berdasarkan kolom, hasilnya kemudian diagregasikan berdasarkan tambak.

Mengisi *Missing Value*

Salah satu permasalahan yang sering dihadapi dalam melakukan *preprocessing* data adalah mengisi *missing value*. *Missing value* dihasilkan karena banyak hal, salah satunya adalah *query* SQL yang kurang teliti sehingga terdapat logika yang salah dan menghasilkan nilai *null*.

```

1 def fill_empty_after_filtering(df, lc) :
2   df.fillna(df.mean(), inplace=True)
3
4   return df

```

Gambar 3.17 Cuplikan *method* mengisi *missing value*

Terdapat beberapa teknik yang digunakan dalam mengatasi *missing value*, di antaranya yaitu, menggunakan rata-rata, mengisi dengan *default value*, dan menggunakan total data. Seluruh teknik tersebut digunakan dalam proyek ini. Gambar 3.17 menunjukkan cuplikan *method* untuk mengisi *missing value*. Pada *method* tersebut, pengisian *missing value* dilakukan dengan metode menghitung rata-rata dari seluruh data pada kolom tersebut. Hasil perhitungannya kemudian dimasukkan untuk mengisi *missing value*.

Memisahkan Data Numerik dan Kategorik

Data numerik adalah data yang bersifat kuantitatif. Data kuantitatif identik dengan skala numerik sehingga dapat dikalkulasikan dengan operasi matematika. Sementara, data kategorik adalah data yang bersifat kualitatif. Data kualitatif menggunakan skala karakter atau *string* sehingga tidak dapat dilakukan operasi matematika.

```

1 farms_data = farms_data.drop(['farm_id', 'farm_name', 'register_year', 'register_month',
2                               'province', 'district', 'is_use_jala'], axis=1)

```

Gambar 3.18 Cuplikan pemisahan data numerik dan kategorik

Dalam *preprocessing* untuk mempersiapkan data ke tahap *clustering*, diperlukan pemisahan antara data kategorik dan data numerik. Proses pemisahan dilakukan dengan menghapus sementara variabel kategorik menggunakan fungsi *drop* sebagaimana yang terlihat pada Gambar 3.18

Penyaringan *Quantile*

Quantile adalah salah satu metode dalam ilmu statistik yang tujuannya adalah membagi data menjadi beberapa kelompok yang sama besar (Taylor, 2018). Median adalah salah satu contoh dari *quantile* yang membagi data menjadi dua kelompok yang sama besar. Dalam implementasi metode *quantile* dapat digunakan untuk mengeluarkan data *outlier*.

```

1 def filtering_quantile(df, lc) :
2     df_temp = pd.DataFrame()
3
4     for column_name in lc :
5         cond = df.loc[:, column_name].between(df.loc[:,column_name].quantile(0.05),
6                                               df.loc[:,column_name].quantile(0.95))
7         val_temp = df.loc[cond, column_name].copy()
8         df_temp = pd.concat([df_temp, val_temp], ignore_index=True, axis=1)
9
10    return df_temp

```

Gambar 3.19 Cuplikan *method* penyaringan *quantile*

Data outlier adalah data yang memiliki nilai yang berbeda jauh dengan mayoritas data lain (Bakar et al., 2006). Gambar 3.19 menunjukkan penerapan dari *quantile*. Pada kode tersebut batas bawah dan atas dari *quantile* bernilai 0.05 dan 0.95. Penggunaan nilai ini merupakan hasil *tuning* nilai dengan melihat perbandingan antara data yang tersedia dan terbuang. Semakin banyak yang terbuang tidak berdampak baik terhadap kualitas data sehingga harus dipastikan bahwa data yang terbuang masih dalam cakupan proporsional. Penetapan proporsional pada proyek ini dilihat dari jumlah data, jumlah data yang ditetapkan tidak diperkenankan kurang dari 1000 baris.

Sebelum

| | |
|---------|---|
| 1 | <code>farms_data.shrimp_current_weight.max()</code> |
| 80000.0 | |

Sesudah

| | |
|--------|--|
| 1 | <code>farm_result.shrimp_current_weight.max()</code> |
| 16.222 | |

Gambar 3.20 Hasil penerapan *quantile*

Salah satu hasil dari penerapan *quantile* ditunjukkan pada Gambar 3.20 yang menunjukkan nilai maksimum dari variabel *shrimp_current_weight*. *Shrimp_current_weight* merupakan variabel yang menunjukkan rata-rata berat udang per ekor dalam satuan gram. Sebelum dilakukan penyaringan *quantile* nilai maksimum menunjukkan angka 80000.0 padahal hal ini tidak mungkin terjadi. Dengan menerapkan *quantile* hal tersebut dapat disaring sehingga nilai maksimum yang dihasilkan sesuai dengan kondisi lapangan, yaitu rata-rata berat udang per ekor adalah 14.99 gram (Supono, 2006).

Standarisasi Data

Perbedaan rentang nilai antara satu variabel dan variabel lain merupakan salah satu masalah yang harus diselesaikan. Perbedaan nilai ini akan berpengaruh terhadap hasil yang akan diperoleh dari tahap *clustering* menggunakan algoritma *K-Means*. Oleh karena itu, diperlukan standarisasi data. Standarisasi data bertujuan untuk mengubah rentang nilai setiap variabel menjadi sama satu sama lain.

Sebelum

| | farm_age | user_count | total_pond | pond_area_mean | total_pond_area |
|-----|----------|------------|------------|----------------|-----------------|
| min | 28.0 | 1.0 | 1.0 | 9.0 | 10.0 |
| max | 631.0 | 3.0 | 6.0 | 7500.0 | 19846.0 |

Sesudah

| StandardScaler Result | |
|-----------------------|-----------|
| min | -1.645280 |
| max | 13.777293 |

Gambar 3.21 Hasil standarisasi data

Dalam pengerjaan proyek ini, standarisasi dilakukan dengan menggunakan fungsi *StandardScaler* dari *library sklearn*. Fungsi ini diimplementasikan berdasarkan Rumus (3.1) yang akan mentransformasikan setiap nilai sehingga hasil akhirnya memiliki rata-rata sama dengan 0 dan standar deviasi sama dengan 1 (Brownlee, 2020). z adalah variabel hasil transformasi, x adalah nilai sampel, μ adalah rata-rata dan σ adalah standar deviasi.

$$z = \frac{x - \mu}{\sigma} \quad (3.1)$$

Perbandingan hasil dari penerapan standarisasi data ditunjukkan pada Gambar 3.21. Pada gambar tersebut terlihat bahwa sebelum dilakukan standarisasi, setiap kolom memiliki rentang nilai yang berbeda-beda, mulai dari puluhan hingga ribuan. Setelah dilakukan standarisasi data, hasilnya menunjukkan seluruh variabel memiliki rentang nilai yang sama.

Penerapan PCA (*Principal Component Analysis*)

PCA merupakan teknik menganalisis multivariabel (*multivariate technique*) yang bertujuan untuk memperkecil dimensi dari suatu data (López del Val & Alonso Pérez de Agreda, 1993). Dalam proses implementasi, penentuan jumlah dimensi yang akan dibentuk dari PCA ditentukan berdasarkan nilai *variance* dari masing-masing variabel.

```

1 np.cumsum(pca.explained_variance_ratio_)
array([0.22065611, 0.31427913, 0.38032754, 0.42939096, 0.47596618,
       0.51924364, 0.5619235 , 0.59829995, 0.63167804, 0.66295968,
       0.68997068, 0.71476315, 0.7387266 , 0.76203991, 0.78368405,
       0.8035308 , 0.82273988, 0.84026495, 0.85704603, 0.87209149,
       0.88677885, 0.89986113, 0.91211676, 0.92374926, 0.93405429,
       0.94390165, 0.95313434, 0.96175222, 0.96984098, 0.97584282,
       0.98167678, 0.98713319, 0.99219709, 0.99542115, 0.99781692,
       1.          , 1.          , 1.          ])

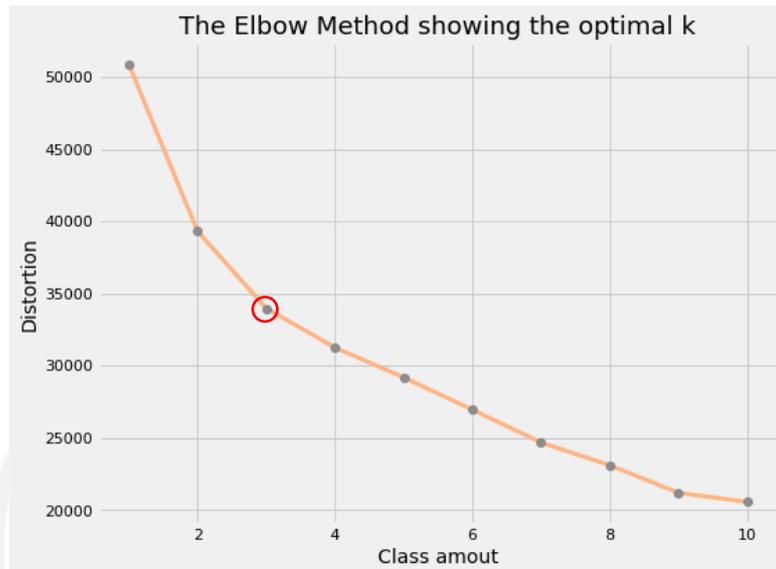
```

Gambar 3.22 Ekstraksi nilai *variance* dari setiap variabel

Gambar 3.22 menunjukkan perhitungan *variance* dari setiap variabel. Nilai *variance* yang digunakan, yaitu kurang dari 0.7 sehingga penerapan *PCA* menghasilkan 11 variabel yang akan menjadi masukan dalam proses selanjutnya.

3.2.4 Clustering

Proses *clustering* dilakukan untuk mengelompokkan data ke dalam beberapa kelompok tertentu. Tujuan dari penerapan *clustering* pada proyek ini adalah untuk menambahkan variabel label hasil *clustering* dan melakukan perbandingan antara data tanpa penambahan label dan dengan penambahan label. *Clustering* dilakukan dengan menerapkan algoritma *K-Means* menggunakan *Elbow Method* dalam menentukan nilai K yang optimum. *Elbow method* berfungsi untuk membantu menentukan berapakah jumlah K yang optimal digunakan pada algoritma *K-Means* sesuai dengan data yang dimiliki. Proses penentuan *Elbow Method* dilakukan dengan melakukan *clustering*, mulai dari satu hingga sejumlah $n-1$ kelas. Variabel n merupakan total jumlah variabel yang digunakan -dalam hal ini 11 variabel-. Langkah selanjutnya adalah menghitung nilai SSE (*Sum Squared Error*) dari setiap kelas yang telah dikelompokkan.



Gambar 3.23 Visualisasi *elbow method*

Pada umumnya, nilai K akan ditentukan secara manual dengan melakukan *plotting* hasil perhitungan setiap SSE. Setelah itu, kelas yang memiliki titik berbentuk sudut ditetapkan sebagai nilai K sebagaimana yang terdapat pada Gambar 3.23. Akan tetapi, pada *python* dapat menggunakan *library kneed* dengan fungsi *KneeLocator* untuk menentukan nilai K dari hasil perhitungan SSE. *Library* ini menerapkan algoritma “*kneedle*” yang merupakan salah satu teknik dalam menghitung *operating point* pada *system behaviour* yang identik dengan pembentukan siku pada diagram (Satopaa, Albrecht, Irwin, & Raghavan, n.d.). Hasil dari perhitungan tersebut didapatkan nilai K = 3.

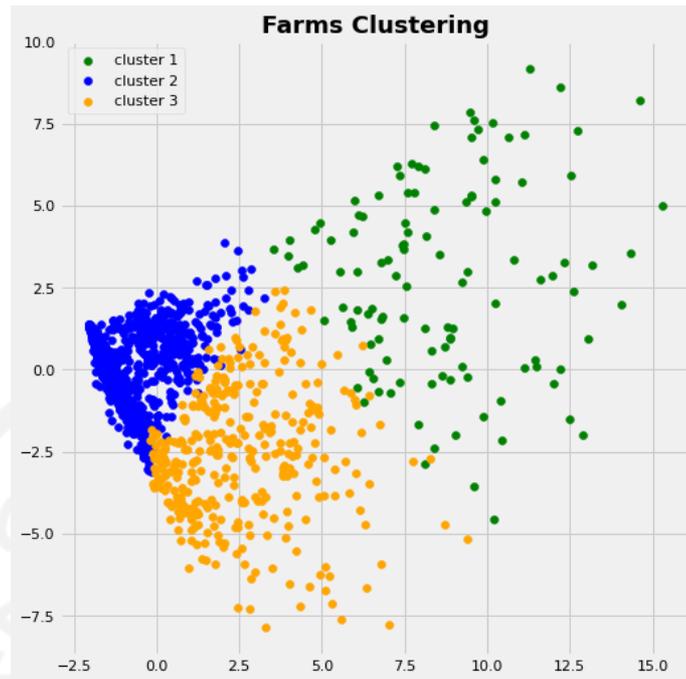
Implement Sklearn.K-Means

```

: 1 kmeans_farm = KMeans(
2     init="random",
3     n_clusters=cluster_needed,
4     n_init=10,
5     max_iter=300
6     # random_state=42
7 )
8 kmeans_farm.fit(pca_farm)

```

Gambar 3.24 Cuplikan penerapan *K-Means*

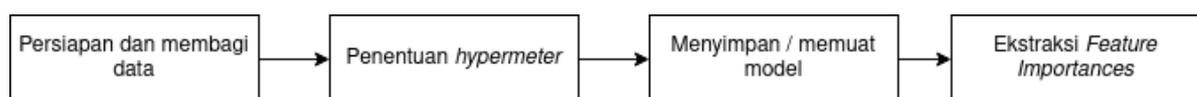


Gambar 3.25 Hasil *clustering*

Nilai K tersebut dimasukkan dalam penerapan algoritma *K-Means*. Pada proyek ini, *K-Means* diterapkan menggunakan *library sklearn*. Gambar 3.24 menunjukkan cuplikan kode penerapan *K-Means*. *cluster_needed* adalah hasil dari penerapan *elbow method* sebelumnya. Gambar 3.25 menunjukkan hasil dari *clustering*. Terdapat tiga kelas yang ditunjukkan dengan warna yang berbeda-beda. Warna hijau menunjukkan *cluster 1*, warna biru merupakan *cluster 2*, dan warna kuning menunjukkan *cluster 3*. Hasil dari *clustering* kemudian dijadikan variabel tambahan dengan nama variabel/kolom label untuk data yang masuk pada tahap *clustering*.

3.2.5 Modeling

Model adalah luaran yang dihasilkan dari penerapan algoritma *Machine Learning* yang dapat digunakan untuk melakukan prediksi pada data baru (Brownlee, 2020). Dalam proyek ini, *modeling* akan dilakukan menggunakan algoritma *Random Forest*. Hasil akhir dari *modeling* adalah nilai signifikansi dari setiap variabel (*feature importances*).



Gambar 3.26 Alur *Modeling*

Terdapat beberapa tahapan yang dilakukan untuk mendapatkan nilai signifikansi dari tiap variabel. Tahapan tersebut, yaitu persiapan & membagi data, penentuan *hyperparameter*,

menyimpan dan memuat model, dan ekstraksi *feature importances* dari masing-masing variabel seperti terdapat pada Gambar 3.26. Penjelasan dari setiap tahapan tersebut adalah sebagai berikut.

Persiapan dan Membagi Data

Terdapat dua jenis data yang akan digunakan dalam proses modeling, yaitu seluruh data tanpa penambahan variabel label (tidak melalui proses *clustering*) dan data dengan penambahan variabel label (melalui proses *clustering*). Hal ini berguna untuk membandingkan hasil antara kedua data tersebut. Setelah data telah dipersiapkan, langkah selanjutnya adalah membagi data menjadi dua bagian, yaitu data *training* dan *test*.

| | num_x_train_data | num_x_test_data | num_y_train_data | num_y_test | total_variable |
|---------------|------------------|-----------------|------------------|------------|----------------|
| without_label | 1624 | 287 | 1624 | 287 | 197 |
| with_label | 1624 | 287 | 1624 | 287 | 198 |

Gambar 3.27 Hasil pembagian data

Data *training* digunakan untuk melatih algoritma *Random Forest*, sementara data *test* digunakan untuk mengevaluasi model yang telah dihasilkan. Pembagian data *training* dan *test* dibagi dengan perbandingan 85:15 persen. Pada proyek ini, pembagian dilakukan menggunakan library *sklearn* dengan fungsi *train_test_split*. Hasil dari pembagian tersebut dapat dilihat pada Gambar 3.27. Berikut adalah keterangan dari setiap variabel.

- num_x_train_data*: jumlah data *training* x (*independent variable*)
- num_x_test_data*: jumlah data *test* x
- num_y_train_data*: jumlah data *training* y (*dependent variable*)
- num_y_test_data*: jumlah data *test* y
- total_variable*: Total keseluruhan variabel.

Penentuan *Hyperparameter*

Hyperparameter berfungsi untuk mengoptimasi algoritma agar dapat memberikan hasil luaran yang akurat. Secara *default*, penentuan *hyperparameter* dilakukan secara manual dengan mengganti nilainya dan melakukan *training*. Akan tetapi, pada *python*, terdapat *library* yang mampu mencari nilai *hyperparameter* terbaik dengan menggunakan fungsi *GridSearchCV*.

```

1 params= {'classifier_max_depth': [None,5, 10, 20, 30],
2         'classifier_criterion': ['entropy'],
3         'classifier_min_samples_split': [2,6,12],
4         'classifier_min_samples_leaf': [1,5,9]}
5
6 selector= GridSearchCV(estimator= model, param_grid= params)
7 selector.fit(x_train, y_train)
8 selected_model= selector.best_estimator_

```

Gambar 3.28 Inisialisasi *hyperparameter*

```

Pipeline(steps=[('scaler', StandardScaler()),
                 ('classifier',
                  RandomForestClassifier(criterion='entropy', max_depth=30,
                                       min_samples_split=6))])

```

Gambar 3.29 *Hyperparameter* paling optimum

Gambar 3.28 menunjukkan *hyperparameter* yang dimasukkan ke dalam fungsi *GridSearchCV*. *Hyperparameter* paling optimum untuk data tanpa label dan dengan label ditunjukkan pada Gambar 3.29.

Menyimpan dan Memuat Model

Melatih suatu model membutuhkan waktu yang cukup lama terlebih apabila data yang dilatih sangat banyak. Oleh karena itu, diperlukan fungsi simpan dan muat model sehingga ketika model telah selesai dilatih dapat disimpan dan bisa langsung dimuat tanpa proses *training* lagi. Terdapat dua model yang dihasilkan dari penelitian ini. Kedua model tersebut kemudian disimpan dalam format *.pickle*.

```

1 def save_model(selected_model, file_name):
2     model_result= {
3         'split': {'train': {'x': x_train, 'y': y_train},
4                 'test': {'x': x_test, 'y': y_test}
5                 },
6         'model': selected_model
7     }
8
9     CURR_DIR = os.path.dirname(os.path.realpath('__file__'))
10
11     with open(os.path.join(CURR_DIR, (file_name+'.pickle')), 'wb') as infile:
12         pickle.dump(model_result, infile)
13         infile.close()
14     del infile

```

```

1 def load_model(file_name) :
2     CURR_DIR = os.path.dirname(os.path.realpath('__file__'))
3
4     with open(os.path.join(CURR_DIR, (file_name+'.pickle')), 'rb') as infile:
5         result= pickle.load(infile)
6         infile.close()
7
8     del infile
9     return result

```

Gambar 3.30 Cuplikan *method* menyimpan dan memuat model

Pickle adalah ekstensi *python* yang berisikan hasil dari serialisasi struktur data ke dalam bentuk bit sehingga struktur tersebut dapat disimpan pada memori atau ditransmisi melalui internet (Python, n.d.). Gambar 3.30 menunjukkan *method* untuk menyimpan dan memuat model.

Ekstraksi *Feature importances*

Model yang berhasil disimpan dapat digunakan untuk mengekstrak nilai *feature importances* dari setiap variabel. *Feature importances* adalah salah satu parameter yang didapatkan dari penerapan algoritma *Random Forest*.

```

1 # Load model
2 result = load_model('random_forest_model_1_web_data')
3
4 x_train= result['split']['train']['x']
5 y_train= result['split']['train']['y']
6 selected_model= result['model']

1 # List fitur-fitur penting
2
3 feature_importances= pd.DataFrame()
4 feature_importances['features']= x_train.columns
5 feature_importances['feature_importances']= selected_model['classifier'].feature_importances_.copy()
6 feature_importances= feature_importances.sort_values('feature_importances', ascending= False)

```

Gambar 3.31 Ekstraksi nilai signifikansi variabel

Parameter inilah yang menjadi penilaian pengaruh variabel terhadap keberlanjutan pengguna. Gambar 3.31 menunjukkan baris kode untuk mengekstraksi nilai signifikansi dari setiap variabel.

BAB IV

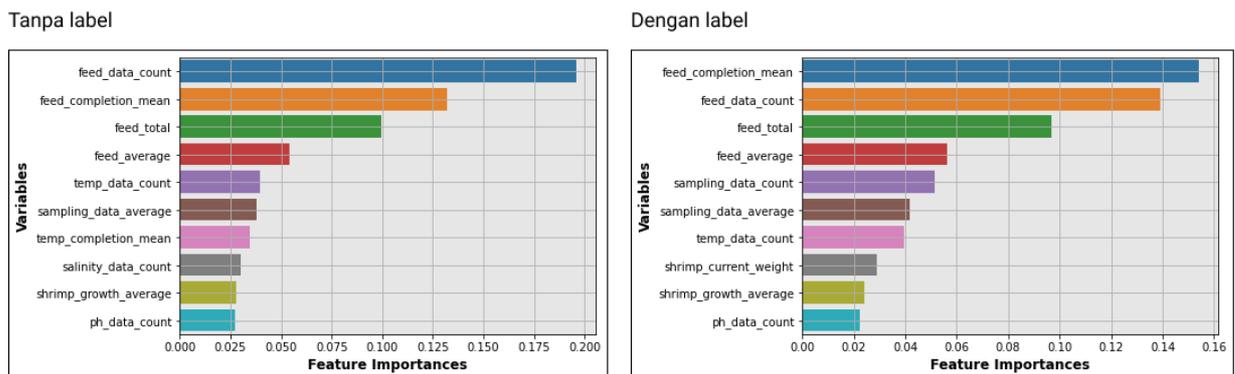
REFLEKSI PELAKSANAAN MAGANG

4.1 Teknis

Terdapat beberapa hal yang dapat direfleksikan selama menjalani proyek analisis keberlanjutan pengguna. Beberapa hal tersebut, yaitu refleksi analisis fitur-fitur paling berpengaruh, analisis fitur media, pengalaman dalam tahap *preprocessing*, dan pengalaman dalam menggunakan algoritma *K-Means*.

4.1.1 Analisis Fitur Berpengaruh

Hasil ekstraksi *feature importances* menjadi landasan untuk melakukan *Factor Analysis*. Terdapat dua jenis data yang digunakan, yaitu data tanpa hasil pelabelan menggunakan *clustering* dan data dengan pelabelan hasil *clustering*.



Gambar 4.1 Analisis fitur yang paling berpengaruh

Gambar 4.1 menunjukkan perbandingan antara dua data yang digunakan sebagai masukan algoritma *Random Forest*. Dari total 205 variabel yang terlibat dalam proses *modeling*, dilakukan pengurutan variabel berdasarkan pengaruhnya terhadap keberlanjutan pengguna mulai dari terbesar ke terkecil. Setelah dilakukan pengurutan, kemudian diambil 10 variabel teratas yang paling berpengaruh terhadap keberlanjutan pengguna dari kedua jenis data yang dimasukkan ke dalam tahap modeling. Hasilnya menunjukkan nilai yang hampir sama satu sama lain, yaitu sepuluh faktor paling dominan berpengaruh terhadap keberlanjutan pengguna adalah variabel yang berkaitan dengan fitur pencatatan pakan, sampling, dan kualitas air.

Fitur pencatatan pakan menjadi fitur yang paling berpengaruh menurut kedua data tersebut dengan persentase pengaruhnya yang paling besar adalah 20%. Setelah fitur pencatatan pakan, fitur selanjutnya yang berpengaruh signifikan adalah fitur sampling dan pencatatan kualitas air dengan persentase masing-masing sebesar 5%.

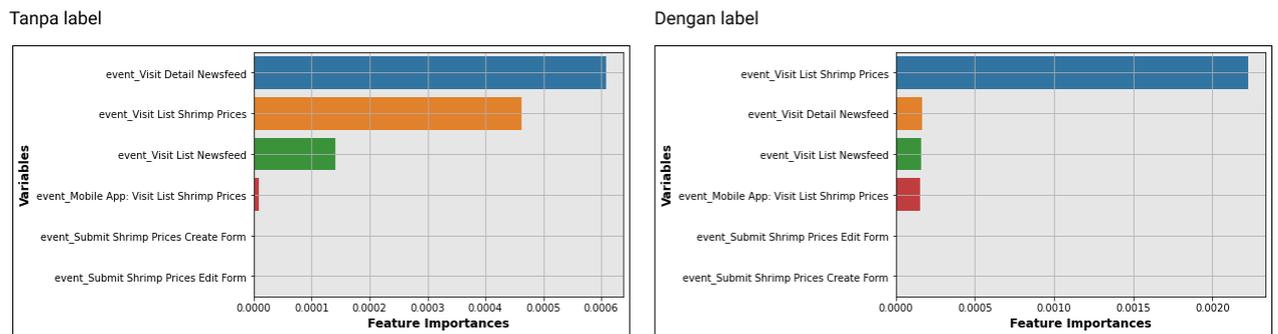
Berikut adalah pengelompokan setiap variabel pada Gambar 4.1 berdasarkan fiturnya.

- a. Pencatatan pakan: *feed_completion_mean, feed_data_count, feed_total, feed_average*.
- b. Pencatatan sampling: *shrimp_growth_average, shrimp_current_weight, sampling_data_average, sampling_data_count*
- c. Pencatatan kualitas air: *ph_data_count, temp_data_count, temp_completion_mean, salinity_data_count*.

Dari hasil ini menunjukkan bahwa intensitas penggunaan fitur pencatatan pakan, sampling, dan kualitas air akan berpengaruh signifikan terhadap keberlanjutan pengguna. Semakin tinggi intensitasnya, semakin besar kemungkinan pengguna untuk lanjut menggunakan Jala ke bulan-bulan berikutnya. Dengan demikian, kemudahan akses, keramahan tampilan, dan edukasi untuk menggunakan ketiga fitur ini pada bulan pertama sangat dianjurkan.

4.1.2 Analisis Fitur Media

Fitur media yang disediakan Jala ternyata belum berpengaruh signifikan terhadap keberlanjutan pengguna. Dari hasil penarikan nilai signifikansi variabel fitur media didapatkan bahwa rata-rata persentase pengaruh fitur media terhadap keberlanjutan pengguna bernilai 0.06%. Jika dibandingkan dengan 10 variabel teratas yang memiliki rata-rata di atas 5%, fitur media masih berpengaruh tidak signifikan terhadap keberlanjutan pengguna. Fitur media tersebut adalah sub fitur harga udang dan kabar udang. Sub fitur penyakit udang belum dimasukkan dalam analisis ini karena sub fitur penyakit udang baru di-*release* tahun 2021.



Gambar 4.2 Analisis fitur media

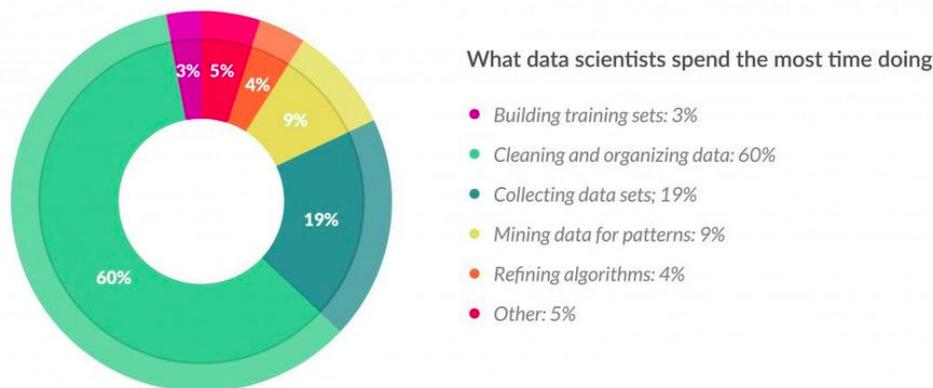
Gambar 4.2 menunjukkan pengaruh dari fitur-fitur tersebut. Berikut adalah penjelasan dari setiap variabel.

- even_Visit Detail Newsfeed*: Intensitas pengguna mengunjungi detail fitur media.
- event_Visit List Shrimp Prices*: Intensitas pengguna mengunjungi halaman daftar harga udang
- event_Visit List Newsfeed*: Intensitas pengguna mengunjungi halaman daftar media
- event_Mobile App Visit List Shrimp Prices*: Intensitas pengguna mengunjungi halaman daftar harga udang menggunakan *mobile apps*
- event_Submit Shrimp Prices Create Form*: Intensitas pengguna melakukan submit pembuatan form harga udang
- event_Submit Shrimp Prices Edit Form*: Intensitas pengguna melakukan submit perubahan form harga udang

Hasil ini menunjukkan bahwa tujuan dari pembuatan fitur media belum tercapai. Dengan demikian, diperlukan optimasi terhadap fitur ini. Mulai dari mengedukasi pengguna untuk menggunakannya, memperbaiki konten, hingga memberikan kemudahan akses terhadap fitur tersebut.

4.1.3 Pengalaman Melakukan *Preprocessing*

Preprocessing merupakan salah satu tahapan yang memegang peranan penting dalam *data science*. Kualitas *preprocessing* yang baik akan sangat membantu meningkatkan kualitas dari penerapan algoritma setelahnya, baik *clustering* maupun *modeling*. Selain itu, *preprocessing* merupakan tahapan yang paling banyak memakan waktu *data scientist*.



Gambar 4.3 Persentase durasi waktu yang digunakan *data scientist*

Sumber: (Press, 2016)

Gambar 4.3 menunjukkan persentase tugas-tugas berdasarkan durasi pengerjaan (Press, 2016). Data cleaning yang merupakan pekerjaan dalam tahap *preprocessing* menunjukkan angka yang paling dominan, yaitu 60% dari total waktu yang dihabiskan. Dengan demikian tahapan ini merupakan tahap yang sangat krusial untuk mendapatkan hasil yang maksimal.

Selama menjalani program magang, salah satu pengalaman yang sangat berharga adalah diberikan kepercayaan untuk melakukan *preprocessing* data di setiap proyek yang diberikan. Hal ini sangat membantu dalam mengembangkan kemampuan mengenal karakteristik data lebih detail. Mulai dari bagaimana menganalisis data *outlier*, melakukan visualisasi, cara mengisi *missing value*, hingga menerapkan standarisasi data. Selain itu, pengalaman menggunakan Pandas sebagai tools *preprocessing* yang populer merupakan suatu nilai lebih.

Pandas sangat memudahkan dalam *preprocessing* karena menerapkan konsep baris-kolom (tabel) yang dapat mudah dimengerti bahkan bagi pemula. Pandas juga menyediakan banyak *library* dengan komputasi yang sangat cepat dalam mengelola data dengan jumlah besar. Pengalaman-pengalaman ini sangat berharga, terutama dapat menjadi bekal untuk berkarier di dunia *data science*.

4.1.4 Pengalaman Menggunakan Algoritma K-Means

Sebagai sebuah algoritma yang umum digunakan dalam penerapan *clustering*, K-Means banyak digunakan untuk menyelesaikan permasalahan pengelompokan data dalam berbagai bidang. Salah satu contohnya adalah pengelompokan *customer*.

Salah satu studi literatur yang dilakukan adalah studi pengelompokan *mall customers* menggunakan algoritma *clustering* (Ishantha, 2021/2021, pp. 1–35). Variabel-variabel yang terlibat dalam literatur tersebut adalah variabel demografi, geografi, dan finansial dari setiap *customers*. Algoritma yang digunakan dalam proses *clustering* ini adalah *K-Means* dengan metode *Elbow Method* dalam menentukan nilai K optimum. Selain itu, dalam penerapannya literatur tersebut menggunakan bahasa pemrograman *python* sehingga sangat tepat untuk dijadikan referensi dalam pengerjaan proyek ini.

Hampir seluruh teori dan praktek dalam literatur tersebut sesuai dengan implementasi yang diterapkan oleh penulis dalam pengerjaan proyek ini. Akan tetapi, terdapat kendala yang dihadapi penulis pada saat menerapkan kode penentuan nilai K optimum. Dalam literatur (Ishantha, 2021/2021, pp. 1–35), nilai K ditentukan secara manual dengan melihat hasil dari penerapan *Elbow Method*. Konsep ini tidak tepat jika diterapkan di Jala. Hal ini disebabkan karena data perusahaan yang terus berubah dari hari ke hari. Akibatnya jenis dan karakteristik dari data juga berubah sehingga nilai K optimum yang ditentukan sebelumnya sangat mungkin tidak relevan dengan data yang baru. Oleh karena itu, berdasarkan permasalahan tersebut penulis mencoba menerapkan salah satu metode di *python* dari *library kneed* yang dapat digunakan untuk menentukan nilai K optimum secara dinamis berdasarkan penerapan *elbow method*. Metode ini belum diterapkan pada literatur yang dikaji oleh penulis sehingga hal ini menjadi salah satu pembelajaran yang didapatkan selama program magang.

4.2 Non Teknis

Secara non teknis terdapat banyak pelajaran yang didapatkan selama menjalani program magang. Beberapa diantaranya adalah sebagai berikut.

4.2.1 Manfaat Magang

Selama enam bulan melaksanakan kegiatan magang, banyak hal yang dipelajari. Dari sisi *hardskill*, salah satu pembelajaran yang paling berharga adalah mengetahui cara menganalisis data *outlier*, yaitu dengan cara memvisualisasikannya. Selain itu, kemampuan *query database* juga meningkat karena kegiatan magang mengharuskan *query* kompleks yang tidak pernah ditemukan selama masa kuliah.

Dari sisi kemampuan *softskill*, magang mengajarkan komunikasi secara terbuka dalam tim. Penulis belajar untuk menyampaikan seluruh kendala yang dihadapi, meskipun terlihat sepele, agar anggota tim yang lain tahu apa yang sedang terjadi sehingga tidak berpengaruh terhadap

pengerjaan proyek. Selain itu, presentasi proyek kepada stakeholder memberikan pengalaman *public speaking* yang berbeda karena harus mengomunikasikan konten terhadap beragam pendengar dengan latar belakang pendidikan yang berbeda-beda.

4.2.2 Hambatan dan Tantangan Magang

Hambatan yang dialami selama magang adalah komunikasi. Berlangsungnya magang dengan konsep WFH, membatasi komunikasi sehingga alur informasi menjadi lebih sulit. Keterbatasan komunikasi terbagi menjadi dua hal, yaitu tidak dapat mengakses informasi secara langsung dan gangguan koneksi menyebabkan diskusi tidak berjalan dengan maksimal.

Tantangan yang dihadapi selama magang adalah menyesuaikan kuliah dengan mengerjakan tugas magang. Kurikulum 2020 yang baru diterapkan informatika sedikit banyak memberikan tantangan bagi mahasiswa magang. Satu sisi penulis harus melaksanakan magang secara *full-time* (selama 8 jam), sementara di sisi lain harus mengikuti beberapa perkuliahan di jam kerja.

4.2.3 Kontribusi Selama Magang

Aktivitas magang yang dilakukan memberi kontribusi kepada Jala berupa penyediaan informasi baru. Seluruh proyek yang sudah diselesaikan ternyata mampu memberikan pandangan baru bagi pihak perusahaan sehingga banyak ide yang muncul didasari dari proyek yang sudah diselesaikan. Salah satunya adalah keinginan untuk mencari tahu seberapa besar pengaruh *mobile apps* terhadap keberlanjutan pengguna Jala.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari pengerjaan proyek Analisis Keberlanjutan Pengguna Jala, terdapat beberapa hal yang dapat disimpulkan, yaitu sebagai berikut.

- a. Tiga fitur utama yang mempengaruhi keberlanjutan pengguna ke bulan-bulan berikutnya adalah fitur pencatatan pakan, sampling, dan kualitas air.
- b. Fitur media masih belum berpengaruh signifikan terhadap keberlanjutan pengguna.
- c. Pandas merupakan tools yang tepat untuk melakukan *preprocessing* data karena banyak kemudahan yang ditawarkan, salah satunya adalah tersedianya fungsi perhitungan statistik yang cepat dan cocok untuk jumlah data besar.

5.2 Saran

Hasil dari penerapan proyek menunjukkan perlunya optimasi baik dari sisi *marketing* maupun kualitas produk agar dapat meningkatkan probabilitas keberlanjutan pengguna. Dari temuan tersebut, berikut adalah saran yang dapat dilakukan Jala untuk mencegah *churn* dan meningkatkan loyalitas pengguna.

- a. Mengedukasi pengguna baru untuk rutin menggunakan fitur pencatatan pakan, sampling, dan kualitas air.
- b. Mengoptimalkan publikasi fitur media, mulai dari sisi konten, kemudahan akses, ataupun edukasi pengguna untuk memanfaatkan fitur tersebut.
- c. Melanjutkan proyek ini ke variabel-variabel yang lain, salah satunya Analisis Keberlanjutan Pengguna dengan parameter perbandingan pengguna menggunakan *website* dan *mobile*.

DAFTAR PUSTAKA

- Agarwal, V. (2015). Research on Data Preprocessing and Categorization Technique for Smartphone Review Analysis. *International Journal of Computer Applications*, 131(4), 30–36. <https://doi.org/10.5120/ijca2015907309>
- Amruthnath, N., & Gupta, T. (2019). *Factor Analysis in Fault Diagnostics Using Random Forest*. April. <https://doi.org/10.4172/2169-0316.1000278>
- Arafat, Y. (2018). Penyelenggaraan Pembangunan NKRI Menuju Negara Maritim Berdasarkan Prinsip Negara Kepulauan. *Akta Yudisia*, 3(1), 1–23.
- Bakar, Z. A., Mohamad, R., Ahmad, A., & Deris, M. M. (2006). A comparative study for outlier detection techniques in data mining. *2006 IEEE Conference on Cybernetics and Intelligent Systems*. <https://doi.org/10.1109/ICCIS.2006.252287>
- Brownlee, J. (2020). *Difference Between Algorithm and Model in Machine Learning*. <https://machinelearningmastery.com/difference-between-algorithm-and-model-in-machine-learning/>
- Brownlee, J. (2020, June 9). How to Use StandardScaler and MinMaxScaler Transforms in Python. Machine Learning Mastery. <https://machinelearningmastery.com/standardscaler-and-minmaxscaler-transforms-in-python/>
- Carvalho, P., Hitzelberger, P., Otjacques, B., Bouali, F., & Venturini, G. (2015). Information visualization for CSV open data files structure analysis. *IVAPP 2015 - 6th International Conference on Information Visualization Theory and Applications; VISIGRAPP, Proceedings, March*, 101–108. <https://doi.org/10.5220/0005265301010108>
- Ghatak, D. (2021, February 25). What Is Factor Analysis in Data Science? [Review of What Is Factor Analysis in Data Science?]. Knowledgehut. <https://www.knowledgehut.com/blog/data-science/factor-analysis-data-science>
- Govindaraju, R., Simatupang, T., & Samadhi, T. A. (2008). Perancangan Sistem Prediksi Churn Pelanggan. *Teknik Informatika*, 9(1), 33–42. <http://puslit2.petra.ac.id/ejournal/index.php/inf/article/view/16893>
- Hoffman, C. (2018). *What Is a CSV File, and How Do I Open It?*

<https://www.howtogeek.com/348960/what-is-a-csv-file-and-how-do-i-open-it/>

- Hussein, A. R. H. (2019). Internet of Things (IOT): Research challenges and future applications. *International Journal of Advanced Computer Science and Applications*, 10(6), 77–82. <https://doi.org/10.14569/ijacsa.2019.0100611>
- Ishantha, A. (2021). (pp. 1–35) [Review of MALL CUSTOMER SEGMENTATION USING CLUSTERING ALGORITHM]. (Original work published 2021)
- Ishaq Bhatti, M., Awan, H. M., & Razaq, Z. (2013). The key performance indicators (KPIs) and their impact on overall organizational performance. *Quality & Quantity*, 48(6), 3127–3143. <https://doi.org/10.1007/s11135-013-9945-y>
- López del Val, J. A., & Alonso Pérez de Agreda, J. P. (1993). Principal components analysis. *Atencion Primaria / Sociedad Española de Medicina de Familia y Comunitaria*, 12(6), 333–338. <https://doi.org/10.5455/ijlr.20170415115235>
- Mahajan, G. (2019). *A step-by-step walkthrough of SQL Inner Join*. <https://www.sqlshack.com/a-step-by-step-walkthrough-of-sql-inner-join/>
- Nasution, A. P. D. (2017). *Analisis Perkembangan Ekspor Subsektor Perikanan di Indonesia*. 1–66. <https://core.ac.uk/download/pdf/225828041.pdf>
- Pedamkar, P. (2020). *SQL with Clause*. <https://www.educba.com/sql-with-clause/>
- Press, G. (2016). *Cleaning Big Data: Most Time-Consuming, Least Enjoyable Data Science Task, Survey Says*. <https://www.forbes.com/sites/gilpress/2016/03/23/data-preparation-most-time-consuming-least-enjoyable-data-science-task-survey-says/?sh=5d57ac966f63>
- Pulungan, R., Fauzia, L., & Emalisa. (2015). Analisis Kelayakan Usaha Tambak Udang. *JOURNAL ON SOCIAL ECONOMIC OF AGRICULTURE AND AGRIBUSINESS*, 1–12. <https://jurnal.usu.ac.id/index.php/ceress/article/view/14431/6342>
- Python, R. (n.d.). The Python pickle Module: How to Persist Objects in Python – Real Python. [Realpython.com. https://realpython.com/python-pickle-module/](https://realpython.com/python-pickle-module/)
- Satopaa, V., Albrecht, J., Irwin, D., & Raghavan, B. (n.d.). [Review of Finding a “Kneedle” in a Haystack: Detecting Knee Points in System Behavior]. (1-6).
- Satoto, K. I., Isnanto, R. R., Kridalukmana, R., & Martono, K. T. (2017). Optimizing MySQL database system on information systems research, publications and community service.

Proceedings - 2016 3rd International Conference on Information Technology, Computer, and Electrical Engineering, ICITACEE 2016, 1–5.
<https://doi.org/10.1109/ICITACEE.2016.7892476>

Shi, N., Liu, X., & Guan, Y. (2010). Research on k-means clustering algorithm: An improved k-means clustering algorithm. *3rd International Symposium on Intelligent Information Technology and Security Informatics, IITSI 2010, 63–67.*
<https://doi.org/10.1109/IITSI.2010.74>

Supono. (2006). [Review of PRODUKTIVITAS UDANG PUTIH PADA TAMBAK INTENSIF DI TULANG BAWANG LAMPUNG]. *Jurnal Saintek Perikanan, Vol. 2(No. 1), 48–53.*

Taylor, C. (2018). *Understanding Quantiles: Definitions and Uses.*
<https://www.thoughtco.com/what-is-a-quantile-3126239>

Trimarsiah, Y., & Arafat, M. (2017). ANALISIS DAN PERANCANGAN WEBSITE SEBAGAI SARANA INFORMASI PADA LEMBAGA BAHASA KEWIRAUSAHAAN DAN KOMPUTER AKMI BATURAJA [Review of ANALISIS DAN PERANCANGAN WEBSITE SEBAGAI SARANA INFORMASI PADA LEMBAGA BAHASA KEWIRAUSAHAAN DAN KOMPUTER AKMI BATURAJA]. *Jurnal Ilmiah Matrik, 19(1), 1–10.*

Witoko, P., Purbosari, N., Noor, N. M., Puji, D., Barades, E., & Bokau, J. (2018). Budidaya Udang Vaname (*Litopenaeus vannamei*) di Keramba Jaring Apung Laut Culture of White Shrimp (*Litopenaeus vannamei*) at Sea Floating Net. *Budidaya Udang Vaname (Litopenaeus Vannamei) Di Keramba Jaring Apung Laut Culture, 410–418.*

Yue, T. (2016). How To Write a Good. *Case Development Centre Rotterdam School of Management Erasmus University, 1–9.*

LAMPIRAN

Sertifikat program magang dari perusahaan Jala





MAGANG:
006/III/JALA/2021

SERTIFIKAT

Andri Wahyu Ahmad R.

Mahasiswa Program Studi Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia dengan NIM 17523221,
telah menyelesaikan magang selama periode 1 Oktober 2020 - 31 Maret 2021 dengan hasil BAIK.
Yang bersangkutan telah memiliki kemampuan data analysis dengan python, pembuatan dashboard analytics dengan Google Data Studio, aplikasi algoritma machine learning klasik dengan scikit-learn, dan ekstraksi data dengan SQL.

Demikian sertifikat ini dikeluarkan untuk dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 31 Maret 2021



LIRIS MADUNINGTYAS
CHIEF EXECUTIVE OFFICER
JALA TECH

Head Office :
Jl. Kaliurang KM 16,3
Yogyakarta 55584, Indonesia
T. +62 274 2874515
F. +62 274 898269



Catatan pekerjaan program magang selama enam bulan

| No | Minggu ke- | Tanggal | Aktivitas |
|----------------|------------|------------|--|
| OKTOBER | | | |
| 1 | I | 01/10/2020 | Onboarding perusahaan |
| 2 | | 02/10/2020 | Onboarding <i>data scientist</i> |
| 3 | II | 05/10/2020 | <i>Weekly meeting</i> Perkenalan kepada karyawan lain Akses database JALA |
| 4 | | 06/10/2020 | Instalasi ubuntu 20.04 Instalasi tools untuk akses database |
| 5 | | 07/10/2020 | Diskusi mengenai database bersama supervisor Pelajari struktur database JALA |
| 6 | | 08/10/2020 | Latihan query database menggunakan mysql |
| 7 | | 09/10/2020 | Mengerjakan latihan query database dan diskusi hasil latihan <i>Weekly meeting</i> |
| 8 | III | 12/10/2020 | <i>Weekly meeting</i> Mengerjakan <i>challenge #1 query database</i> |
| 9 | | 13/10/2020 | Mengerjakan <i>challenge #2</i> , yaitu query database untuk mendapatkan informasi detail dari setiap siklus |
| 10 | | 14/10/2020 | Pindah ke <i>challenge #3 dan #4</i> , yaitu ekstraksi data dari database untuk mendapatkan data kualitas air dan pakan dari tambak-tambak yang menggunakan JALA |
| 11 | | 15/10/2020 | <i>Challenge #5 & #6</i> mengerjakan query untuk mengambil data Sampling (pengambilan udang untuk dilakukan pengukuran pertumbuhannya) dan <i>Treatment</i> |
| 12 | | 16/10/2020 | <i>Challenge #6</i> bertujuan untuk mengambil data <i>treatment</i> udang |
| 13 | | IV | 19/10/2020 |
| 14 | 20/10/2020 | | Mengerjakan <i>challenge #7</i> , yaitu mengambil data panen |
| 15 | 21/10/2020 | | Project 1: Analisis Efisiensi Tambak Query database untuk mengambil data efisiensi |
| 16 | 22/10/2020 | | Project 1: Analisis Efisiensi Tambak Diskusi mengenai hasil query bersama supervisor. Sekaligus melakukan query kembali dari hasil revisi |
| 17 | 23/10/2020 | | Umum Diskusi mengenai project baru yang akan di kick-off oleh JALA, yaitu project Disease Mapping. <i>Weekly meetings</i> |
| 18 | 26/10/2020 | | Project 1: Analisis Efisiensi Tambak Analisis hasil query efisiensi tambak. Memastikan query sudah dilakukan dengan benar. |
| 19 | 27/10/2020 | | Project 1: Analisis Efisiensi Tambak |

| | | | |
|-----------------|------------|------------|--|
| | | | Diskusi hasil query bersama supervisor Pengenalan Pandas |
| NOVEMBER | | | |
| 20 | I | 02/11/2020 | Project 1: Analisis Efisiensi Tambak Belajar struktur syntax Pandas |
| 21 | | 03/11/2020 | Project 1: Analisis Efisiensi Tambak Belajar menggabungkan tabel Pandas |
| 22 | | 04/11/2020 | Project 1: Analisis Efisiensi Tambak Menggabungkan data efisiesi yang masih terpisah-pisah menjadi satu tabel menggunakan pandas |
| 23 | | 05/11/2020 | Project 1: Analisis Efisiensi Tambak Mengisi dan membuang data yang tidak layak menggunakan Pandas |
| 24 | | 06/11/2020 | Project 1: Analisis Efisiensi Tambak Revisi hasil preprocessing |
| 25 | II | 09/11/2020 | Project 1: Analisis Efisiensi Tambak Belajar penggunaan Google Data Studio <i>Weekly Meetings</i> |
| 26 | | 10/11/2020 | Project 1: Analisis Efisiensi Tambak Belajar penerapan Google Data Studio |
| 27 | | 11/11/2020 | Project 1: Analisis Efisiensi Tambak Implementasi data hasil preprocessing ke Google Data Studio tahap 1 |
| 28 | | 12/11/2020 | Project 1: Analisis Efisiensi Tambak Implementasi data hasil preprocessing ke Google Data Studio tahap 1 |
| 29 | | 13/11/2020 | Project 1: Analisis Efisiensi Tambak Implementasi data hasil preprocessing ke Google Data Studio tahap 1 |
| 30 | III | 16/11/2020 | Project 1: Analisis Efisiensi Tambak Melakukan query database lagi karena masih terdapat data yang salah Preprocessing data tahap 2 |
| 31 | | 17/11/2020 | Project 1: Analisis Efisiensi Tambak Preprocessing data tahap 2 |
| 32 | | 18/11/2020 | Project 1: Analisis Efisiensi Tambak Diskusi hasil preprocessing tahap 2 |
| 33 | | 19/11/2020 | Project 1: Analisis Efisiensi Tambak Analisis data outlier hasil dari preprocessing tahap 2 |
| 34 | | 20/11/2020 | Project 1: Analisis Efisiensi Tambak Implementasi filter data outlier |

| | | | |
|-----------------|----|------------|---|
| 35 | IV | 23/11/2020 | Project 1: Analisis Efisiensi Tambak <i>Weekly meetings</i> Membuat halaman pertama dashboard efisiensi tambak |
| 36 | | 24/11/2020 | Project 1: Analisis Efisiensi Tambak Desain tampilan dashboard halaman pertama |
| 37 | | 25/11/2020 | Project 1: Analisis Efisiensi Tambak Implementasi dashboard halaman kedua |
| 38 | | 26/11/2020 | Project 1: Analisis Efisiensi Tambak Desain tampilan dashboard halaman kedua |
| 39 | | 27/11/2020 | Project 1: Analisis Efisiensi Tambak Integrasi halaman satu dan dua dari Google Data Studio <i>Weekly meetings</i> |
| 40 | | 30/11/2020 | Project 1: Analisis Efisiensi Tambak Weekly meetings Presentasi hasil project. |
| DESEMBER | | | |
| 41 | I | 01/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna Query data pengguna dan diskusi bersama supervisor |
| 42 | | 02/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna Implementasi penggabungan data siklus dan kolam menggunakan pandas |
| 43 | | 03/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna Integrasi data hasil preprocessing sebelumnya dengan informasi detail tambak |
| 44 | | 04/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna Menganalisis cara untuk mengisi data yang kosong agar tidak banyak data yang hilang <i>Weekly meetings</i> |
| 45 | II | 07/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna <i>Weekly Meetings</i> Diskusi detail project bersama supervisor |
| 46 | | 08/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna Analisis setiap variabel untuk melihat data yang outlier |
| 47 | | 09/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna Implementasi coding cohort analisis pada sampel data |
| 48 | | 10/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna Implementasi coding cohort analysis versi 1 terhadap data tambak |
| 49 | | 11/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna |

| | | | |
|----------------|------------|------------|---|
| | | | Visualisasi cohort analysis pada data dengan kelengkapan (50-75%) Visualisasi cohort analysis pada data dengan kelengkapan (50-100%) |
| 50 | III | 14/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna <i>Weekly Meetings</i> Diskusi hasil penerapan cohort analisis bersama supervisor |
| 51 | | 15/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna Kajian literatur penerapan algoritma K-Means |
| 52 | IV | 21/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna Implementasi algoritma K-Means menggunakan data sampel |
| 53 | | 22/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna Diskusi implementasi K-Means Visualisasi hasil clustering versi 1 |
| 54 | | 23/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna Daily meetings Penerapan PCA (Principal Component Anlysis) pada data |
| 55 | | 28/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna <i>Weekly meetings</i> Analisis data menggunakan Whisker & box plot |
| 56 | | 29/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna <i>Daily Meetings</i> Analisis persebaran data seluruh variabel menggunakan whisker & box plot |
| 57 | | 30/12/2020 | Project 2: Factor Analisis Keberlanjutan Pengguna Implementasi PCA versi 2 |
| JANUARI | | | |
| 58 | I | 04/01/2021 | Project 2: Factor Analisis Keberlanjutan Pengguna Analisis data outlier |
| 59 | | 05/01/2021 | Project 2: Factor Analisis Keberlanjutan Pengguna <i>Plotting</i> data cohort berdasarkan tahun dan bulan registrasi |
| 60 | | 06/01/2021 | Project 2: Factor Analisis Keberlanjutan Pengguna Ekstrak kelengkapan data pakan setelah dua bulan pertama penggunaan JALA |
| 61 | | 07/01/2021 | Project 2: Factor Analisis Keberlanjutan Pengguna Analisis data outlier menggunakan histogram |
| 62 | | 08/01/2021 | Project 2: Factor Analisis Keberlanjutan Pengguna Analisis setiap variabel untuk melihat data yang outlier |
| 63 | II | 11/01/2021 | Project 2: Factor Analisis Keberlanjutan Pengguna <i>Weekly meetings</i> Mencoba metode quantile untuk filter data outlier |

| | | | |
|-----------------|------------|------------|---|
| 64 | | 12/01/2021 | Project 2: Factor Analisis Keberlanjutan Pengguna Implementasi code menghilangkan data outlier |
| 65 | | 13/01/2021 | Project 2: Factor Analisis Keberlanjutan Pengguna Visualisasi distribusi data hasil filter |
| 66 | | 14/01/2021 | Project 2: Factor Analisis Keberlanjutan Pengguna Diskusi dan analisis distribusi data |
| 67 | | 15/01/2021 | Project 2: Factor Analisis Keberlanjutan Pengguna Implementasi Factor Analysis menggunakan algoritma random forest |
| 68 | III | 18/01/2021 | Project 2: Factor Analisis Keberlanjutan Pengguna Analisis hasil |
| 69 | | 20/01/2021 | Project 2: Factor Analisis Keberlanjutan Pengguna Analisis Top 4 Faktor keberlanjutan Pengguna |
| 70 | | 21/01/2021 | Project 2: Factor Analisis Keberlanjutan Pengguna Pembuatan laporan project |
| 71 | | 22/01/2021 | Project 2: Factor Analisis Keberlanjutan Pengguna <i>Weekly meetings</i> Presentasi project |
| 72 | IV | 25/01/2021 | Diskusi tugas selanjutnya |
| 73 | | 27/01/2021 | Task: Analisis Pengaruh Aktivitas Marketing dengan Penggunaan Jala Analisis google sheets <i>marketing</i> |
| 74 | | 28/01/2021 | Task: Analisis Pengaruh Aktivitas Marketing dengan Penggunaan Jala Ekstrak data siklus pro |
| 75 | | 29/01/2021 | Diskusi perubahan prioritas <i>task</i> <i>Weekly Meetings</i> |
| FEBRUARI | | | |
| 76 | I | 01/02/2021 | Task: Evaluasi Akurasi Prediksi (KPI) <i>Weekly Meetings</i> Ekstrak data ABW Ekstrak data ADG |
| 77 | | 02/02/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Ekstrak data feeds Ekstrak data tonase |
| 78 | | 03/02/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Ekstrak data FCR Ekstrak data SR Diskusi mengenai kelengkapan data prediksi |
| 79 | | 04/02/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Pemisahan antara data baru dan data lama |

| | | | |
|--------------|------------|------------|---|
| 80 | | 05/02/2021 | Task: Evaluasi Akurasi Prediksi (KPI) <i>Weekly Meetings</i> Merge data prediksi dan data aktual |
| 81 | II | 08/02/2021 | Task: Evaluasi Akurasi Prediksi (KPI) <i>Weekly Meetings</i> Hitung R2 score Plotting error value ABW |
| 82 | | 09/02/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Hitung R2 score Plotting error value ADG dan SR |
| 83 | | 10/02/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Hitung nilai FCR <i>Tunning</i> nilai ADG dan SR |
| 84 | | 11/02/2021 | Task: Evaluasi Akurasi Prediksi (KPI) <i>Tunning</i> nilai tonase Update data prediksi |
| 85 | | 15/02/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Update data prediksi <i>Tunning</i> nilai sampling <i>Weekly Meetings</i> |
| 86 | III | 16/02/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Diskusi tentang kondisi data Plotting nilai regresi dari setiap variabel |
| 87 | | 17/02/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Hitung nilai MAPE (<i>Mean Average Percentage Error</i>) |
| 88 | | 18/02/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Terapkan <i>filtering</i> data berdasarkan nilai <i>quantile</i> |
| 89 | | 19/02/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Diskusi konten <i>google data studio</i> |
| 90 | IV | 22/02/2021 | Task: Evaluasi Akurasi Prediksi (KPI) <i>Tunning</i> data prediksi |
| 91 | | 23/02/2021 | Task: Evaluasi Akurasi Prediksi (KPI) <i>Connecting</i> data ke <i>google data studio</i> |
| MARET | | | |
| 97 | I | 03/03/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Revisi data SR |
| 98 | | 04/03/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Update data <i>google data studio</i> sesuai hasil revisi |
| 99 | | 05/03/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Buat <i>google sheets</i> baru untuk data yang telah diperbaiki |

| | | | |
|-----|-----|------------|---|
| 100 | II | 08/03/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Menambahkan halaman baru pada google data studio untuk data hasil perbaikan |
| 101 | | 09/03/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Diskusi mengenai R2 <i>score</i> dan MAPE |
| 102 | | 10/03/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Perbaiki struktur google data studio |
| 103 | | 12/03/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Menjelaskan hasil pengerjaan kepada Supervisor <i>Weekly Meetings</i> |
| 104 | III | 15/03/2021 | Task: Evaluasi Akurasi Prediksi (KPI) <i>Weekly Meetings</i> Diskusi mengenai topik TA |
| 105 | | 16/03/2021 | Task: Evaluasi Akurasi Prediksi (KPI) <i>Update</i> tampilan google data studio |
| 106 | | 17/03/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Diskusi mengenai topik TA bersama CTO |
| 107 | | 18/03/2021 | Task: Evaluasi Akurasi Prediksi (KPI) Finalisasi tampilan google data studio |
| 108 | | 19/03/2021 | <i>Weekly Meetings</i> |
| 109 | IV | 22/03/2021 | <i>Weekly Meetings</i> |
| 110 | | 23/03/2021 | Task: Tunning model SR Diskusi mengenai <i>tunning</i> model SR |
| 111 | | 24/03/2021 | Task: Tunning model SR Baca code training model <i>Running training</i> model Random Forest |
| 112 | | 25/03/2021 | Task: Tunning model SR Diskusi hasil model <i>Random Forest</i> |
| 113 | | 26/03/2021 | Task: Tunning model SR Buat code untuk split hasil <i>training</i> |
| 114 | | 29/03/2021 | Task: Tunning model SR <i>Retraining</i> model <i>Random Forest</i> |
| 115 | | 30/03/2021 | Task: Tunning model SR <i>Training</i> model <i>Gradient Boosting</i> |
| 116 | | 31/03/2021 | Task: Tunning model SR <i>Training</i> model <i>MLP</i> |