



KERJA PRAKTIK - EF234603

Migrasi Platform Visualisasi dan Implementasi RFM serta Cohort Analysis untuk Mendukung Analisis Data E-Commerce PT Paragon Technology and Innovation

PT Paragon Technology and Innovation Jl. Kp. Baru IV No.4,
RT.5/RW.2, Ulujami, Kec. Pesanggrahan, Kota Jakarta Selatan,
Daerah Khusus Ibukota Jakarta 12250

Periode: 1 Juli 2024 - 30 September 2024

Oleh:

Anneu Tsabita Putri 5025211026

Dewangga Dika Darmawan 5025211109

Pembimbing Jurusan

Dr. Baskoro Adi P., S.Kom., M.Kom.

Pembimbing Lapangan

Adji Gunhardi, S.T.

DEPARTEMEN TEKNIK INFORMATIKA

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Surabaya 2024



KERJA PRAKTIK - EF234603

**Migrasi Platform Visualisasi dan Implementasi RFM serta
Cohort Analysis untuk Mendukung Analisis Data
E-Commerce PT Paragon Technology and Innovation**

PT Paragon Technology and Innovation Jl. Kp. Baru IV No.4,
RT.5/RW.2, Ulujami, Kec. Pesanggrahan, Kota Jakarta Selatan, Daerah
Khusus Ibukota Jakarta 12250

Periode: 1 Juli 2024 - 30 September 2024

Oleh:

Anneu Tsabita Putri 5025211026

Dewangga Dika Darmawan 5025211109

Pembimbing Jurusan

Dr. Baskoro Adi P., S.Kom., M.Kom.

Pembimbing Lapangan

Adji Gunhardi, S.T.

DEPARTEMEN TEKNIK INFORMATIKA
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2024

[Halaman ini sengaja dikosongkan]

DAFTAR ISI

DAFTAR ISI.....	iv
DAFTAR GAMBAR	ix
DAFTAR TABEL	xi
LEMBAR PENGESAHAN.....	xiii
KATA PENGANTAR.....	xviii
BAB I PENDAHULUAN	1
1.1. Latar Belakang.....	1
1.2. Tujuan.....	2
1.3. Manfaat.....	3
1.4. Rumusan Masalah	3
1.5. Lokasi dan Waktu Kerja Praktik	4
1.6. Metodologi Kerja Praktik	4
1.6.1. Perumusan Masalah	4
1.6.2. Studi Literatur	5
1.6.3. Analisis dan Perancangan Sistem	5
1.6.4. Implementasi Sistem.....	6
1.6.5. Pengujian dan Evaluasi.....	6
1.6.6. Kesimpulan dan Saran	7
1.7. Sistematika Laporan	7
1.7.1. Bab I Pendahuluan	7
1.7.2. Bab II Profil Perusahaan	7
1.7.3. Bab III Tinjauan Pustaka	7

1.7.4.	Bab IV Analisis dan Perancangan Infrastruktur Sistem	7
1.7.5.	Bab V Implementasi Sistem.....	7
1.7.6.	Bab VI Pengujian dan Evaluasi	8
1.7.7.	Bab VII Kesimpulan dan Saran	8
BAB II PROFIL PERUSAHAAN		9
2.1.	Profil PT Paragon Technology and Innovation	9
2.2.	Visi dan Misi	10
2.2.1.	Visi.....	10
2.2.2.	Misi	10
2.3.	Lokasi	11
BAB III TINJAUAN PUSTAKA.....		13
3.1.	Platform Visualisasi.....	13
3.2.	Server <i>On-premise</i>	14
3.3.	<i>Containerization Platform</i>	14
3.4.	<i>Deployment Orchestrator</i>	15
3.5.	<i>Deployment Orchestrator Packaging Tool</i>	16
3.6.	<i>Cloud-based Relational Database</i>	16
3.7.	SQL	17
3.8.	Data Warehouse dan Data Mart	18
3.9.	ETL.....	19
3.10.	<i>RFM Analysis</i>	19
3.11.	<i>Cohort Analysis</i>	21

3.12.	<i>Interquartile Range</i>	22
BAB IV ANALISIS DAN PERANCANGAN		
INFRASTRUKTUR SISTEM.....		
4.1.	Analisis Sistem	25
4.1.1.	Deskripsi Sistem	25
4.2.	Perancangan Infrastruktur Sistem.....	26
4.2.1.	Arsitektur Sistem	26
4.2.2.	Tahapan Migrasi	27
4.2.3.	Desain Sistem Ekstraksi.....	28
4.2.4.	Desain Sistem Transformasi	30
4.2.5.	Desain Sistem Analisis	34
BAB V IMPLEMENTASI SISTEM		
5.1.	<i>Dump Database</i>	38
5.1.1.	Identifikasi <i>Database</i>	38
5.1.2.	Melakukan <i>Dump Database</i>	39
5.2.	<i>Restore Database</i>	39
5.3.	Deploy Platform Visualisasi	40
5.4.	Implementasi Sistem Ekstraksi.....	41
5.5.	Implementasi Sistem Transformasi	41
5.5.1.	<i>Monthly Aggregated</i>	42
5.5.2.	<i>Final User Analysis</i>	43
5.6.	Implementasi Sistem Analisis	43
5.6.1.	<i>RFM Analysis</i>	44

5.6.2. Cohort Analysis.....	45
BAB VI PENGUJIAN DAN EVALUASI.....	47
6.1. Tujuan Pengujian.....	47
6.2. Kriteria Pengujian.....	47
6.3. Skenario Pengujian.....	48
6.4. Evaluasi Pengujian	49
BAB VII KESIMPULAN DAN SARAN	51
7.1. Kesimpulan.....	51
7.2. Saran.....	52
DAFTAR PUSTAKA.....	53
BIODATA PENULIS 1.....	56
BIODATA PENULIS 2.....	56

[Halaman ini sengaja dikosongkan]

DAFTAR GAMBAR

Gambar 2.1 Logo PT Paragon Technology and Innovation.....	9
Gambar 3.1 Model Piramida	21
Gambar 4.1 Arsitektur Platform Visualisasi Sebelum dan Setelah Migrasi	26
Gambar 4.2 Alur Tahapan Migrasi.....	27
Gambar 4.3 Data Lineage.....	30
Gambar 5.1 Diagram alir sistem ekstraksi	41
Gambar 5.2 Diagram Alir Sistem Transformasi Awal	42
Gambar 5.3 Diagram Alir Sistem Transformasi Final	43
Gambar 5.4 Diagram Alir Sistem Visualisasi <i>RFM Analysis</i>	44
Gambar 5.5 <i>Preview</i> Hasil Visualisasi <i>RFM Analysis</i>	45
Gambar 5.6 Diagram Alir Sistem Visualisasi <i>Cohort Analysis</i> ...	46

[Halaman ini sengaja dikosongkan]

DAFTAR TABEL

Tabel 4.1 Katalog data yang diekstrak.....	29
Tabel 4.2 Katalog data pada tabel ‘aggregated_monthly’	31
Tabel 4.3 Katalog data untuk visualisasi RFM analysis.....	35
Tabel 4.4 Katalog data untuk visualisasi cohort analysis	36
Tabel 6.1 Hasil Evaluasi Pengujian.....	49

[Halaman ini sengaja dikosongkan]

LEMBAR PENGESAHAN
KERJA PRAKTIK

**Migrasi Platform Visualisasi dan Implementasi RFM
serta Cohort Analysis untuk Mendukung Analisis Data
E-Commerce PT Paragon Technology and Innovation**

Oleh:

Anneu Tsabita Putri

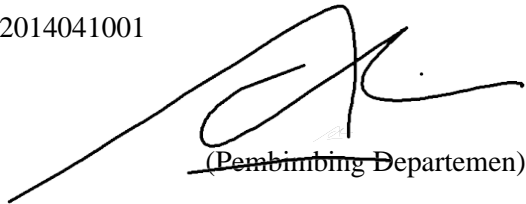
5025211026

Dewangga Dika Darmawan

5025211109

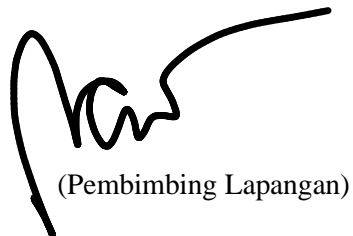
Disetujui oleh Pembimbing Kerja Praktik:

1. Dr. Baskoro Adi P., S.Kom.,
M.Kom.
NIP. 198702182014041001



(Pembimbing Departemen)

2. Adji Gunhardi, S.T.



(Pembimbing Lapangan)

[Halaman ini sengaja dikosongkan]

Migrasi Platform Visualisasi dan Implementasi RFM serta Cohort Analysis untuk Mendukung Analisis Data E-Commerce PT Paragon Technology and Innovation

Nama Mahasiswa : Anneu Tsabita Putri
NRP : 5025211026
Nama Mahasiswa : Dewangga Dika Darmawan
NRP : 5025211109
Departemen : Teknik Informatika FTEIC-ITS
Pembimbing Departemen : Dr. Baskoro Adi P., S.Kom.,
M.Kom.
Pembimbing Lapangan : Adji Gunhardi, S.T.

ABSTRAK

Seiring dengan meningkatnya kebutuhan perusahaan dalam mengelola data dan analisis untuk mendukung pengambilan keputusan, platform visualisasi data menjadi alat penting, termasuk bagi PT Paragon Technology and Innovation. Dalam rangka mendukung pengembangan bisnis di sektor E-Commerce, perusahaan ini membutuhkan analisis mendalam terhadap pola berbelanja pelanggan melalui implementasi RFM analysis dan cohort analysis. Namun, dengan meningkatnya volume data dan pengguna, platform visualisasi data yang berbasis server on-premise menghadapi tantangan dalam hal skalabilitas dan kinerja. Proyek ini bertujuan untuk mengatasi tantangan tersebut dengan memigrasikan platform visualisasi ke lingkungan deployment orchestrator. Dengan pendekatan ini, platform dapat menerapkan auto-scaling dan pemulihan otomatis terhadap container, serta memindahkan database ke layanan cloud-based relational database untuk meningkatkan performa. Hasil implementasi diharapkan mampu mendukung perusahaan dalam mengambil

keputusan berbasis data secara efisien, sekaligus mengoptimalkan pengelolaan platform visualisasi untuk mendukung pengembangan bisnis E-Commerce.

Kata Kunci : Platform Visualisasi, E-Commerce, RFM Analysis, Cohort analysis, Deployment Orchestrator

[Halaman ini sengaja dikosongkan]

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah SWT atas penyertaan dan karunia-Nya sehingga penulis dapat menyelesaikan salah satu kewajiban penulis sebagai mahasiswa Departemen Teknik Informatika ITS yaitu Kerja Praktik yang berjudul: Migrasi Platform Visualisasi dan Implementasi RFM serta Cohort Analysis untuk Mendukung Analisis Data E-Commerce PT Paragon Technology and Innovation.

Penulis menyadari bahwa masih banyak kekurangan baik dalam melaksanakan kerja praktik maupun penyusunan buku laporan kerja praktik ini. Namun penulis berharap buku laporan ini dapat menambah wawasan pembaca dan dapat menjadi sumber referensi.

Melalui buku laporan ini penulis juga ingin menyampaikan rasa terima kasih kepada orang-orang yang telah membantu menyusun laporan kerja praktik baik secara langsung maupun tidak langsung antara lain:

1. Penulis.
2. Bapak Dr. Baskoro Adi P., S.Kom., M.Kom. selaku dosen pembimbing kerja praktik sekaligus koordinator kerja praktik.
3. Bapak Adji Gunhardi, S.T. selaku pembimbing lapangan selama kerja praktik berlangsung.
4. Teman-teman penulis yang senantiasa memberikan semangat ketika penulis melaksanakan KP.

Surabaya, 19 Desember 2024

Anneu Tsabita Putri dan Dewangga Dika Darmawan

[Halaman ini sengaja dikosongkan]

BAB I

PENDAHULUAN

1.1. Latar Belakang

Seiring dengan berkembangnya kebutuhan untuk mengelola data dan analisis yang lebih efisien, perusahaan semakin bergantung pada penggunaan platform visualisasi data untuk mendukung proses pengambilan keputusan. PT Paragon Technology and Innovation, sebagai salah satu perusahaan manufaktur kosmetik nasional terbesar di Indonesia, telah memanfaatkan platform visualisasi data untuk menganalisis dan mendukung keputusan bisnis. Seiring dengan perubahan zaman dan meningkatnya kebutuhan akan layanan belanja *online*, perusahaan ini juga mengembangkan bisnisnya ke sektor *E-Commerce*. Dalam pengembangan bisnis tersebut, analisis data *E-Commerce* sangat penting, terutama untuk menganalisis pola berbelanja pelanggan dan memprediksi kebutuhan masa depan.

Salah satu informasi yang dapat digunakan adalah data berbelanja pelanggan, yang menunjukkan produk apa saja yang diminati pelanggan, seberapa loyal pelanggan tersebut, dan pola berbelanja pelanggan untuk memprediksi kebutuhan atau perilaku di masa depan, seperti musim belanja tertentu. Oleh karena itu, pada kerja praktik ini, penulis melakukan implementasi *RFM analysis* dan *cohort analysis* untuk menganalisis data belanja *customer* menggunakan platform visualisasi pada mitra *E-commerce* PT Paragon Technology and Innovation.

Namun, dengan semakin banyaknya pengguna dan meningkatnya volume data, pengelolaan platform visualisasi data dapat menghadapi berbagai tantangan, terutama dalam hal skalabilitas dan kinerja. Seiring dengan itu, dalam menghadapi lonjakan pengguna dan *traffic*, pengelolaan platform ini harus dioptimalkan. Oleh karena itu, proyek kerja praktik ini juga bertujuan untuk

mengatasi tantangan tersebut dengan memigrasikan platform visualisasi dari server *on-premise* ke dalam lingkungan *deployment orchestrator* yang dimiliki oleh PT Paragon Technology and Innovation.

Sebelumnya, platform visualisasi yang ada saat ini, di-*deploy* menggunakan *containerization platform* pada server *on-premise*, di mana setelah proses *deployment* terdapat beberapa *container* yang menjalankan fungsi spesifik untuk mendukung operasional platform tersebut. Ketika jumlah pengguna meningkat atau terjadi lonjakan *traffic*, beberapa *container* sering mengalami kelebihan beban atau bahkan berhenti bekerja. Untuk memperbaiki hal tersebut diperlukan intervensi manual. Namun, dengan menggunakan *deployment orchestrator*, platform visualisasi dapat secara otomatis menghidupkan kembali *container* yang berhenti bekerja dan menerapkan mekanisme *auto-scaling*, memungkinkan platform menangani beban lebih besar secara efisien. Selain itu, *database* yang digunakan oleh platform visualisasi akan dipindahkan ke layanan *cloud-based relational database* agar terpisah dari lingkungan *deployment orchestrator*. Hal tersebut bertujuan untuk menjaga performa platform tetap optimal tanpa membebani sumber daya *deployment orchestrator*.

Dengan langkah ini, diharapkan PT Paragon Technology and Innovation dapat terus mengembangkan bisnis *E-Commerce* dan mengambil keputusan yang lebih cerdas berdasarkan analisis data yang lebih efisien dan terkelola dengan baik.

1.2. Tujuan

Tujuan kerja praktik ini adalah untuk memenuhi kewajiban akademik dengan memperoleh nilai kerja praktik sebesar 4 SKS serta membantu PT Paragon Technology and Innovation dalam menghasilkan *insight*

yang bermanfaat untuk merancang strategi pemasaran, meningkatkan loyalitas pelanggan, dan memaksimalkan potensi pasar perusahaan melalui penggunaan platform visualisasi yang telah dimigrasikan dari server *on-premise* ke lingkungan *deployment orchestrator*.

1.3. Manfaat

Manfaat dari kerja praktik ini adalah membantu PT Paragon Technology and Innovation dalam mengumpulkan data perilaku pelanggan *E-Commerce* untuk keperluan analisis dan menghasilkan *insight* yang dapat digunakan sebagai dasar pengambilan keputusan bisnis di masa depan. Selain itu, kerja praktik ini membantu menemukan solusi untuk melakukan migrasi infrastruktur platform visualisasi ke dalam lingkungan *deployment orchestrator* guna mengatasi masalah kelebihan beban pada server *on-premise*. Di sisi lain, kerja praktik ini turut meningkatkan pemahaman dan kemampuan dalam pemecahan masalah terkait proses migrasi platform dari satu lingkungan infrastruktur ke lingkungan yang lain.

1.4. Rumusan Masalah

Rumusan masalah dari kerja praktik ini adalah sebagai berikut:

1. Bagaimana memproses data yang diambil dari beberapa sumber menjadi data yang digunakan untuk analisis?
2. Bagaimana menganalisis perilaku pelanggan *E-Commerce*?
3. Bagaimana memastikan sistem dapat menangkap perubahan data terbaru secara otomatis tanpa adanya inkonsistensi?
4. Bagaimana cara melakukan migrasi platform visualisasi dari server *on-premise* ke dalam lingkungan *deployment orchestrator*?
5. Bagaimana cara memastikan data pada platform visualisasi di *deployment orchestrator* tetap konsisten dan

sesuai dengan data yang ada di server *on-premise* sehingga siap untuk digunakan?

1.5. Lokasi dan Waktu Kerja Praktik

Kerja praktik dilaksanakan di PT Paragon Technology and Innovation yang berlokasi di Jalan Kampung Baru IV, Ulujami, Jakarta Selatan. Kegiatan ini berlangsung mulai tanggal 1 Juli 2024 hingga 30 September 2024. Pada dua bulan pertama, yaitu 1 Juli hingga 31 Agustus 2024, pengerjaan dilakukan di kantor (*Work From Office*) dengan jam kerja pukul 07.30 - 17.00 untuk hari Senin hingga Kamis dan pukul 07.30 - 16.30 untuk hari Jumat. Pada bulan terakhir, pengerjaan dilakukan dari rumah (*Work From Home*) karena telah memasuki periode perkuliahan.

1.6. Metodologi Kerja Praktik

Metodologi dalam pembuatan buku laporan kerja praktik ini meliputi:

1.6.1. Perumusan Masalah

Untuk memahami kebutuhan proyek ini secara mendalam, kami mengikuti sesi *sprint* yang dipimpin oleh mentor, Kak Adji Gunhardi. Dalam sesi tersebut, pertama-tama kami diberikan penjelasan mengenai konsep platform visualisasi. Selanjutnya, kami mendapatkan penjelasan terkait kebutuhan untuk menganalisis perilaku belanja pelanggan serta proyek migrasi platform visualisasi berdasarkan permasalahan yang ada. Selanjutnya, penulis 1 mengadakan sesi *pairing* bersama tim *data analyst* untuk mendapatkan penjelasan mengenai konteks data *E-commerce* dan data apa saja yang dibutuhkan untuk analisis berdasarkan urgensi dan manfaatnya. Diskusi tersebut meliputi platform yang akan digunakan, sistem yang akan dibuat, data yang akan diambil, dan hal-hal teknis lainnya. Kemudian, penulis 2 mengadakan sesi *pairing*

bersama tim *data engineer* untuk mendapatkan teknologi apa saja yang digunakan untuk melakukan proses migrasi platform visualisasi.

1.6.2. Studi Literatur

Setelah memperoleh penjelasan mengenai aspek teknis dan non-teknis yang akan diimplementasikan, penulis melakukan tinjauan mendalam terhadap hal-hal tersebut. Penulis 1 mempelajari karakteristik data *E-Commerce* serta implementasi metode analisis dan visualisasinya. Proses pembelajaran ini dilakukan melalui diskusi dengan mentor dan tim data, serta pencarian informasi dari berbagai sumber di internet. Sementara itu, penulis 2 mempelajari teknologi yang akan digunakan dalam proses migrasi, seperti *containerization platform*, *deployment orchestrator*, *deployment orchestrator packaging tool*, *cloud-based relational database*, dan platform visualisasi itu sendiri. Selain itu, penulis 2 juga mendapatkan penjelasan tentang langkah - langkah untuk melakukan migrasi secara aman agar tidak mengganggu sistem yang sedang berjalan.

1.6.3. Analisis dan Perancangan Sistem

Setelah melakukan tinjauan terhadap aspek teknis dan non-teknis, langkah berikutnya adalah menyusun proses migrasi dengan sistem yang terstruktur. Proses migrasi dimulai dengan melakukan *dump* pada platform visualisasi yang berada di server *on-premise*. Data tersebut kemudian di-*restore* ke *cloud-based relational database*. Selanjutnya, platform visualisasi di-*deploy* pada lingkungan *deployment orchestrator* dengan melakukan penyesuaian konfigurasi pada *packaging tool* untuk menyambungkan platform visualisasi ke *cloud-based relational database* serta menyesuaikan kebutuhan sistem.

Setelah tahap migrasi selesai, proses ekstraksi data *E-Commerce* dilakukan menggunakan platform manajemen data untuk menyimpan dan mengelola data. Data yang telah diekstraksi kemudian diproses dan ditransformasikan untuk analisis lebih lanjut menggunakan metode *RFM analysis* dan *cohort analysis*. Hasil analisis ini divisualisasikan melalui platform visualisasi, yang menyediakan visualisasi interaktif guna mempermudah pembacaan dan interpretasi data.

1.6.4. Implementasi Sistem

Implementasi merupakan tahap realisasi dari perancangan yang telah dibuat. Pada tahap ini, penulis 2 melaksanakan migrasi platform visualisasi dari server *on-premise* ke lingkungan *deployment orchestrator*. Selanjutnya, penulis 1 bertanggung jawab atas pembuatan dan manajemen data *E-Commerce* yang relevan untuk dianalisis. Selanjutnya, data tersebut diproses melalui tahap ETL (*Extract, Transform, Load*) ke dalam platform manajemen data. Setelah itu, data dianalisis menggunakan metode *RFM analysis* dan *cohort analysis*. Terakhir, hasil analisis divisualisasikan menggunakan platform visualisasi.

1.6.5. Pengujian dan Evaluasi

Setelah proses migrasi selesai, perlu dilakukan evaluasi untuk menguji apakah platform visualisasi yang kini berada di lingkungan *deployment orchestrator* berfungsi dengan baik, sebagaimana di server *on-premise*. Jika ditemukan ketidaksesuaian atau masalah, maka perbaikan harus dilakukan dan proses migrasi perlu diulang. Kemudian, dilakukan validasi juga untuk memastikan sistem bekerja sesuai dengan

apa yang diinginkan dan data yang dihasilkan tetap terjaga integritasnya selama proses ETL.

1.6.6. Kesimpulan dan Saran

Kesimpulan dan saran akan dibuat setelah semua proses selesai dilakukan. Kesimpulan dan saran akan dibuat berdasarkan sistem yang telah dibuat.

1.7. Sistematika Laporan

1.7.1. Bab I Pendahuluan

Bab ini berisi latar belakang, tujuan, manfaat, rumusan masalah, lokasi dan waktu kerja praktik, metodologi, dan sistematika laporan.

1.7.2. Bab II Profil Perusahaan

Bab ini berisi gambaran umum PT Paragon Technology and Innovation mulai dari profil hingga lokasi perusahaan.

1.7.3. Bab III Tinjauan Pustaka

Bab ini memuat teori dasar mengenai teknologi yang digunakan dalam pengerjaan proyek kerja praktik.

1.7.4. Bab IV Analisis dan Perancangan Infrastruktur Sistem

Bab ini membahas tahap perancangan dan analisis sistem yang akan diimplementasikan dalam pengerjaan proyek kerja praktik.

1.7.5. Bab V Implementasi Sistem

Bab ini berisi uraian langkah - langkah yang dilakukan untuk proses implementasi sistem.

1.7.6. Bab VI Pengujian dan Evaluasi

Bab ini berisi hasil uji coba dan evaluasi dari sistem yang telah dikembangkan selama pelaksanaan kerja praktik.

1.7.7. Bab VII Kesimpulan dan Saran

Bab ini berisi kesimpulan dan saran yang didapat dari proses pelaksanaan kerja praktik.

BAB II

PROFIL PERUSAHAAN

2.1. Profil PT Paragon Technology and Innovation

PT Paragon Technology and Innovation adalah salah satu perusahaan kosmetik terbesar di Indonesia yang didirikan pada tahun 1985 oleh Nurhayati Subakat. Berawal sebagai industri rumahan, perusahaan ini berkembang pesat hingga memiliki R&D (Research and Development) Center kosmetik terbesar di Asia Tenggara. Paragon menaungi berbagai merek unggulan seperti Wardah, Make Over, Emina, Kahf, Labore, dan Biodef. Saat ini, PT Paragon Technology and Innovation memiliki lebih dari 12.000 karyawan dari seluruh Indonesia, memiliki lebih dari 1600 SKU produk, area pabrik sebesar 20Ha, dan 41 *distribution center* yang tersebar di Indonesia dan Malaysia.



Gambar 2.1 Logo PT Paragon Technology and Innovation

PT Paragon Technology and Innovation memiliki visi untuk selalu menjadi sebuah perusahaan yang berkomitmen untuk memiliki tata kelola perusahaan yang terbaik dan melakukan perbaikan yang berkesinambungan, untuk menjadikan setiap hari lebih baik dari hari kemarin, melalui produk-produk berkualitas tinggi yang bermanfaat bagi Paragon, mitra, masyarakat, dan lingkungan. Sebagai perusahaan yang peduli terhadap pemberdayaan manusia, PT Paragon Technology and Innovation juga berfokus pada pengembangan sumber daya manusia melalui pelatihan dan pendidikan berkelanjutan. Upaya ini

bertujuan untuk menciptakan tenaga kerja yang kompeten, berilmu, dan siap bersaing di industri global.

2.2. Visi dan Misi

PT Paragon Technology and Innovation memiliki visi dan misi yang dijabarkan sebagai berikut:

2.2.1. Visi

Sebuah perusahaan yang berkomitmen untuk memiliki tata kelola perusahaan yang terbaik dan peningkatan berkelanjutan, demi menjadikan setiap hari lebih baik dari hari sebelumnya, melalui produk-produk berkualitas tinggi yang memberikan manfaat bagi Paragonians, mitra, masyarakat, dan lingkungan.

2.2.2. Misi

1. Terus belajar dan mengembangkan Paragonian yang kompeten dengan keunggulan kompetitif.
2. Mendengarkan kebutuhan konsumen dan menciptakan produk yang melebihi ekspektasi mereka.
3. Selalu berinovasi dalam setiap proses untuk meningkatkan kualitas produk kami.
4. Bekerja sama dengan mitra bisnis untuk saling menguntungkan.
5. Berupaya sebaik mungkin melindungi bumi secara berkelanjutan karena bumi menyediakan semua yang kita butuhkan untuk hidup.
6. Mendukung perkembangan generasi baru melalui pengembangan sumber daya manusia agar menjadi pribadi yang berpengetahuan, berpengalaman, dan sehat.
7. Memperluas jangkauan produk dan layanan melalui ekspansi bisnis.

Visi dan misi Paragon Corp didukung dengan 5 *core values* perusahaan, yaitu *Faith in God* (percaya pada keberadaan dan kekuatan Tuhan), *Care* (menjunjung

tinggi nilai kebersamaan dan kasih sayang), *Humility* (setiap orang memiliki kelebihan dan kekurangan), *Grit* (menjalani hidup dengan semangat besar dan tekad yang kuat, dan *Innovation* (selalu mengembangkan hal-hal baru dan lebih baik untuk memenuhi dan melampaui ekspektasi pelanggan).

Paragon memiliki 4 pilar CSR (*Corporate Social Responsibility* atau Tanggung Jawab Sosial Perusahaan), yakni kesehatan, edukasi, pemberdayaan perempuan dan juga lingkungan dalam mendukung keempat SDGs yakni: *SDG 3 Good health and wellbeing*, *SDG 4 Quality education*, *SDG 8 Decent work and economic growth*, dan *SDG 12 Responsible consumption and production*. PT Paragon Technology and Innovation dibangun dengan keyakinan dengan jiwa semangat anak muda yang dapat saling membangun dan terbuka satu dengan yang lainnya dalam setiap proses.

2.3. Lokasi

Jl. Kp. Baru IV No.4, RT.5/RW.2, Ulujami, Kec. Pesanggrahan, Kota Jakarta Selatan, Daerah Khusus Ibukota Jakarta 12250.

[Halaman ini sengaja dikosongkan]

BAB III

TINJAUAN PUSTAKA

3.1. Platform Visualisasi

Platform Visualisasi adalah sebuah platform yang digunakan untuk visualisasi data dan pembuatan *dashboard*. Platform ini membantu pengguna, terutama *data analyst*, untuk mengambil, mengeksplorasi, serta memvisualisasikan data dari berbagai sumber dengan cepat dan mudah (Leibzon & Leibzon, 2018). Platform ini dirancang untuk memungkinkan tim membuat laporan berbasis data dan berbagi *insight* secara efisien. Jenis visualisasi yang tersedia pada platform ini mencakup grafik, tabel, *heat map*, *choropleth map*, dan berbagai bentuk visualisasi lainnya. Selain itu, hasil visualisasi dapat dibagikan, sehingga memudahkan kolaborasi antar anggota tim.

Arsitektur dari *visualization platform* terdiri dari beberapa komponen penting yang saling terintegrasi. Komponen tersebut meliputi *frontend*, *backend*, *database*, *task queue*, dan *data source integration*. *Frontend* bertanggung jawab dalam menyediakan antarmuka pengguna yang interaktif dan intuitif, sedangkan *backend* berfungsi untuk memproses permintaan pengguna dan mengelola data. *Database* digunakan untuk menyimpan konfigurasi, hasil *query*, serta *metadata*, sementara *task queue* berperan dalam mengeksekusi tugas secara asinkron.

Keunggulan utama dari platform ini terletak pada fitur *data source integration*, di mana platform ini mampu mengakses dan menggunakan berbagai sumber data, baik yang bersifat *on-premise* maupun berbasis *cloud*. Fleksibilitas ini memungkinkan pengguna untuk menggabungkan berbagai sumber data dalam satu tempat, sehingga analisis data dapat dilakukan secara lebih komprehensif dan efisien.

3.2. **Server *On-premise***

Server on-premise adalah jenis infrastruktur server yang dikelola dan dioperasikan secara langsung di lokasi fisik perusahaan. Infrastruktur ini mencakup perangkat keras server, perangkat lunak pendukung, serta sumber daya tambahan seperti jaringan, sistem penyimpanan data, dan pusat data yang sepenuhnya dimiliki dan dikendalikan oleh perusahaan. Berbeda dengan layanan *cloud computing* yang di-*hosting* dan dikelola oleh penyedia layanan pihak ketiga, server *on-premise* memberikan kontrol penuh kepada perusahaan terhadap operasional, keamanan, dan pengelolaan data. Infrastruktur ini memungkinkan perusahaan untuk menyesuaikan konfigurasi server sesuai dengan kebutuhan spesifik mereka, memastikan tingkat keamanan yang lebih ketat, serta mematuhi kebijakan atau regulasi internal yang mewajibkan data untuk disimpan secara lokal (Rawoof et al., 2022).

3.3. **Containerization Platform**

Containerization Platform adalah teknologi yang memungkinkan pengemasan, distribusi, dan eksekusi aplikasi di dalam unit yang disebut *container*. Teknologi ini menyediakan lingkungan yang terisolasi bagi aplikasi, sehingga dapat dijalankan secara konsisten di berbagai sistem operasi atau infrastruktur, tanpa terpengaruh oleh perbedaan lingkungan tersebut (Narasimhulu et al., 2023). *Container* mengemas semua komponen yang diperlukan untuk menjalankan aplikasi, seperti kode sumber, *libraries*, *runtime*, dependensi, dan konfigurasi sistem, sehingga memastikan aplikasi dapat berjalan dengan stabil dan efisien.

Dalam *Containerization Platform*, terdapat beberapa istilah penting, seperti *image* dan *registry*. *Image*

adalah file *read-only* yang berisi semua komponen yang diperlukan untuk menjalankan aplikasi. *Image* berperan sebagai cetakan dasar (*blueprint*) untuk membuat dan menjalankan *container*. Sementara itu, *registry* adalah tempat penyimpanan *image* yang berfungsi untuk mendistribusikan, mengelola, dan berbagi *image* di antara pengguna atau sistem.

3.4. *Deployment Orchestrator*

Deployment Orchestrator adalah sebuah sistem orkestrasi yang digunakan untuk mengelola dan mengotomatisasi *deployment*, *scaling*, serta pengelolaan aplikasi yang dikemas di dalam *container* (Hightower et al., 2017). Sistem ini memungkinkan aplikasi berjalan secara efisien dan konsisten di berbagai lingkungan, baik itu lokal, server *on-premise*, maupun infrastruktur berbasis *cloud*. *Deployment orchestrator* bekerja dengan cara mengelompokkan *container* ke dalam unit yang disebut *pod*. *Pod* merupakan unit terkecil dalam sistem orkestrasi yang dapat berisi satu atau lebih *container* yang saling berhubungan dan berbagi sumber daya, seperti jaringan dan penyimpanan.

Sistem ini dilengkapi dengan berbagai fitur utama untuk mendukung pengelolaan aplikasi, seperti *deployment* otomatis dengan fitur *rollout*, *auto-scaling* untuk menyesuaikan sumber daya berdasarkan kebutuhan, *load-balancing* untuk distribusi *traffic* secara merata, *resource management* untuk mengoptimalkan penggunaan CPU, memori, dan penyimpanan, serta *self-healing* yang memastikan stabilitas sistem. Pada proyek ini, fitur *self-healing* menjadi salah satu fitur unggulan, di mana *deployment orchestrator* secara otomatis memonitor *container* yang berjalan dan memperbaiki jika terjadi kegagalan, seperti memulai ulang *container* yang berhenti atau menggantikan *pod* yang rusak (Shamim et al., 2022).

Hal ini memastikan aplikasi tetap berjalan dengan optimal tanpa memerlukan intervensi manual.

3.5. *Deployment Orchestrator Packaging Tool*

Deployment Orchestrator Packaging Tool adalah alat yang digunakan untuk mengelola, mengemas, dan mendistribusikan konfigurasi aplikasi yang akan di-*deploy* pada lingkungan *deployment orchestrator* (Minna et al., 2024). Alat ini menyediakan *template* yang terstruktur dan dapat disesuaikan dengan kebutuhan dalam proses *deployment*. Dengan menggunakan alat ini, pengelolaan konfigurasi aplikasi menjadi lebih efisien, konsisten, dan mudah dipelihara.

Deployment Orchestrator Packaging Tool menggunakan konsep *chart*, yaitu kumpulan file yang berisi *template* dan parameter untuk mendeskripsikan sumber daya yang dibutuhkan oleh aplikasi. *File* ini mencakup definisi *service*, *deployment*, konfigurasi *volume*, dan *resource quota* yang diperlukan untuk menjalankan aplikasi di lingkungan *deployment orchestrator*. Dengan adanya alat ini, proses *deployment* menjadi lebih sederhana, terstruktur, dan dapat diatur secara otomatis di dalam lingkungan *deployment orchestrator*.

3.6. *Cloud-based Relational Database*

Cloud-based Relational Database adalah layanan manajemen *database* relasional yang berjalan di lingkungan *cloud computing*. Layanan ini memungkinkan pengguna untuk menyimpan, mengelola, dan mengakses data tanpa perlu menangani infrastruktur fisik secara langsung (Curino et al., 2011). Seperti *database* pada umumnya, layanan ini mendukung berbagai fungsi dasar, seperti pembuatan tabel, pengelolaan data, dan eksekusi *query*.

Karena berjalan di lingkungan *cloud computing*, *cloud-based relational database* dapat diakses melalui jaringan internet, sehingga memudahkan pengguna untuk mengelola data dari mana saja. Dengan menggunakan layanan ini, perusahaan dapat fokus pada pengelolaan data dan pengembangan aplikasi tanpa perlu khawatir tentang pemeliharaan infrastruktur fisik, seperti *hardware*, *backup*, atau peningkatan kapasitas.

3.7. SQL

SQL (dulunya SEQUEL, atau *Structured English Query Language*) berkembang dari dasar teori *query* relasional menjadi bahasa utama dalam sistem manajemen basis data relasional karena standarisasi dan pengaruh vendor DBMS (*Database Management System*). Diperkenalkan pada akhir 1970-an, SQL tetap relevan hingga kini sebagai bahasa serbaguna untuk mengambil, menyimpan, memodifikasi, dan menghapus data, serta mengelola objek basis data seperti tabel, kolom, prosedur, dan pengguna. Perintah pada SQL disebut sebagai *query*, terdiri dari pernyataan seperti SELECT, FROM, dan WHERE yang berfungsi untuk mengelola data dalam basis data. Pernyataan ini mencakup nama objek basis data, predikat (seperti LIKE, BETWEEN), operator (seperti AND, OR), kuantifier (seperti ANY, ALL), dan fungsi (seperti COUNT, SUM). SQL biasanya dibagi menjadi dua sub bahasa utama: *Data Manipulation Language* (DML) untuk operasi seperti SELECT, INSERT, UPDATE, DELETE, dan *Data Definition Language* (DDL) untuk operasi seperti CREATE, ALTER, dan DROP (Taipalus, T., & Seppänen, 2020).

SQL telah diterapkan dalam berbagai konteks teknologi informasi, termasuk pengelolaan data besar (*big data*), *cloud computing*, dan *artificial intelligence*. Penelitian oleh (Kim et al., 2020) menerapkan SQL dalam

konversi bahasa alami menjadi *query* SQL melalui model seperti SQL-Net dan NL2SQL.

3.8. **Data Warehouse dan Data Mart**

Data warehouse adalah sistem manajemen data yang menyimpan data historis untuk mendukung pengambilan keputusan dengan mengintegrasikan data dari sistem operasional dan sumber eksternal melalui ekstraksi, transformasi, integrasi, dan pembersihan. Tujuan utamanya adalah memberikan informasi tentang kinerja organisasi secara keseluruhan dan mendukung keputusan yang lebih efisien dan akurat. Berbeda dengan *database* operasional yang digunakan untuk transaksi sehari-hari (OLTP), *data warehouse* memiliki struktur yang lebih cocok untuk analisis karena dapat menangani *query* kompleks dan menyimpan data historis, sementara *database* operasional tidak dapat digunakan untuk analisis karena data yang terlalu detail, tanpa riwayat, dan performa yang buruk. (Filiana et al., 2020)

Data Mart, bagian kecil dari *data warehouse*, merupakan sebuah struktur data yang dirancang untuk mendukung kebutuhan analitik khusus, seperti analisis penjualan atau pemasaran. Berbeda dengan *Data Warehouse* yang menyimpan data dari berbagai sumber di seluruh organisasi, *Data Mart* hanya mencakup data yang relevan untuk keperluan analisis tertentu dalam lingkup departemen tersebut. *Database* operasional memberikan pandangan terintegrasi dengan skema yang kompleks, sedangkan *data warehouse* menyederhanakan data menjadi *data mart* dengan *star schema* atau skema bintang, yang memiliki tabel fakta dan dimensi. Tabel fakta berisi pengukuran kuantitatif dan tabel dimensi menjelaskan aspek bisnis. Proses normalisasi dapat menghasilkan skema *Snowflake* yang lebih kompleks namun mengurangi duplikasi data. Sebelum mendesain *data mart*, perlu

mendesripsikan sistem informasi perusahaan secara konseptual, seperti data apa yang perlu dikumpulkan, berapa banyak dimensi yang diperlukan, dan tingkat granularitas data yang akan disimpan. Metode pembangunan data *mart* melibatkan analisis *top-down* yang fokus pada kebutuhan pengguna, analisis *bottom-up* yang mempertimbangkan data operasional, dan integrasi keduanya untuk menghasilkan solusi yang sesuai (Bonifati et al., 2001).

3.9. ETL

Extract, Transform, Load (ETL) merupakan proses yang digunakan untuk mengumpulkan, menyaring, mengolah, dan menggabungkan data relevan dari berbagai sumber ke dalam sistem penyimpanan seperti data *warehouse* atau data *mart*, sehingga menghasilkan data historis, terangkum, statis, dan terstruktur untuk analisis (Melladia et al., 2024). Dalam pengelolaan data *mart*, ETL sangat penting untuk memastikan bahwa data yang dianalisis akurat dan relevan.

Langkah pertama pada proses ETL merupakan *Extract*, yaitu proses ekstraksi data dari berbagai sumber serta mengubahnya ke dalam format yang digunakan untuk proses kedua, yaitu *Transform*. Pada tahap kedua dilakukan transformasi untuk menyesuaikan dengan skema data *warehouse*, seperti melakukan *join* dengan sumber data lain dengan proses agregasi, *sorting*, atau filter. Data yang sudah diproses tersebut selanjutnya dimasukkan ke dalam sistem penyimpanan melalui tahap *Load*.

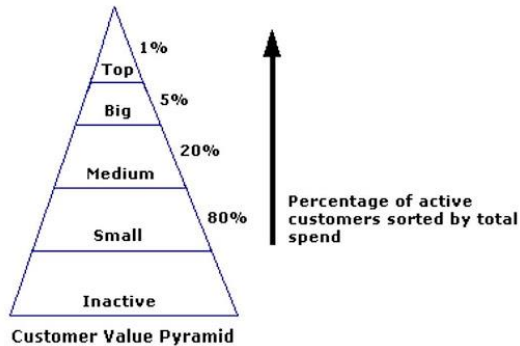
3.10. RFM Analysis

RFM analysis adalah metode analisis data yang digunakan untuk mengelompokkan pelanggan berdasarkan perilaku belanja mereka, yang terdiri dari tiga

komponen utama: *Recency* (waktu terakhir interaksi), *Frequency* (frekuensi transaksi), dan *Monetary* (jumlah uang yang dibelanjakan). Konsep RFM diperkenalkan oleh Bult dan Wansbeek (1995) dan terbukti sangat efektif ketika diterapkan pada basis data pemasaran (Birant, 2011). *Recency* mengukur seberapa baru interaksi pelanggan dengan produk atau layanan, yang dapat mencakup transaksi terakhir, kunjungan ke situs web, penggunaan aplikasi, atau interaksi di media sosial. Semakin baru interaksi tersebut, semakin besar kemungkinan pelanggan tersebut untuk melakukan pembelian lagi. *Frequency* mengukur seberapa sering pelanggan melakukan pembelian atau berinteraksi dengan produk dalam periode waktu tertentu. Pelanggan dengan frekuensi tinggi dianggap lebih loyal dan lebih terlibat dengan produk atau layanan, yang menunjukkan bahwa mereka cenderung lebih sering melakukan pembelian atau memberikan *feedback*. *Monetary*, di sisi lain, mengukur total uang yang dikeluarkan pelanggan dalam suatu periode. Pelanggan yang menghabiskan lebih banyak uang biasanya dianggap lebih berharga karena mereka menyumbang lebih banyak pendapatan untuk perusahaan.

Analisis RFM adalah metode tiga dimensi untuk mengklasifikasikan atau memberi peringkat pelanggan guna menentukan 20% teratas atau pelanggan terbaik. Metode ini didasarkan pada prinsip 80/20, yang menyatakan bahwa 20% pelanggan menghasilkan 80% pendapatan. Untuk mengelompokkan pelanggan dan melakukan analisis, digunakan model segmentasi pelanggan yang dikenal sebagai model piramida. Model piramida ini mengelompokkan pelanggan berdasarkan pendapatan yang mereka hasilkan ke dalam kategori yang ditunjukkan dalam Gambar 3.1. Kategori atau segmen nilai ini kemudian digunakan dalam berbagai analitik, yang fokus pada kategori dan istilah yang langsung relevan

dengan bisnis. Model piramida telah terbukti sangat berguna bagi perusahaan, organisasi keuangan, dan bank (Aggelis, n.d.).



Gambar 3.1 Model Piramida

3.11. *Cohort Analysis*

Cohort analysis adalah metode analisis data yang fokus pada pengelompokan individu ke dalam kohort, yaitu kelompok yang memiliki kesamaan atribut tertentu, seperti waktu pendaftaran atau tanggal pertama kali melakukan pembelian. Dengan memantau perilaku kelompok ini seiring waktu, *cohort analysis* membantu memahami pola perubahan atau tren yang terjadi dalam kohort tertentu, sehingga berguna untuk mengevaluasi efektivitas strategi pemasaran, pengembangan produk, atau analisis perilaku pelanggan. Pendekatan ini sering digunakan untuk mengidentifikasi faktor-faktor yang memengaruhi retensi, loyalitas, dan tingkat kepuasan pengguna dalam berbagai konteks bisnis.

Secara lebih luas, studi kohort merupakan jenis studi observasional di mana sekelompok individu dengan karakteristik tertentu diikuti untuk mengukur kejadian

suatu hasil, baik secara prospektif (hasil belum terjadi saat penelitian dimulai) maupun retrospektif (hasil sudah terjadi). Studi ini berguna untuk mempelajari insidensi, penyebab, dan prognosis suatu kondisi, serta mengukur risiko relatif atau hazard ratio. Meskipun tidak dapat memastikan hubungan sebab-akibat, studi kohort memberikan bukti kuat tentang hubungan antara faktor risiko dan hasil. Kelebihan dari pendekatan ini adalah kemampuannya untuk mempelajari berbagai hasil dari satu paparan, meskipun terdapat kelemahan seperti biaya tinggi dan potensi bias, terutama pada studi prospektif dengan periode pengamatan yang panjang. (Wang & Kattan, 2020)

3.12. *Interquartile Range*

Interquartile Range (IQR) adalah metode statistik yang digunakan untuk mengidentifikasi *outlier* dalam data dengan memanfaatkan distribusi data kuartil. IQR dihitung sebagai selisih antara kuartil ketiga dan kuartil pertama, yang mencerminkan variabilitas separuh data bagian tengah.

Dalam analisis statistik tradisional, membandingkan variabilitas antara dua kelompok biasanya dilakukan dengan uji kesetaraan variansi, seperti *Levene's Test* atau *F-test*. Namun, metode ini mengharuskan data memiliki distribusi normal. Ketika data tidak simetris, memiliki bentuk yang tidak teratur, atau mengandung pencilan dimana terdapat data yang berbeda jauh dari pola mayoritas, hasil uji bisa menjadi tidak valid. Sebagai alternatif, pengujian kesetaraan *Interquartile Range* (IQR) tidak memerlukan asumsi distribusi normal. Karena IQR hanya mempertimbangkan kuartil pertama dan kuartil ketiga (50% data tengah), metode ini lebih tahan terhadap pengaruh pencilan atau

nilai-nilai ekstrem. Pendekatan ini sangat berguna ketika distribusi data tidak simetris, berbentuk tidak teratur, atau terdapat kekurangan lain yang membuat metode para metrik kurang cocok (Greco et al., 2024).

[Halaman ini sengaja dikosongkan]

BAB IV

ANALISIS DAN PERANCANGAN INFRASTRUKTUR SISTEM

4.1. Analisis Sistem

Pada bab ini akan dijelaskan mengenai rancangan tahapan dalam proses migrasi platform visualisasi dari server *on-premise* ke dalam lingkungan *deployment orchestrator* yang nantinya akan dilakukan analisis perilaku pelanggan *E-Commerce* pada PT Paragon Technology and Innovation.

4.1.1. Deskripsi Sistem

Sistem yang dirancang dalam proses migrasi ini bertujuan untuk memindahkan platform visualisasi dari server *on-premise* ke lingkungan *deployment orchestrator*. Proses migrasi ini melibatkan perubahan infrastruktur untuk memastikan platform visualisasi dapat berjalan secara konsisten dan selalu tersedia. Dengan memanfaatkan *deployment orchestrator*, sistem akan secara otomatis memperbaiki dan menghidupkan ulang *container* yang bermasalah tanpa memerlukan intervensi manual, sehingga platform visualisasi tetap beroperasi dengan optimal.

Sistem ini menggunakan *deployment orchestrator packaging tool* untuk mempermudah manajemen konfigurasi dan pengemasan aplikasi dalam bentuk yang lebih terstruktur dan efisien. Data platform visualisasi yang sebelumnya berada di server *on-premise* akan melalui proses *dump* terlebih dahulu, kemudian di-*restore* ke *cloud-based relational database* yang berfungsi sebagai pusat penyimpanan data di lingkungan *deployment orchestrator*. Langkah ini dilakukan untuk memastikan pemisahan sumber daya antara *deployment orchestrator* dan platform

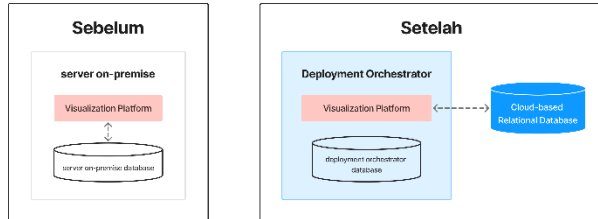
visualisasi, sehingga kinerja dan skalabilitas sistem dapat ditingkatkan.

Selain itu, sistem dapat mengelola dan menganalisis data pelanggan *E-Commerce* pada proyek melalui proses ekstraksi, transformasi, dan pemuatan (ETL) data dari berbagai sumber, pengelolaan data melalui data *mart*, serta penerapan analisis menggunakan metode RFM dan *cohort analysis*. Ditambah, sistem ini juga menyajikan visualisasi hasil analisis, yang menjelaskan data dalam bentuk grafik.

4.2. Perancangan Infrastruktur Sistem

4.2.1. Arsitektur Sistem

Pada sub bab ini, dijelaskan gambaran perubahan infrastruktur dari platform visualisasi yang sebelumnya berjalan di server *on-premise* menjadi lingkungan berbasis *deployment orchestrator*.



Gambar 4.1 Arsitektur Platform Visualisasi Sebelum dan Setelah Migrasi

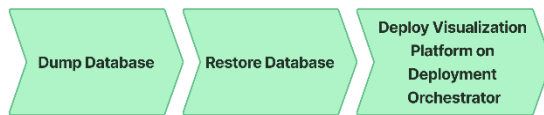
Pada Gambar 4.1, ditunjukkan bahwa sebelum proses migrasi dilakukan, server *on-premise* menjalankan platform visualisasi dalam arsitektur monolitik. Penyimpanan data masih menggunakan sumber daya atau *storage* yang dimiliki oleh server, sehingga dapat mengurangi performa server. Selain itu, pengaturan dan pemeliharaan sistem dilakukan secara

manual tanpa adanya dukungan sistem orkestrasi, yang menyebabkan kurangnya efisiensi dan fleksibilitas dalam pengelolaan.

Kemudian, ditunjukkan bahwa setelah proses migrasi dilakukan, platform visualisasi akan berada di lingkungan *deployment orchestrator*. Di dalam *deployment orchestrator*, proses *deployment*, *scaling*, dan *self-healing* dapat diatur secara otomatis. Aplikasi akan dikemas ke dalam sebuah *container* dan dikelola sepenuhnya oleh *deployment orchestrator*. Sementara itu, penyimpanan data akan dipindahkan ke *cloud-based relational database*, sehingga pengelolaan data menjadi lebih fleksibel dan terpisah dari server maupun platform.

4.2.2. Tahapan Migrasi

Tahapan proses migrasi direncanakan secara sistematis untuk memastikan platform visualisasi dapat dipindahkan dari server *on-premise* ke lingkungan *deployment orchestrator* secara efisien tanpa mengganggu sistem yang sedang berjalan. Proses migrasi ini terbagi seperti yang dapat dilihat pada Gambar 4.2 supaya platform dapat dipindahkan dengan lancar.



Gambar 4.2 Alur Tahapan Migrasi

Tahap pertama adalah melakukan *dump* atau pembuatan salinan *database* dari platform visualisasi yang berada di server *on-premise*. Proses *dump* ini memerlukan waktu yang cukup lama, terutama jika ukuran data sangat besar. Jika data terlalu besar, maka

perlu dilakukan proses kompresi untuk mempercepat pemindahan.

Setelah data berhasil disalin, tahap berikutnya adalah menyiapkan *cloud-based relational database*. Persiapan ini mencakup konfigurasi nama *database*, *endpoint*, *username*, *password*, dan *port* yang akan digunakan untuk mengakses *database*. Setelah semua konfigurasi selesai, langkah selanjutnya adalah *restore file database* yang telah disalin ke dalam *cloud-based relational database* yang sudah disiapkan.

Setelah proses *restore* selesai, tahap terakhir adalah deployment platform visualisasi di dalam *deployment orchestrator*. Sebelum itu, perlu dilakukan konfigurasi *values* pada template *deployment orchestrator packaging tool* agar sesuai dengan kebutuhan. Untuk menyambungkan *database* yang berada di *cloud* dengan platform, *values* pada *packaging tool* harus diubah dari yang awalnya menggunakan *local database* menjadi *external database*.

Dengan pemisahan *database* ke dalam *cloud-based relational database*, kinerja platform dapat ditingkatkan karena sumber daya untuk penyimpanan data tidak lagi membebani server aplikasi. Hal ini juga memungkinkan pengelolaan *database* menjadi lebih fleksibel dan mudah diakses sesuai kebutuhan.

4.2.3. Desain Sistem Ekstraksi

Tahap ekstraksi data pelanggan *E-commerce* dilakukan dengan menggunakan bahasa SQL dengan mengekstrak beberapa tabel pada data *mart* yang memberikan informasi tentang setiap order yang dilakukan oleh pelanggan *E-commerce*. Katalog data telah dibuat untuk mempermudah pada tahapan implementasi. Katalog berisi penjelasan dari semua

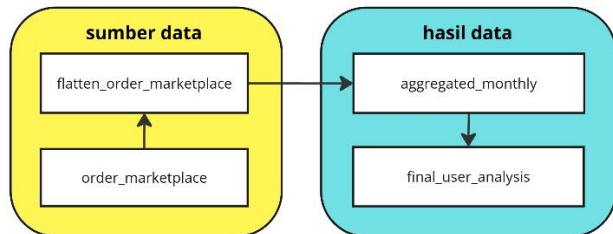
kolom data yang akan diekstrak dari beberapa tabel, ditunjukkan pada Tabel 4.1.

Tabel 4.1 Katalog data yang diseekstrak

Kolom	Tipe Data	Deskripsi	Sumber Tabel
marketplace	varchar	e-commerce platform	flatten_order_marketplace
shop_id	varchar	unique shop identifier	flatten_order_marketplace
shop_name	varchar	shop's name	flatten_order_marketplace
user_id	varchar	unique user identifier	flatten_order_marketplace
magpie_region_code	varchar	regional code	flatten_order_marketplace
kurs_rate	integer	currency exchange rate	flatten_order_marketplace
create_timestamp	timestamp	order creation time	flatten_order_marketplace
cancelled_flag	varchar	cancellation status	flatten_order_marketplace
returned_flag	varchar	return status	flatten_order_marketplace
order_id	varchar	unique order identifier	flatten_order_marketplace
product_gmv_amount	integer	product gross sales	flatten_order_marketplace
marketplace_product_id	varchar	product identifier	flatten_order_marketplace
marketplace_variant_id	varchar	variant identifier	flatten_order_marketplace
Marketplace_product_barcode	varchar	product's barcode	flatten_order_marketplace
order_shipping_fee_buyer_amount	varchar	buyer's shipping fee	flatten_order_marketplace
address_shipping_name	varchar	shipping address name	order_marketplace

4.2.4. Desain Sistem Transformasi

Tahap transformasi dilakukan setelah proses ekstraksi untuk mengolah data mentah dari sumber menjadi data yang siap digunakan dalam analisis perilaku pelanggan *E-Commerce*. Penulis memulai proses ini di skema ‘*playground*’ sebagai area uji coba tanpa mengubah data asli. *Query* yang dibuat kemudian akan diotomasi oleh tim data agar data dapat diperbarui setiap hari. Oleh karena itu, penulis merancang *query* yang mendukung proses pembaruan harian. Proses transformasi ini dilakukan dalam dua tahap utama. Pertama, data pelanggan *E-Commerce* digabungkan berdasarkan atribut pelanggan, atribut belanja, serta bulan dan tahun transaksi. Tabel ini merekap data belanja pelanggan yang dikelompokkan per bulan sebagai tahap awal analisis. Tabel 4.2 menjelaskan data-data yang berada pada tahap ini. Lalu, Gambar 4.3 menunjukkan visualisasi *data lineage*, menggambarkan alur data dari tahap ekstraksi hingga transformasi.



Gambar 4.3 Data Lineage

Tabel 4.2 Katalog data pada tabel ‘aggregated_monthly’

Kolom	Tipe Data	Deskripsi	Sumber Tabel	Sumber Atribut
customer_key	varchar	unique identifier for combined customer and order attributes	flatten_order_marketplace	marketplace, shop_id, shop_name, user_id, created_timestamp, magpie_region_code
marketplace	varchar	e-commerce platform	flatten_order_marketplace	marketplace
shop_id	varchar	unique shop identifier	flatten_order_marketplace	shop_id
shop_name	varchar	shop’s name	flatten_order_marketplace	shop_name
user_id	varchar	unique customer identifier	flatten_order_marketplace	user_id
month_of_order	varchar	order month	flatten_order_marketplace	created_timestamp, cancelled_flag, order_return_flag
magpie_region_code	varchar	regional code	flatten_order_marketplace	magpie_region_code
kurs_rate	integer	currency exchange rate	flatten_order_marketplace	kurs_rate
first_order_timestamp	timestamp	first order date	flatten_order_marketplace	created_timestamp, cancelled_flag, order_return_flag

Kolom	Tipe Data	Deskripsi	Sumber Tabel	Sumber Atribut
last_order_timestamp	timestamp	last order date	flatten_order_marketplace	created_timestamp, cancelled_flag, order_return_flag
total_order	integer	total orders count	flatten_order_marketplace	created_timestamp, cancelled_flag, order_return_flag
total_cancelled	integer	cancelled orders count	flatten_order_marketplace	cancelled_flag
total_returned	integer	returned orders count	flatten_order_marketplace	returned_flag
total_gmv_amount	integer	total gross merchandise value	flatten_order_marketplace	product_gmv_amount, cancelled_flag, order_return_flag
total_sku	integer	total product SKUs	flatten_order_marketplace	marketplace_product_id, marketplace_variant_id, cancelled_flag, order_return_flag
sku_list	object	product SKU list	flatten_order_marketplace	marketplace_product_id, marketplace_

Kolom	Tipe Data	Deskripsi	Sumber Tabel	Sumber Atribut
				variant_id, cancelled_flag, order_return_flag
total_quantity	integer	total item quantity	flatten_order_marketplace	market place_product_barcode, cancelled_flag, order_return_flag
address_shipping_name	varchar	shipping address name	order_marketplace	address_shipping_name, last_order_timestamp, cancelled_flag, order_return_flag
min_shipping_amount	integer	minimum shipping fee	order_marketplace	order_shipping_fee_buyer_amount, cancelled_flag, order_return_flag
max_shipping_amount	integer	maximum shipping fee	order_marketplace	order_shipping_fee_buyer_amount, cancelled_flag, order_return_flag
avg_shipping_amount	integer	average shipping fee	order_marketplace	order_shipping_fee_buyer_

Kolom	Tipe Data	Deskripsi	Sumber Tabel	Sumber Atribut
				amount, cancelled_flag, order_return_flag
load_timestamp	timestamp	Data load time	-	-
etl_batch_time	timestamp	ETL batch time	-	-

Selanjutnya, data dari tabel pertama selanjutnya diagregasi untuk menggabungkan perilaku belanja berdasarkan atribut pelanggan dan atribut belanja saja, membentuk tabel final yang diperlukan untuk analisis. Pada hasil akhir tabel, satu-satunya perbedaan dengan tabel sebelumnya adalah hilangnya kolom 'month_of_order' pada tabel final, yang artinya data final tersebut sudah menunjukkan perilaku belanja secara keseluruhan berdasarkan atribut pelanggan dan atribut belanja.

4.2.5. Desain Sistem Analisis

Sebelum masuk ke tahap analisis menggunakan metode RFM dan *cohort*, penulis melakukan analisis terhadap karakteristik data yang dihasilkan dari proses ETL. Untuk melakukan analisis RFM, perlu menentukan *threshold* untuk menentukan nilai *recency*, *frequency*, dan *monetary* untuk setiap pelanggan *E-commerce*. Penentuan *threshold* tersebut menggunakan metode IQR karena distribusi data yang dihasilkan tidak normal atau tidak simetris. Analisis RFM yang

dilakukan pada platform visualisasi memerlukan rekonstruksi terhadap kolomnya agar dapat digunakan untuk visualisasi. Kolom data yang diperlukan untuk visualisasi ini dijelaskan dalam Tabel 4.3.

Tabel 4.3 Katalog data untuk visualisasi RFM analysis

Kolom	Tipe Data	Deskripsi	Sumber Tabel	Sumber Atribut
user_id	varchar	unique user identifier	final_user_analysis	user_id
recency	integer	days since last order	final_user_analysis	last_order_timestamp
r_score	integer	recency score (1-3)	-	recency
frequency	integer	total number of orders	final_user_analysis	total_order
f_score	integer	frequency score (1-3)	-	frequency
monetary	integer	total GMV value	final_user_analysis	total_gmv_amount
m_score	integer	monetary score (1-3)	-	monetary
total_score	integer	sum of RFM scores	-	r_score, f_score, m_score

Selanjutnya, untuk visualisasi terhadap cohort analysis juga memiliki kriteria yang sudah ditentukan sehingga butuh diolah kembali menjadi data-data yang ada pada Tabel 4.4.

Tabel 4.4 Katalog data untuk visualisasi cohort analysis

Kolom	Tipe Data	Deskripsi	Sumber Tabel	Sumber Atribut
cohort_month	timestamp	month of user first order	final_user_analysis	month_of_order
cohort_size	integer	number of unique users	final_user_analysis	user_id
Months	integer	months since user registration	final_user_analysis	month_of_order
value	integer	number of active users	final_user_analysis	user_id

[Halaman ini sengaja dikosongkan]

BAB V

IMPLEMENTASI SISTEM

Bab ini membahas tentang implementasi dari sistem yang telah dirancang.

5.1. *Dump Database*

Proses *dump database* dilakukan sebagai langkah awal dalam migrasi platform visualisasi dari server *on-premise* ke *deployment orchestrator*. Tahap ini bertujuan untuk membuat salinan lengkap dari *database* yang digunakan oleh platform visualisasi agar dapat di-*restore* pada *database cloud* yang baru.

5.1.1. *Identifikasi Database*

Pertama, akses server *on-premise* menggunakan SSH. Setelah berhasil masuk, identifikasi *container* yang menjalankan platform visualisasi dan berfungsi sebagai *database*. Setelah *container* tersebut ditemukan, akses *container* untuk mengumpulkan informasi yang diperlukan. Dengan melakukan alur yang ada di Alur 5.1, didapatkan informasi penting mengenai *database* yang digunakan oleh platform visualisasi. Informasi yang perlu dikumpulkan meliputi jenis *database*, nama *database*, dan ukuran *database*. Berikut adalah langkahnya:

1. Akses server menggunakan SSH.
2. Identifikasi *container* yang berjalan sebagai *database*.
3. Masuk ke dalam *container*.
4. Masuk ke dalam *database*.
5. Identifikasi informasi *database* dengan menggunakan *query*.
6. Keluar dari *database*.

5.1.2. Melakukan *Dump Database*

Proses *dump* dilakukan menggunakan *command line tools* (CLI) sesuai dengan jenis *database* yang digunakan. Apabila ukuran data yang telah diidentifikasi terlalu besar dan ruang penyimpanan terbatas, proses kompresi perlu dilakukan dengan menambahkan beberapa argumen tambahan. Data yang di-*dump* dengan kompresi akan memiliki format `\.dump\``. Setelah proses *dump* selesai, data perlu dikeluarkan dari *container* dengan cara menyalinnya. Kemudian, data dapat dicek pada direktori tujuan dan ukuran file dapat diverifikasi untuk memastikan proses berjalan dengan benar. Berikut adalah langkahnya:

1. *Dump database*.
2. *Copy file dump* keluar dari *container*.
3. Cek apakah *file* sudah keluar dari *container*.

5.2. *Restore Database*

Proses *restore* akan menggunakan *cloud-based relational database* sebagai *database* tujuan. Setelah memiliki *endpoint*, nama, *username*, *password*, dan *port* dari *cloud-based relational database*, tahap *restore* dapat dimulai. Mengingat ukuran file *dump* yang sangat besar, proses *restore* akan memakan waktu yang cukup lama. Oleh karena itu, proses *restore* akan dijalankan di latar belakang. Untuk menjalankan proses *restore* di latar belakang, perlu dibuat file yang berisi *endpoint* dari *cloud-based relational database*, nama *database*, *username*, dan *password*. Setelah itu, file tersebut perlu diberikan izin akses agar dapat dibaca. Seluruh proses dapat dilakukan dengan mengikuti langkah berikut:

1. Tes dan cek koneksi *database*.
2. Simpan *file* yang berisi akses ke *database* untuk proses *restore* di latar belakang.
3. Edit akses file supaya dapat dibaca.
4. *Restore database*.
5. Monitor proses *restore* apakah sudah selesai atau belum.
6. Masuk ke dalam *database*.
7. Cek apakah data sudah sepenuhnya ter-*restore*.

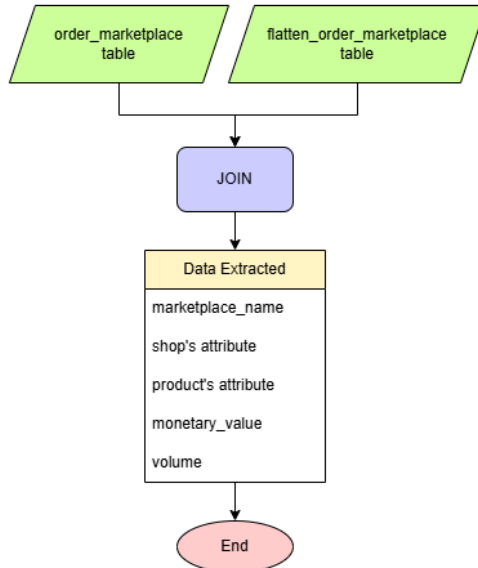
5.3. Deploy Platform Visualisasi

Tahap selanjutnya adalah proses *deployment* platform visualisasi ke dalam lingkungan *deployment orchestrator* menggunakan *deployment orchestrator packaging tool*. Platform visualisasi terdiri dari beberapa *container*, yaitu *server*, *worker*, *scheduler*, *migrations*, dan *database*. Setelah *template packaging tool* untuk *deployment platform* ini diperoleh, Perlu dilakukan edit pada bagian konfigurasi *database*, yaitu dengan menggunakan *endpoint* dari *cloud-based relational database* yang telah diisi sebelumnya. Selain itu, bagian ``env`` pada *server*, *worker*, *scheduler*, dan *migrations* juga perlu disesuaikan dengan *endpoint* tersebut.

Karena platform visualisasi memiliki sistem untuk mengamankan data, maka dalam *packaging tool*, perlu diedit *value* dari *cookie secret* dan *secret key* yang dapat diperoleh dari server *on-premise*. Hal ini diperlukan agar data dapat dibaca dengan aman oleh platform visualisasi yang ada di *deployment orchestrator*. Dengan ini, platform visualisasi dapat di-*deploy* pada *deployment orchestrator* dengan menggunakan *cloud-based relational database* sebagai tempat penyimpanannya dan dapat digunakan.

5.4. Implementasi Sistem Ekstraksi

Implementasi sistem ekstraksi meliputi proses pengambilan data dari data sumber, di mana pada proyek ini merupakan 2 tabel pada data *mart*. Gambar 5.1 menjelaskan diagram alir pada sistem ekstraksi.



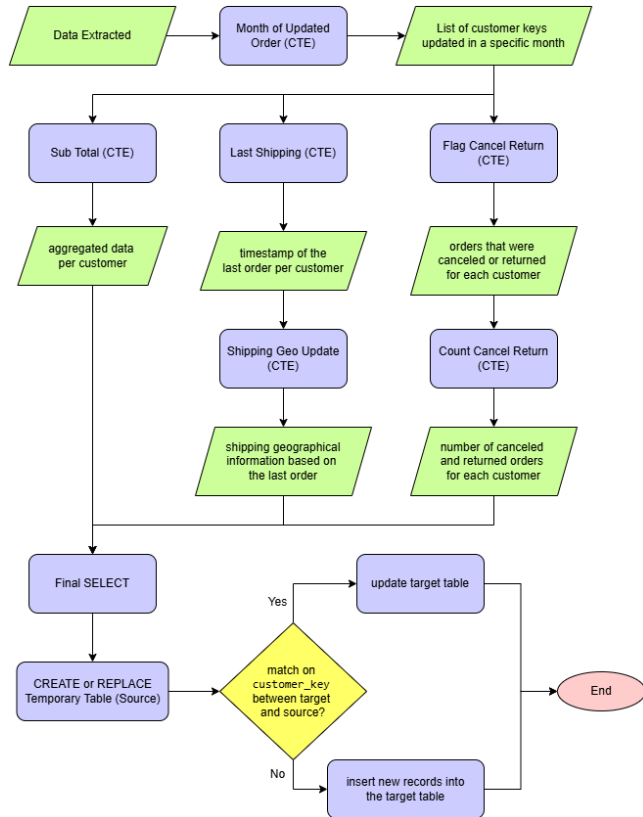
Gambar 5.1 Diagram alir sistem ekstraksi

5.5. Implementasi Sistem Transformasi

Sistem transformasi diimplementasikan setelah proses ekstraksi data, dilakukan dalam 2 tahap dengan membuat data pelanggan *E-Commerce* berdasarkan atribut pelanggan, atribut belanja, serta bulan dan tahun transaksi, dan tahap selanjutnya dilakukan dengan membuat tabel final yang digunakan untuk proses analisis.

5.5.1. Monthly Aggregated

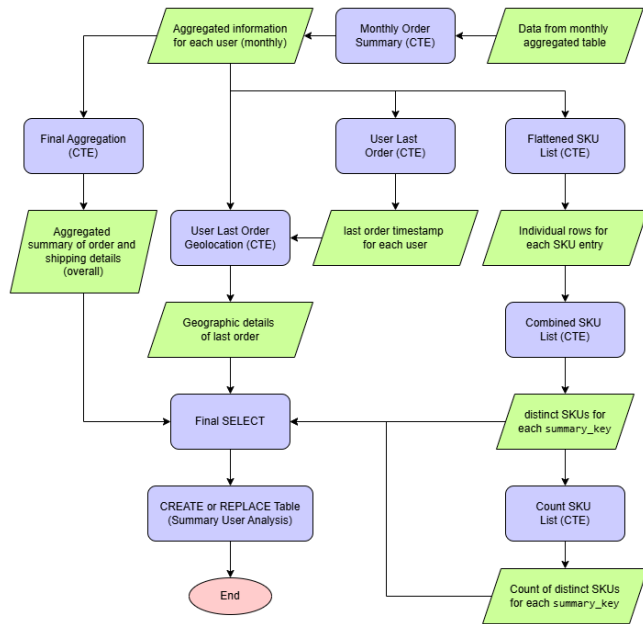
Pada tabel 'aggregated_monthly' menyimpan data pelanggan *E-Commerce* dengan mengelompokkannya berdasarkan bulan dan tahun mereka berbelanja. Proses ini dijelaskan dengan diagram alir pada Gambar 5.2.



Gambar 5.2 Diagram Alir Sistem Transformasi Awal

5.5.2. Final User Analysis

Selanjutnya, pada tahap ini dilakukan agregasi order pelanggan hanya berdasarkan atribut pelanggan dan atribut pesannya tanpa dipisahkan berdasarkan bulan dan tahun. Proses ini dijelaskan dengan diagram alir pada Gambar 5.3.



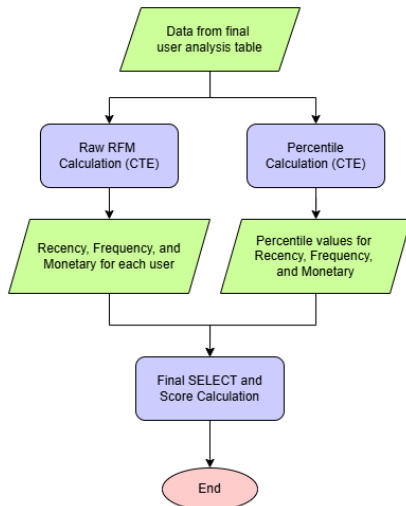
Gambar 5.3 Diagram Alir Sistem Transformasi Final

5.6. Implementasi Sistem Analisis

Sistem analisis mencakup visualisasi dari semua analisis yang dihasilkan yaitu *RFM analysis* dan *cohort analysis*.

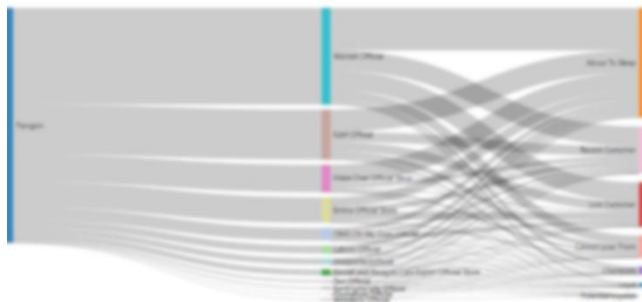
5.6.1. *RFM Analysis*

Sistem analisis pada tahap ini menggunakan pendekatan berbasis *query* SQL untuk menghitung metrik RFM dan mengkategorikan skor pelanggan ke dalam kuadran tertentu berdasarkan nilai kuartil dari masing-masing dimensi. Proses implementasi dimulai dengan langkah ekstraksi data pelanggan dari tabel final yang dihasilkan dari proses transformasi sebelumnya untuk membangun metrik RFM mentah. Data ini mencakup perhitungan *recency* sebagai selisih hari dari transaksi terakhir hingga saat ini, jumlah total pesanan, dan total nilai transaksi. Selanjutnya, distribusi data diolah untuk menentukan kuartil pertama (Q1) dan kuartil ketiga (Q3) pada masing-masing dimensi, yang digunakan sebagai batas kategori skor. Adapun implementasi dari sistem visualisasi untuk *RFM analysis* dijelaskan pada Gambar 5.4.



Gambar 5.4 Diagram Alir Sistem Visualisasi *RFM Analysis*

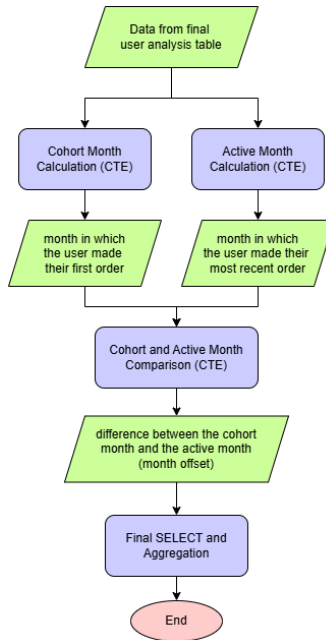
Pada Gambar 5.5, ditampilkan hasil visualisasi *RFM analysis* menggunakan diagram *sankey* secara *preview* dan dikaburkan karena data bersifat rahasia dan konfidensial.



Gambar 5.5 *Preview* Hasil Visualisasi *RFM Analysis*

5.6.2. *Cohort Analysis*

Implementasi cohort analysis dilakukan untuk mengelompokkan pengguna berdasarkan bulan pertama pembelian mereka (*cohort_month*) dan melacak aktivitas mereka dari waktu ke waktu untuk memahami pola retensi pengguna. Proses dimulai dengan pembentukan CTE untuk mengidentifikasi *cohort_month* berdasarkan pembelian pertama setiap pengguna dan *active_month* untuk mencatat bulan transaksi mereka. Data cohort dan aktivitas digabungkan menggunakan *user_id*, lalu dihitung selisih bulan (*month_offset*) antara *cohort_month* dan *active_month*. Selanjutnya, data tersebut diolah untuk menghitung jumlah pengguna unik (*active_users*) per kombinasi *cohort_month* dan *month_offset*, memberikan informasi mengenai retensi pengguna dari waktu ke waktu. Proses pada tahap ini dijelaskan pada Gambar 5.6.



Gambar 5.6 Diagram Alir Sistem Visualisasi *Cohort Analysis*

BAB VI

PENGUJIAN DAN EVALUASI

Bab ini membahas tentang tahap pengujian terhadap sistem yang dibuat penulis. Pengujian dan evaluasi dilakukan untuk memastikan bahwa sistem telah berhasil dibuat dan berjalan sesuai rencana awal. Pengujian dilakukan secara mandiri oleh penulis dan secara manual dengan mengamati hasil uji coba.

6.1. Tujuan Pengujian

Pengujian dilakukan terhadap platform visualisasi yang berada di lingkungan *deployment orchestrator* untuk memastikan platform berfungsi sesuai dengan perancangan. Pengujian ini juga mencakup kemampuan sistem dalam mengelola dan menganalisis data pelanggan *E-Commerce* melalui proses ekstraksi, transformasi, dan pemuatan (ETL) data dari berbagai sumber, pengelolaan data melalui data *mart*, serta penerapan analisis menggunakan metode *RFM analysis* dan *cohort analysis*. Selain itu, sistem ini juga diuji dalam menyajikan visualisasi hasil analisis dalam bentuk grafik yang informatif untuk mempermudah interpretasi data.

6.2. Kriteria Pengujian

Penilaian atas pencapaian tujuan pengujian didapatkan dengan memperhatikan beberapa hasil yang diharapkan berikut:

1. Memastikan platform visualisasi tetap tersedia dan dapat diakses tanpa gangguan setelah migrasi dilakukan.
2. Memverifikasi bahwa seluruh fungsi utama aplikasi bekerja sesuai dengan kebutuhan, termasuk kemampuan untuk memuat dan menampilkan data visualisasi.
3. Memastikan konektivitas antara platform visualisasi dan *cloud-based relational database*

- berjalan dengan baik, serta data dapat diakses dan diperbarui tanpa kendala.
4. Memastikan *self-healing* bekerja sebagaimana mestinya, seperti menghidupkan ulang *container* yang gagal.
 5. Memastikan kemampuan sistem dalam mengekstrak data dari beberapa sumber dan menyimpan data pelanggan *E-Commerce*.
 6. Memastikan kesesuaian hasil data yang dihasilkan setelah data ditransformasi.
 7. Memastikan kesesuaian data dan visualisasi dengan kebutuhan pengguna untuk memberikan *insight* yang *actionable*.

6.3. Skenario Pengujian

Skenario pengujian sistem dilakukan dengan menjalankan sistem dan melakukan pengujian langsung dan diamati hasilnya. Berikut langkah-langkah yang dilakukan untuk pengujian:

1. Pengguna membuka platform visualisasi yang berada di lingkungan *deployment orchestrator*.
2. Memeriksa kesesuaian *list* data visualisasi antara yang terdapat di server *on-premise* dan yang telah dimigrasikan.
3. Memeriksa salah satu detail dari data visualisasi.
4. Menambahkan data visualisasi baru dan memeriksa apakah data yang baru ditambahkan tersimpan.
5. Memberhentikan *container* yang sedang berjalan secara manual.
6. Menjalankan sistem ETL untuk menghasilkan data analisis pelanggan *E-commerce*.
7. Mengamati data yang dihasilkan dari proses ETL.
8. Menjalankan *query* untuk validasi data yang dihasilkan dengan membandingkan hasil data dari

sistem yang dibuat oleh penulis dengan hasil data yang diagregat langsung dari data sumber.

9. Menguji hasil visualisasi ketika data diperbarui.
10. Melakukan simulasi perubahan data dari sumber untuk pengujian sistem.

6.4. Evaluasi Pengujian

Hasil pengujian dilakukan terhadap pengamatan mengenai perilaku sistem yang telah dirancang terhadap kasus skenario uji coba. Tabel 6.1 di bawah ini menjelaskan hasil uji coba terhadap sistem yang telah dibuat.

Tabel 6.1 Hasil Evaluasi Pengujian

Kriteria Pengujian	Hasil Pengujian
Platform visualisasi tetap tersedia dan dapat diakses tanpa gangguan setelah migrasi dilakukan	Terpenuhi
Fungsi utama aplikasi bekerja sesuai dengan kebutuhan, termasuk kemampuan untuk memuat dan menampilkan data visualisasi	Terpenuhi
Konektivitas antara platform visualisasi dan <i>cloud-based relational database</i> berjalan dengan baik, serta data dapat diakses dan diperbarui tanpa kendala	Terpenuhi
<i>Self-healing</i> bekerja	Terpenuhi

sebagaimana mestinya, seperti menghidupkan ulang <i>container</i> yang gagal	
Kemampuan sistem mengekstrak data dari beberapa sumber dan menyimpan data pelanggan <i>E-commerce</i>	Terpenuhi
Data yang dihasilkan setelah proses transformasi tetap lengkap, akurat, dan konsisten	Terpenuhi
Kesesuaian data dengan kebutuhan pengguna untuk memberikan <i>insight</i> yang <i>actionable</i>	Terpenuhi

BAB VII KESIMPULAN DAN SARAN

7.1. Kesimpulan

Kesimpulan yang diperoleh setelah mengerjakan proyek pada kerja praktik ini, penulis dapat mengambil beberapa kesimpulan:

- a. Proses migrasi berhasil dilakukan dengan memastikan sistem dapat berfungsi sesuai kebutuhan tanpa mengganggu layanan yang berjalan.
- b. Dengan menggunakan *deployment orchestrator*, pengelolaan platform menjadi lebih fleksibel, mendukung fitur *scaling*, *self-healing*, dan pemantauan sistem secara lebih terstruktur.
- c. Proses ekstraksi data pelanggan E-commerce dari berbagai sumber berhasil diimplementasikan dengan menggunakan teknik ETL.
- d. Data yang telah diekstraksi berhasil disimpan dan diproses dalam sistem dengan transformasi yang tepat, menghasilkan data yang lengkap dan akurat.
- e. Proses ekstraksi tetap berjalan dengan baik walaupun dengan adanya pembaruan data.
- f. Dengan adanya sistem ini, perusahaan dapat menganalisis perilaku pelanggan dan tren bisnis untuk mendukung pertumbuhan dan pengembangan bisnis yang berkelanjutan.
- g. *RFM analysis* berguna untuk *treatment* yang lebih *personalized* berdasarkan segmen pelanggan, namun dalam kasus *e-commerce* yang menerapkan sistem PDP (Perlindungan Data Pelanggan), perlu mengidentifikasi pelanggan secara jelas agar RFM dapat dimanfaatkan untuk strategi pemasaran yang lebih tepat sasaran.

7.2. Saran

Setelah mengerjakan proyek pada kerja praktik ini, saran yang dapat diberikan penulis adalah sebagai berikut:

- a. Perbarui dan tinjau konfigurasi di *deployment orchestrator* secara berkala untuk memastikan skalabilitas dan keamanan platform tetap optimal.
- b. Lakukan dokumentasi lengkap mengenai setiap proses migrasi, konfigurasi, dan langkah pemeliharaan untuk mendukung keberlanjutan sistem.
- c. Menentukan jumlah segmen dalam *RFM analysis* tergantung pada apakah ada aksi yang spesifik yang dapat diambil untuk segmen tersebut; jika tidak ada, tidak perlu memisahkan segmen-segmen tersebut.
- d. Untuk memaksimalkan manfaat dari *RFM analysis*, perlu dilakukan upaya lebih dalam mengidentifikasi pelanggan secara jelas melalui pengumpulan data pelanggan yang sah dan sesuai dengan aturan PDP.
- e. Gunakan *cohort analysis* untuk mengukur dampak dari promosi atau kampanye pemasaran terhadap pelanggan yang ada dan bagaimana perilaku mereka berubah setelah promosi.

DAFTAR PUSTAKA

- Curino, C., Jones, E. P. C., Popa, R. A., Malviya, N., Wu, E., Madden, S., Balakrishnan, H., & Zeldovich, N. (2011). Relational Cloud: a Database Service for the cloud. *Conference on Innovative Data Systems Research*. <https://api.semanticscholar.org/CorpusID:1608241>
- Hightower, K., Burns, B., & Beda, J. (2017). *Kubernetes: Up and Running: Dive into the Future of Infrastructure*. <https://api.semanticscholar.org/CorpusID:29836329>
- Leibzon, A., & Leibzon, Y. (2018). *Redash v5 Quick Start Guide*. <https://api.semanticscholar.org/CorpusID:215924835>
- Minna, F., Massacci, F., & Tuma, K. (2024). Analyzing and Mitigating (with LLMs) the Security Misconfigurations of Helm Charts from Artifact Hub. *ArXiv*, *abs/2403.09537*. <https://doi.org/https://doi.org/10.48550/arXiv.2403.09537>
- Narasimhulu, M., Mounika, D. V., Varshini, P., K, A., & Rao, T. K. R. K. (2023). Investigating the Impact of Containerization on the Deployment Process in DevOps. *2023 2nd International Conference on Edge Computing and Applications (ICECAA)*, 679–685. <https://doi.org/https://doi.org/10.1109/ICECAA58104.2023.10212240>
- Rawoof, F. M., Tajammul, M., & Jamal, F. (2022). On-Premise Server Monitoring with Prometheus and Telegram Bot. *INTERANTIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT*. <https://doi.org/https://doi.org/10.55041/ijssrem12276>
- Shamim, S. I., Gibson, J. A., Morrison, P. J., & Rahman, A. A. U. (2022). Benefits, Challenges, and Research Topics: A Multi-

- vocal Literature Review of Kubernetes. *ArXiv*, *abs/2211.07032*.
<https://doi.org/https://doi.org/10.48550/arXiv.2211.07032>
- Aggelis, V. (n.d.). *Customer Clustering using RFM analysis*. 1–5.
- Birant, D. (2011). Data Mining Using RFM Analysis. *Knowledge-Oriented Applications in Data Mining*, iii.
<https://doi.org/10.5772/13683>
- Bonifati, A., Cattaneo, F., Ceri, S., Fuggetta, A., & Paraboschi, S. (2001). Designing data marts for data warehouses. *ACM Transactions on Software Engineering and Methodology*, 10(4), 452–483. <https://doi.org/10.1145/384189.384190>
- Filiana, A., Prabawati, A. G., Rini, M. N. A., Virginia, G., & Susanto, B. (2020). Perancangan Data Warehouse Perguruan Tinggi untuk Kinerja Penelitian dan Pengabdian kepada Masyarakat. *Jurnal Teknik Informatika Dan Sistem Informasi*, 6(2), 174–183.
<https://doi.org/10.28932/jutisi.v6i2.2557>
- Greco, L., Luta, G., & Wilcox, R. (2024). On testing the equality between interquartile ranges. *Computational Statistics*, 39(5), 2873–2898. <https://doi.org/10.1007/s00180-023-01415-8>
- Kim, H., So, B. H., Han, W. S., & Lee, H. (2020). Natural language to SQL: Where are we today? *Proceedings of the VLDB Endowment*, 13(10), 1737–1750.
<https://doi.org/10.14778/3401960.3401970>
- Melladia, M., Indah Febri Annisa, & Surya Kharisma Karnefo. (2024). Perancangan Data Mart Analisis Data Nilai Siswa pada SMAN 2 Tebo. *JUMINTAL: Jurnal Manajemen Informatika Dan Bisnis Digital*, 3(1), 18–30.
<https://doi.org/10.55123/jumintal.v3i1.3501>
- Taipalus, T., & Seppänen, V. (2020). SQL education: a systematic mapping study and future research agenda. *Environmental*

Modelling and Software, 144(2), 32.

<https://doi.org/10.1016/j.envsoft.2021.105134>

Wang, X., & Kattan, M. W. (2020). Cohort Studies: Design, Analysis, and Reporting. *Chest*, 158(1), S72–S78.

<https://doi.org/10.1016/j.chest.2020.03.014>

BIODATA PENULIS 1

Nama : Anneu Tsabita Putri
Tempat, Tanggal Lahir : Jakarta, 25 April 2003
Jenis Kelamin : Perempuan
Telepon : +6285715452778
Email : anneutsabita@gmail.com

AKADEMIS

Kuliah : Departemen Teknik Informatika –
FTEIC , ITS
Angkatan : 2021
Semester : 7 (Tujuh)

BIODATA PENULIS 2

Nama : Dewangga Dika Darmawan
Tempat, Tanggal Lahir : Magelang, 30 Juni 2003
Jenis Kelamin : Laki-laki
Telepon : +6281229549586
Email : dewangga3d@gmail.com

AKADEMIS

Kuliah : Departemen Teknik Informatika –
FTEIC , ITS
Angkatan : 2021
Semester : 7 (Tujuh)