



PENERAPAN METODE HAVERSINE UNTUK PENCARIAN DAN PEMESANAN RESTORAN DAN KAFE TERDEKAT PADA APLIKASI PICNICKER



UNIVERSITAS
Dinamika

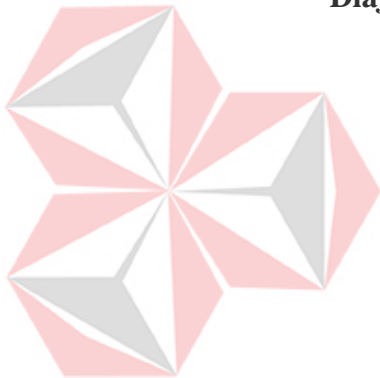
Oleh:
KEVIN OWEN DAVID KURNIAWAN
18410100015

FAKULTAS TEKNOLOGI DAN INFORMATIKA
UNIVERSITAS DINAMIKA
2022

**PENERAPAN METODE HAVERSINE UNTUK PENCARIAN DAN
PEMESANAN RESTORAN DAN KAFE TERDEKAT PADA APLIKASI
PICNICKER**

TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk menyelesaikan
Program Sarjana Komputer**



**UNIVERSITAS
Dinamika**

Oleh :

Nama : Kevin Owen David Kurniawan

NIM : 18.41010.0015

Program Studi : S1 Sistem Informasi

**FAKULTAS TEKNOLOGI DAN INFORMATIKA
UNIVERSITAS DINAMIKA**

2022

Tugas Akhir

PENERAPAN METODE *Haversine* UNTUK PENCARIAN DAN PEMESANAN RESTORAN DAN KAFE TERDEKAT PADA APLIKASI PICNICKER

Dipersiapkan dan disusun oleh
KEVIN OWEN DAVID KURNIAWAN
NIM : 18410100015

Telah diperiksa, diuji dan disetujui oleh Dewan Pembahas
Pada : Kamis, 18-08-2022

Susunan Dewan Pembahas

Pembimbing :

I. Julianto Lemantara, S.Kom., M.Eng.
NIDN. 0722108601


Digitally signed
by Julianto
Date: 2022.08.18
11:42:09 +07'00'

II. Ayouvi Poerna Wardhanie, S.M.B., M.M.
NIDN. 0721068904


Digitally signed
by Ayouvi Poerna
Wardhanie
Date: 2022.08.18
10:23:17 +07'00'

Pembahas :

I. **Ir. Henry Bambang Setyawan, M.M.**


Digitally signed
by Henry
Bambang S

Tugas Akhir ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Sarjana

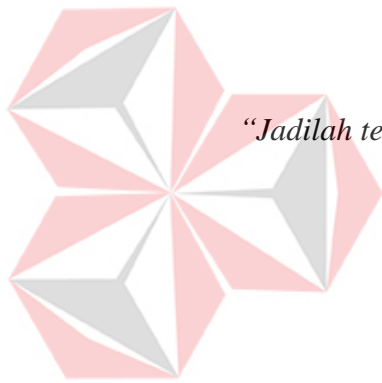


Digitally signed by
Universitas Dinamika
Date: 2022.08.18
13:56:01 +07'00'

Tri Sagirani, S.Kom., M.MT.

NIDN. 0731017601

Dekan Fakultas Teknologi dan Informatika
UNIVERSITAS DINAMIKA



“Jadilah terang dan garam dunia, walaupun untuk satu orang saja karena tiap orang memiliki dunianya sendiri-sendiri”.

Kevin Owen

UNIVERSITAS
Dinamika



*“Terima kasih Tuhan atas berkatMu dihidupku.
Kupersembahkan hasil karya dan perjuanganku kepada orang tua tercinta,
keluarga, partner tersayang, sahabat dan teman terbaik,
serta semua orang yang mengenaliku.*

*Pak Julianto, Ibu Ayowwi dan Pak Henry yang selalu membimbing
dengan arahan yang baik, bijaksana dan maksimal.*

Terima kasih, kuucapkan”

SURAT PERNYATAAN

PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH

Sebagai mahasiswa Universitas Dinamika, saya :

Nama : Kevin Owen David Kurniawan

NIM : 18410100015

Program Studi : S1 Sistem Informasi

Fakultas : Fakultas Teknologi dan Informatika

Jenis Karya : Tugas Akhir

Judul Karya : **PENERAPAN METODE HAVERSINE UNTUK PENCARIAN DAN PEMESANAN RESTORAN DAN KAFE TERDEKAT PADA APLIKASI PICNICKER**

Menyatakan dengan sesungguhnya bahwa:

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, saya menyetujui memberikan kepada Universitas Dinamika Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalti Free Right*) atas seluruh isi/ sebagian karya ilmiah saya tersebut di atas untuk disimpan, dialihmediakan dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama saya sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta
2. Karya tersebut di atas adalah karya asli saya, bukan plagiat baik sebagian maupun keseluruhan. Kutipan, karya atau pendapat orang lain yang ada dalam karya ilmiah ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiat pada karya ilmiah ini, maka saya bersedia untuk menerima pencabutan terhadap gelar keserjanaan yang telah diberikan kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Surabaya, 11 Agustus 2022

Yang menyatakan



Kevin Owen David Kurniawan
NIM : 18410100015

ABSTRAK

Penurunan kasus Covid-19 di tahun 2022 menyebabkan aktivitas di luar rumah mengalami peningkatan. Hal tersebut menjadi peluang bagi *startup* Picnicker, dimana saat ini masih belum terdapat platform untuk melakukan pemesanan tempat pada restoran atau kafe yang menyediakan informasi mendetail, seperti ketentuan pemesanan (minimal biaya pemesanan, kapasitas maksimal), fasilitas dan layanan (tempat parkir, *smoking area*, *Wi-Fi*, stop kontak). Dengan peluang tersebut, Picnicker ingin menyediakan platform yang dapat melakukan pemesanan tempat untuk restoran atau kafe dengan informasi sedetail mungkin yang terletak di sekitar lokasi penggunanya. Solusi yang ditawarkan berupa aplikasi Picnicker dengan fitur pencarian dan pemesanan restoran atau kafe untuk menampilkan restoran atau kafe terdekat dari penggunanya. Untuk mencari jarak terdekat dari restoran dan kafe yang ada dengan lokasi pengguna digunakanlah metode *haversine*, sehingga lokasi yang didapatkan akan lebih akurat. Dengan menggunakan metode *haversine* didapatkan tingkat akurasi sebesar 83% dan selisih jarak dengan menggunakan Google Maps memiliki rata-rata jarak sejauh 0.7km dengan selisih terendah 0.1km. Fitur-fitur yang terdapat pada aplikasi Picnicker terbukti mampu mengatasi kebutuhan pengguna, dimana 84% responden merasa terbantu dan puas dengan fitur yang terdapat pada aplikasi Pinicker yang membuat pemesanan restoran atau kafe menjadi lebih mudah dan cepat hingga 3.5x lipat dibandingkan dengan cara pemesanan konvensional.

Kata Kunci: Wisata Kuliner, Aplikasi, Pencarian, Pemesanan, Metode *Haversine*,

KATA PENGANTAR

Puji syukur kepada Tuhan Yang Maha Esa, atas berkat rahmat dan karunia-Nya penulis mampu menyelesaikan Tugas Akhir yang berjudul “Penerapan Metode *Haversine* untuk Pencarian dan Pemesanan Restoran dan Kafe Terdekat pada Aplikasi *Picnicker*” dengan lancar dan baik. Adapun maksud Tugas Akhir ini disusun untuk memenuhi persyaratan dalam menyelesaikan Program Studi Strata Satu (S1) Sistem Informasi di Universitas Dinamika. Rasa terima kasih pun penulis ucapkan atas bantuan berupa kritik, saran, dorongan motivasi, serta memberikan hiburan kepada penulis maka terselesaikanlah laporan ini. Dengan begitu, penulis ingin mengucapkan terima kasih kepada:

1. Kedua orang tua dan keluarga yang selalu memberikan semangat, dukungan dan doa yang terbaik untuk penulis dengan segala pengertiannya.
2. Bapak Prof. Dr. Budi Jatmiko, M.Pd. selaku Rektor Universitas Dinamika.
3. Bapak Dr. Anjik Sukmaaji, S.Kom., M.Eng. selaku Ketua Program Studi S1 Sistem Informasi.
4. Bapak Julianto Lemantara selaku Dosen Pembimbing I yang telah memberikan bimbingan yang selalu maksimal.
5. Ibu Ayouvi Poerna Wardhanie selaku Dosen Pembimbing II yang telah memberikan bimbingan yang selalu maksimal.
6. Bapak Ir. Henry Bambang Setyawan, M.M. selaku Dosen Penguji atas bimbingan, koreksi, saran dan arahan yang diberikan.
7. Partner tercinta, sahabat dan teman-teman terbaik yang telah membantu dalam pembuatan laporan Tugas Akhir ini, dan selalu memberikan dukungan moral dan solusi kepada penulis.
8. *Startup Picnicker* yang bersedia menerima dan menjadi objek penelitian Tugas Akhir ini.
9. Pihak-pihak lain yang tidak bisa dijelaskan satu-persatu yang telah membantu dan mendukung penulis.

Semoga Tuhan Yang Maha Esa memberikan yang terbaik kepada semua pihak atas bantuan yang telah diberikan kepada penulis baik itu dukungan moral maupun materi. Penulis menyadari bahwa laporan ini masih jauh dari kata sempurna sehingga kritik dan saran yang membangun dari semua pihak sangat memiliki arti bagi penulis dan semoga laporan ini dapat memberikan manfaat kepada pembaca.

Surabaya, 25 Juli 2022

Penulis

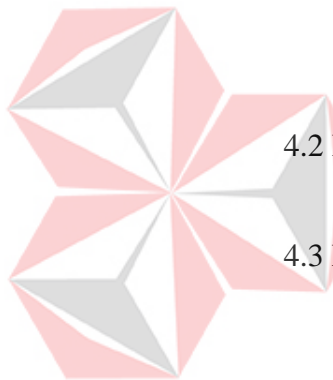


UNIVERSITAS
Dinamika

DAFTAR ISI

	Halaman
ABSTRAK	vii
KATA PENGANTAR.....	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR	xii
DAFTAR TABEL	xiv
DAFTAR LAMPIRAN	xv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Batasan Masalah.....	4
1.4 Tujuan.....	5
1.5 Manfaat.....	5
BAB II LANDASAN TEORI	6
2.1 Wisata Kuliner.....	6
2.2 <i>Haversine Method</i>	6
2.3 <i>Software Development Life Cycle (SDLC)</i>	8
2.4 <i>Metode Waterfall</i>	9
2.5 <i>Black Box Testing</i>	11
2.6 Penelitian Terdahulu	11
BAB III METODOLOGI PENELITIAN	13
3.1 Communication	13
3.1.1 Observasi.....	13
3.1.2 Survei dan Kuisisioner.....	13
3.1.3 Studi Literatur	14
3.1.4 Identifikasi Masalah	14
3.2 Perencanaan.....	14
3.3 Analisis.....	15
3.3.1 Analisis Pengguna.....	15
3.3.2 Analisis Sistem.....	21

BAB IV HASIL DAN IMPLEMENTASI.....	31
4.1 Implementasi Sistem	31
4.1.1 Penerapan Metode <i>Haversine</i>	31
4.1.2 Halaman Awal Pada Sisi Pelanggan	32
4.1.3 Halaman Pencarian <i>Merchant</i> Pada Sisi Pelanggan.....	33
4.1.4 Halaman <i>Merchant</i> Pada Sisi Pelanggan	33
4.1.5 Halaman Pemesanan Pada Sisi Pelanggan.....	34
4.1.6 Halaman Pendaftaran <i>Merchant</i> Pada Sisi <i>Merchant</i>	36
4.1.7 Halaman Transaksi Pemesanan Pada Sisi <i>Merchant</i>	37
4.1.8 Halaman Pencairan Dana Pada Sisi <i>Merchant</i>	37
4.1.9 Halaman Laporan Pemesanan Pada Sisi <i>Merchant</i>	37
4.1.10 Halaman Pengelolaan Data Master Pada Sisi Admin	38
4.1.11 Halaman Transaksi Pemesanan Pada Sisi Admin	38
4.1.12 Halaman Pencairan Dana Pada Sisi Admin	39
4.1.13 Halaman Laporan Pemesanan Pada Sisi Admin	39
4.2 Hasil Pengujian	39
4.2.1 Hasil Pengujian Metode <i>Haversine</i>	39
4.3 Evaluasi Sistem	41
4.3.1 Evaluasi Penerapan Metode <i>Haversine</i>	42
4.3.2 Evaluasi Pembangunan Sistem	43
BAB V PENUTUP	45
5.1 Kesimpulan.....	45
5.2 Saran.....	45
DAFTAR PUSTAKA.....	46
LAMPIRAN.....	48

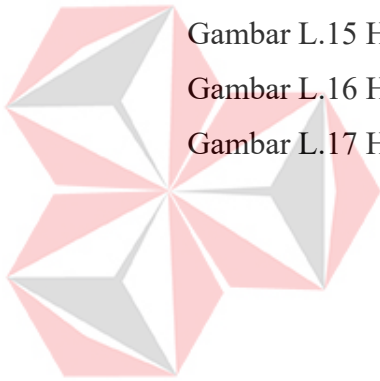


UNIVERSITAS
Dinamika

DAFTAR GAMBAR

	Halaman
Gambar 1.1 Grafik Covid-19 di Indonesia 13 Maret 2022	1
Gambar 2.1 Penggambaran metode <i>Haversine</i>	7
Gambar 2.2 Metode SDLC Waterfall.....	10
Gambar 3.1 Proses bisnis awal	15
Gambar 3.2 Proses bisnis usulan.....	16
Gambar 3.3 Diagram IPO	21
Gambar 3.4 Diagram IPO Lanjutan	22
Gambar 3.6 Diagram IPO Penerapan Metode <i>Haversine</i>	23
Gambar 3.7 <i>Sysflow</i> Pencarian Restoran / Kafe Dengan Metode <i>Haversine</i>	25
Gambar 3.8 <i>Sysflow</i> Pencarian & Pemesanan Restoran / Kafe.....	26
Gambar 3.9 Context Diagram	27
Gambar 3.11 Data Flow Diagram	29
Gambar 4.1 Penerapan Metode <i>Haversine</i>	31
Gambar 4.2 Fungsi Metode <i>Haversine</i>	31
Gambar 4.3 Halaman Utama Sisi Pelanggan	32
Gambar 4.4 Halaman Pencarian Sisi Pelanggan.....	33
Gambar 4.5 Halaman Merchant Sisi Pelanggan	34
Gambar 4.6 Halaman Pemesanan Sisi Pelanggan.....	35
Gambar 4.7 Halaman Detail Pemesanan Sisi Pelanggan.....	35
Gambar 4.8 Halaman Transaksi Pemesanan Sisi <i>Merchant</i>	36
Gambar 4.9 Halaman Transaksi Pemesanan Sisi <i>Merchant</i>	37
Gambar 4.10 Halaman Pencairan Dana Sisi <i>Merchant</i>	37
Gambar 4.11 Halaman Laporan Pemesanan Sisi <i>Merchant</i>	38
Gambar 4.12 Halaman Pengelolaan Data Master Sisi Admin	38
Gambar 4.13 Halaman Transaksi Pemesanan Sisi Admin	38
Gambar 4.14 Halaman Pencairan Dana Sisi Admin	39
Gambar 4.15 Halaman Laporan Pemesanan Sisi Admin	39
Gambar 4.16 Hasil Pengujian Metode <i>haversine</i>	41
Gambar L.1 Hasil Survei Awal 1	48

Gambar L.2 Hasil Survei Awal 2	48
Gambar L.3 Hasil Survei Awal 3	49
Gambar L.4 CDM Diagram	56
Gambar L.5 PDM Diagram.....	57
Gambar L.6 DFD Level 1 Pengelolaan Data Master	58
Gambar L.7 DFD Level 1 Transaksi Pemesanan	59
Gambar L.8 DFD Level 1 Laporan Transaksi Pemesanan.....	60
Gambar L.9 Desain UI Aplikasi.....	61
Gambar L.10 Hasil Survei Akhir 1	62
Gambar L.11 Hasil Survei Akhir 2.....	62
Gambar L.12 Hasil Survei Akhir 3	63
Gambar L.13 Hasil Survei Akhir 4	63
Gambar L.14 Hasil Survei Akhir 5	64
Gambar L.15 Hasil Survei Akhir 6	64
Gambar L.16 Hasil Survei Akhir 7	65
Gambar L.17 Hasil Turnitin	68

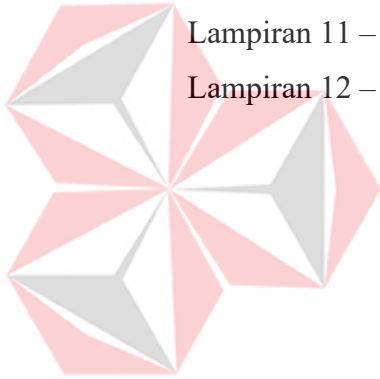


DAFTAR TABEL

	Halaman
Tabel 1.1 Perbedaan Aplikasi Yang Ada Saat Ini	2
Tabel 2.1 Perhitungan Menggunakan Metode <i>Haversine</i>	7
Tabel 2.2 Penelitian Terdahulu	11
Tabel 3.1 Identifikasi Masalah	14
Tabel 3.2 Analisis Kebutuhan Pengguna - Pelanggan.....	17
Tabel 3.3 Analisis Kebutuhan Pengguna - <i>Merchant</i>	18
Tabel 3.4 Analisis Kebutuhan Pengguna - Admin	19
Tabel 3.5 Analisis Kebutuhan Sistem Aplikasi	20
Tabel 4.3 Pengujian Metode <i>Haversine</i>	40
Tabel 4.4 Evaluasi Sistem	43
Tabel L.1 Jadwal Kerja	50
Tabel L.2 <i>User Persona</i> Pelanggan.....	51
Tabel L.3 <i>User Persona Merchant</i>	51
Tabel L.4 <i>User Persona Admin</i>	51
Tabel L.5 Sampel Data Penelitian.....	53
Tabel L.6 <i>Black box Testing</i>	66

DAFTAR LAMPIRAN

	Halaman
Lampiran 1 - Hasil Survei Awal	48
Lampiran 2 - Jadwal Kerja Penelitian	50
Lampiran 3 - User Persona Pengguna Picnicker	51
Lampiran 4 – Sampel Data Restoran / Kafe.....	53
Lampiran 5 – <i>Conceptual Data Model</i>	56
Lampiran 6 – <i>Physical Data Model</i>	57
Lampiran 7 - <i>Data Flow Diagram</i> (DFD) Level 1	58
Lampiran 8 - Desain <i>User Interface</i> (UI).....	61
Lampiran 9 – Hasil Survei Akhir	62
Lampiran 10 – <i>Black Box Testing</i>	66
Lampiran 11 – Hasil Turnitin.....	68
Lampiran 12 – Biodata Penulis	69



UNIVERSITAS
Dinamika

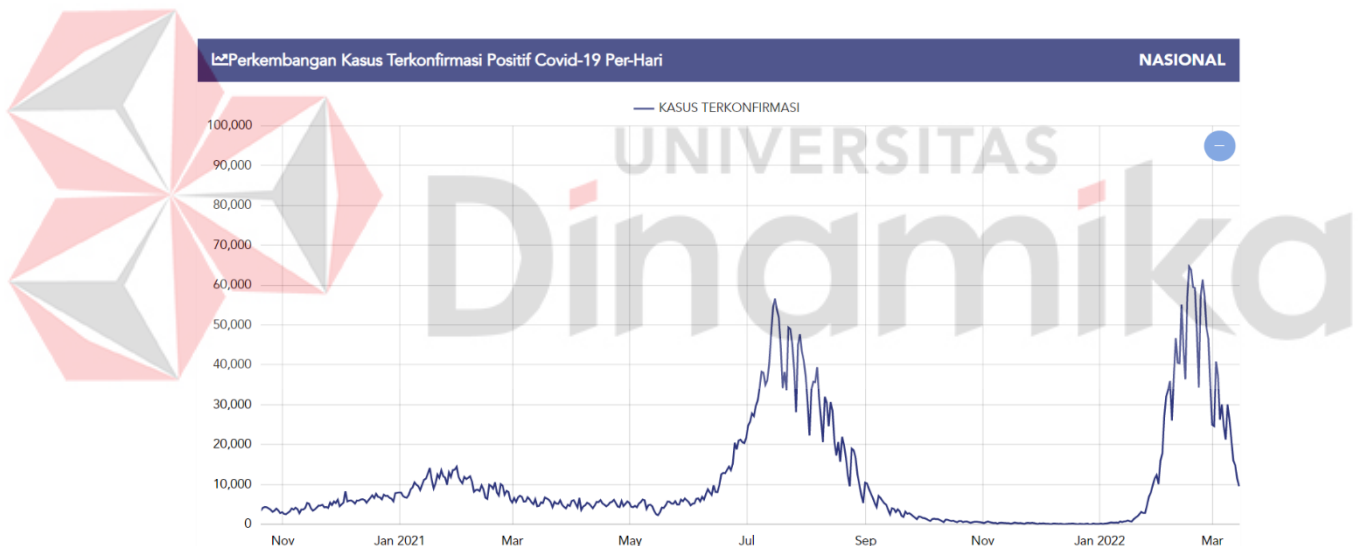
BAB I

PENDAHULUAN

1.1 Latar Belakang

Picnicker merupakan salah satu *startup* yang bergerak di bidang pariwisata, *startup* ini berdiri pada tahun 2020 di Surabaya. *Startup* Picnicker memiliki tujuan untuk menjadi *one stop travel* terdepan di Indonesia dan turut serta dalam mengembangkan tempat wisata yang aman dan terpercaya. Untuk mencapai tujuan tersebut Picnicker bertekad untuk memberikan informasi dan layanan pendukung dalam bidang pariwisata untuk mempermudah masyarakat berwisata dan turut berpartisipasi dalam memajukan objek wisata di Indonesia.

Pada tahun 2022 terlihat adanya penurunan kasus Covid-19 di Indonesia, hal ini ditunjukkan dengan nilai kurva kasus terkonfirmasi yang sedang bergerak



Gambar 1.1 Grafik Covid-19 di Indonesia 13 Maret 2022

menurun seperti yang terlihat pada Gambar 1.1 (covid19.go.id, 2022). Beberapa daerah di Indonesia terutama pada pulau Jawa dan Bali juga mengalami penurunan kasus, sehingga saat ini berada pada PPKM level 2 seperti pada kawasan Jabodetabek dan Surabaya Raya (Hakim, 2022). Dengan turunnya kasus Covid-19 dan level PPKM akan menyebabkan aktivitas di luar rumah seperti belajar, bekerja, *meeting* dan berkumpul bersama atau nongkrong mulai meningkat kembali. Hal ini dibuktikan dengan banyaknya jumlah perusahaan yang sudah menerapkan aturan untuk *work from office* (WFO) mulai dari 50% hingga 100% dari jumlah

karyawannya (Rafie, 2021). Beberapa destinasi wisata seperti di daerah Bandung dan Semarang sudah dibuka kembali dengan peraturan Covid-19 yang lebih longgar dan telah mengalami peningkatan jumlah pengunjung (Haryanto, 2022). Dengan meningkatnya aktivitas di luar rumah, *startup* Picnicker melihat peluang di mana saat ini masih belum ada platform yang menyediakan informasi mendetail mengenai restoran dan kafe yang ada di sekitar masyarakat yang dapat melakukan *booking* atau pemesanan tempat secara langsung. Sedangkan saat ini merupakan momentum yang tepat bagi pemilik bisnis terutama restoran dan kafe untuk kembali meningkatkan performa bisnisnya, dengan dikurangnya level PPKM, pemilik wisata kuliner dapat memanfaatkan 100% kapasitas bisnis yang mereka miliki. Menurut ketua umum APKRINDO Eddy Sutanto, saat ini sudah terbentuk *herd immunity* dan masyarakat sudah sadar serta dapat menangani atau mengantisipasi Covid 19 dengan baik (Pratama, 2022).

Tabel 1.1 Perbedaan Aplikasi Yang Ada Saat Ini

Perbedaan / Aplikasi	Google Maps	Pengiriman Makanan	Traveloka Xperience	Aplikasi Picnicker
Maps, Direksi & Jarak	Ya	Maps & Jarak	Maps & Direksi	Ya
Informasi Mendetail	Informasi Dasar	Informasi Dasar	Cukup Detail	Ya
Pemesanan Tempat	-	-	-	Ya
Pencarian	Ya	Ya	Ya	Ya & dilengkapi dengan berbagai fitur filter

Platform yang ada saat ini seperti Google Maps, Bing Maps, aplikasi pengiriman makanan dan Traveloka Xperience hanya menyediakan informasi dasar dari restoran atau kafe tersebut seperti nama, jam buka-tutup, ulasan dan penilaian, serta direksi menuju restoran atau kafe tersebut yang dapat dilihat pada Tabel 1.1. Informasi dan fitur yang ada saat ini tentunya masih kurang lengkap dan tidak cukup, terutama untuk orang-orang yang membutuhkan informasi mengenai restoran atau kafe dengan cepat dan lengkap. Seperti seorang pegawai yang

ditugaskan untuk mencari tempat *meeting* dengan *client*, akan kesusahan, kebingungan dan membutuhkan waktu yang lama untuk menentukan tempat yang diinginkan. Dengan informasi dan aplikasi yang ada saat ini, pegawai tersebut perlu menghubungi satu-persatu restoran atau kafe yang diminati untuk menanyakan informasi lebih lanjut (seperti kapasitas meja, minimal biaya pemesanan, dan maksimal durasi pemesanan) dan melakukan pemesanan tempat. Hal tersebut akan menghabiskan waktu yang cukup lama sehingga membuat pekerjaannya menjadi tidak efisien. Sedangkan apabila terdapat aplikasi yang dapat menyediakan, dan memenuhi kebutuhan tersebut, akan dapat menghemat waktu secara signifikan dan membuat pekerjaan tersebut lebih efisien.

Berdasarkan permasalahan tersebut maka akan dilakukan survei menggunakan metode statistik parametrik yang sekurang-kurangnya membutuhkan 30 data responden (Nursyafitri, 2021). Target responden survei ini dapat dilihat pada tabel *user persona* pengguna akhir pada sub-bab identifikasi pengguna. Berdasarkan survei yang telah diikuti oleh 43 responden didapatkan hasil bahwa 77% responden ingin kembali nongkrong ke restoran / kafe setelah menurunnya kasus Covid-19. Selain itu 47% responden juga sering kehabisan / kekurangan tempat pada restoran / kafe yang mereka kunjungi yang membuat mereka merasa kurang nyaman. Serta 36% responden juga kesulitan dalam mencari restoran / kafe terdekat yang sesuai dengan keinginan / selera mereka. Responden juga merasa bahwa informasi mengenai restoran dan kafe yang ada saat ini masih kurang, responden membutuhkan informasi seperti ketentuan pemesanan (minimal biaya pemesanan, maksimal durasi kunjungan), fasilitas (tempat parkir, *smoking area*, *meeting room*), serta layanan (mainan, *wi-fi*, stop kontak) yang disediakan. Selain itu 85% responden juga merasa terbantu dengan adanya informasi tambahan seperti jarak dari lokasi pengguna ke restoran / kafe yang dituju, kapasitas pengunjung per meja, dan fitur pemesanan tempat / meja pada aplikasi yang akan dibuat.

Dengan adanya permasalahan dan kebutuhan tersebut, *startup* Picnicker ingin menyediakan platform yang dapat menampilkan dan melakukan pemesanan tempat atau reservasi untuk restoran maupun kafe dengan informasi sedetail mungkin yang terletak di sekitar lokasi penggunaannya. Platform Picnicker juga akan mempermudah pengguna dalam mencari restoran atau kafe yang sesuai dengan kebutuhan

pengguna melalui fasilitas dan layanan seperti *wi-fi*, stopkontak, parkir gratis yang disediakan oleh pihak restoran atau kafe. Dengan adanya platform ini pengguna akan mendapatkan informasi mengenai restoran dan kafe yang lebih lengkap seperti, maksimal jumlah kapasitas per meja, minimal biaya untuk reservasi tempat, layanan-layanan yang disediakan oleh restoran atau kafe tersebut seperti stopkontak, *boardgames*, *card games*, dan lainnya, sehingga pengguna akan mendapatkan informasi lebih dan mempermudah pengguna dalam pembuatan keputusan. Untuk mendapatkan lokasi secara akurat maka akan menggunakan koordinat geografis yaitu *latitude* dan *longitude* untuk menentukan titik pengguna dengan titik restoran maupun kafe. Dengan adanya informasi geografis antara pengguna dengan restoran atau kafe maka dapat digunakan metode *haversine* untuk menentukan jarak dari kedua titik tersebut. Metode *haversine* adalah metode untuk menentukan sebuah jarak antara dua titik di permukaan bumi menggunakan perhitungan garis lintang dan garis bujur dari kedua titik tersebut (Y. Yulianto et al., 2018). Dengan menggunakan metode *haversine*, hasil yang didapatkan akan lebih akurat karena tidak menggunakan bidang datar melainkan bidang yang memiliki derajat kelengkungan. Metode *haversine* dipilih karena dapat menyajikan tingkat akurasi sebesar 98,66% (Miftahuddin et al., 2020) dari kedua titik yang dipilih, sehingga dapat mempermudah keputusan pengguna dalam memilih lokasi restoran maupun kafe terdekat yang sesuai dengan selera dan kebutuhan mereka.

1.2 Rumusan Masalah

Dari latar belakang di atas, maka dapat dirumuskan masalah yang ada saat ini adalah bagaimana merancang dan menerapkan metode *haversine* untuk pencarian dan pemesanan restoran dan kafe terdekat pada aplikasi Picnicker?

1.3 Batasan Masalah

Batasan masalah dari penelitian ini adalah sebagai berikut.

1. Penelitian ini menghasilkan aplikasi berbasis *website* yang akan disesuaikan dengan aplikasi Picnicker yang ada.
2. Hasil jarak pada aplikasi Picnicker di utamakan untuk perjalanan menggunakan kendaraan roda dua atau berjalan kaki.

3. Data yang digunakan merupakan data dari tempat wisata kuliner (restoran dan kafe) yang ada di provinsi Jawa Timur terutama pada daerah Surabaya Raya dan terakhir diperbarui pada tahun 2022.
4. Metode yang digunakan dalam proses pengembangan sistem adalah metode SDLC *Waterfall*, dan metode *haversine* untuk menentukan jarak antara objek wisata kuliner dengan pengguna.
5. Pengujian sistem akan dilakukan menggunakan metode *Black-box Testing*.

1.4 Tujuan

Berdasarkan rumusan masalah di atas, tujuan akhir yang akan dicapai adalah menerapkan metode *haversine* pada aplikasi Picnicker untuk mencari dan memesan restoran dan kafe terdekat, untuk mempermudah pengguna dalam melakukan pemesanan restoran atau kafe.

1.5 Manfaat

Manfaat yang dapat diperoleh dalam pengembangan aplikasi yaitu :

1. Membantu *startup* Picnicker dalam pengembangan sistemnya.
2. Membantu pengguna aplikasi Picnicker dalam mencari dan memesan tempat wisata kuliner (restoran dan kafe) yang sesuai di area sekitar pengguna.
3. Memperluas jangkauan bisnis dan jumlah kunjungan bagi *merchant* atau partner dari Picnicker.

BAB II

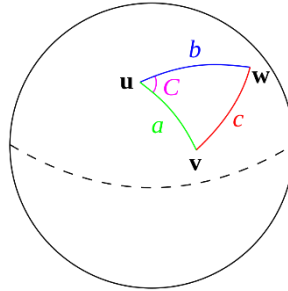
LANDASAN TEORI

2.1 Wisata Kuliner

Wisata adalah sebuah perjalanan menuju tempat lain yang dilakukan seseorang atau sekelompok orang, memiliki sifat sementara dengan tujuan untuk menikmati objek dan atraksi atau layanan pada tempat yang akan dikunjungi (Suwerna & Widyatmaja, 2017). Kuliner adalah suatu unsur budaya yang menunjukkan adanya hubungan sosial mengenai apa yang kita makan, bersama siapa kita makan dan bagaimana cara penyajian makanan tersebut menunjukkan peran dalam relasi sosial. Menurut (Utami, 2018) kuliner adalah hasil konstruksi budaya yang mengeksplorasi bagaimana makanan dan makan dipahami sebagai alat menyatukan beragam organisme. Berdasarkan kedua pengertian di atas wisata kuliner dapat diartikan sebagai sebuah perjalanan yang di dalamnya terdapat interaksi dengan objek kuliner seperti makanan, minuman, restoran maupun kafe dengan tujuan untuk menikmati makanan dan minuman maupun pengalaman yang didapatkan ketika mengonsumsi makanan atau minuman tersebut.

2.2 Haversine Method

Metode *Haversine* adalah sebuah metode untuk menghitung jarak antara dua titik lokasi di bumi menggunakan garis bujur (*longitude*) dan garis lintang (*latitude*). Menurut (Hartanto et al., 2017), metode *haversine* merupakan perhitungan navigasi yang penting karena dapat memperhitungkan jarak melingkar yang besar antara dua titik pada permukaan lingkaran. Visualisasi metode *haversine* dapat dilihat pada Gambar 2.1 di mana titik koordinat (u,v,w) yang terdapat pada lingkaran membentuk pembangun jarak untuk metode *haversine*.



Gambar 2.1 Penggambaran metode *Haversine*

Perhitungan metode *haversine* dapat dilihat pada rumus di bawah ini:

$$d = 2r \arcsin\left(\sqrt{\sin^2\left(\frac{\varphi_2 - \varphi_1}{2}\right) + \cos(\varphi_1) \cos(\varphi_2) \sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}\right) \dots (1)$$

Keterangan:

$\varphi_{1,2}$: *latitude* dalam satuan radians titik pertama dan kedua

$\lambda_{1,2}$: *longitude* dalam satuan radians titik pertama dan kedua

r : radius lingkaran

d : jarak

Berikut contoh perhitungan menggunakan metode *haversine*

Tabel 2.1 Perhitungan Menggunakan Metode *Haversine*

Lokasi	<i>Latitude</i>	<i>Longitude</i>	<i>Latitude</i> (Radians)	<i>Longitude</i> (Radians)
Universitas Dinamika (Lokasi saat ini)	-7.3132295	112.778276	- 0.127639934	1.968352241
Burger King Merr (destinasi 1)	-7.3210058	112.7778126	- 0.127775656	1.968344153
Soto Ayam Lamongan Cak Har (destinasi 2)	-7.3023939	112.7789205	- 0.127450817	1.96836349

Untuk mempermudah perhitungan maka akan menggunakan rumus pada aplikasi Microsoft Excel sebagai berikut:

$$d = r * 2 * \text{asin}(\text{sqrt}(\text{sin}((\text{lat}2 - \text{lat}1) / 2)^2 + \text{cos}(\text{lat}1) * \text{cos}(\text{lat}2) * \text{sin}((\text{lon}2 - \text{lon}1) / 2)^2)))$$

Dimana nilai r merupakan konstanta radius bumi sebesar **6371**km, dan nilai dari *latitude* dan *longitude* yang didapatkan harus diubah ke dalam bentuk satuan radians menggunakan rumus **radians(x)** kemudian *latitude* dan *longitude* 1 akan mewakili lokasi saat ini, sedangkan *latitude* dan *longitude* 2 akan mewakili destinasi tujuan sehingga akan menghasilkan jarak dalam satuan kilometer.

$$\begin{aligned} \text{Jarak destinasi 1} &= 6371 * 2 * \text{asin}(\text{sqrt}(\sin((-0.127775656 - 0.127639934)/2)^2 + \\ &\text{cos}(-0.127639934) * \text{cos}(-0.127775656) * \sin((1.968344153 - 1.968352241)/2)^2))) \\ &= \mathbf{0.866 \text{ km}} \end{aligned}$$

$$\begin{aligned} \text{Jarak destinasi 2} &= 6371 * 2 * \text{asin}(\text{sqrt}(\sin((-0.127450817 - 0.127639934)/2)^2 + \\ &\text{cos}(-0.127639934) * \text{cos}(-0.127450817) * \sin((1.96836349 - 1.968352241)/2)^2))) = \\ &= \mathbf{1.207 \text{ km}} \end{aligned}$$

Berdasarkan perhitungan di atas didapati jarak antara lokasi saat ini (Universitas Dinamika) dengan destinasi 1 (Burger King Merr) adalah sejauh 0.866km dan jarak ke destinasi 2 (Soto Ayam Cak Har) adalah sejauh 1.207km, Sehingga destinasi 1 lebih dekat dibandingkan destinasi 2.

2.3 Software Development Life Cycle (SDLC)

Software *Development Life Cycle* (SDLC) adalah tahapan-tahapan pekerjaan yang dilakukan oleh analis sistem dan *programmer* dalam membangun suatu sistem. Ada beberapa model SDLC. Model yang cukup populer dan banyak digunakan adalah *waterfall*. Beberapa model lain SDLC misalnya *fountain*, *spiral*, *rapid*, *prototyping*, *incremental*, *build & fix*, dan *synchronize & stabilize*. Dengan siklus SDLC, proses membangun sistem dibagi menjadi beberapa langkah dan pada sistem yang besar, masing-masing langkah dapat dikerjakan oleh tim yang berbeda. Dalam sebuah siklus SDLC, terdapat enam langkah yaitu:

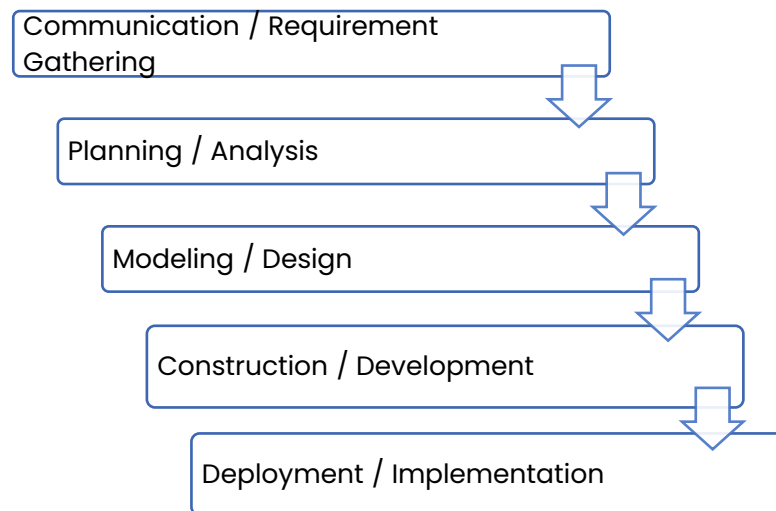
1. Analisis sistem, yaitu membuat analisis alur bisnis yang sudah ada dan sedang berjalan.
2. Spesifikasi kebutuhan sistem, yaitu melakukan perincian mengenai apa saja yang dibutuhkan dalam pengembangan sistem dan membuat perencanaan yang berkaitan dengan proyek sistem.

3. Perancangan sistem, yaitu membuat desain alur kerja bisnis dan desain pemrograman yang diperlukan untuk pengembangan sistem informasi.
4. Pengembangan sistem, yaitu tahap pengembangan sistem informasi dengan membuat program yang diperlukan.
5. Pengujian sistem, yaitu melakukan pengujian terhadap sistem yang telah dibuat.
6. Implementasi dan pemeliharaan sistem, yaitu menerapkan sistem yang telah dibuat dan memelihara sistem secara berkala.

Siklus SDLC dijalankan secara berurutan, mulai dari langkah pertama hingga langkah terakhir. Setiap langkah yang telah selesai harus dikaji ulang, terutama dalam langkah spesifikasi kebutuhan dan perancangan sistem untuk memastikan bahwa langkah telah dikerjakan dengan benar dan sesuai harapan. Jika tidak maka langkah tersebut perlu diulangi lagi atau kembali ke langkah sebelumnya. Dengan menerapkan siklus SDLC ini diharapkan aplikasi atau sistem yang sedang dibuat dapat berhasil memenuhi kebutuhan penggunanya. (Steven Dharmawan et al., 2018).

2.4 Metode Waterfall

Menurut Pressman model *waterfall* adalah model klasik yang bersifat sistematis, berurutan dalam membangun *software*. Metode *waterfall* juga bersifat fokus pada setiap fase sehingga proses pengerjaannya dapat maksimal dengan presentase kendala yang lebih kecil di bandingkan dengan pengerjaan secara paralel (Steven Dharmawan et al., 2018). Berikut ini fase-fase dalam model *waterfall* menurut Pressman yang dapat dilihat pada Gambar 2.2:



Gambar 2.2 Metode SDLC Waterfall

A. *Communication*

Langkah ini merupakan analisis terhadap kebutuhan *software*, dan tahap untuk melakukan pengumpulan data dengan cara melakukan pertemuan dengan *pelanggan* atau *client*, maupun mengumpulkan data-data tambahan baik yang ada di jurnal, artikel, maupun dari internet.

B. *Planning*

Proses *planning* merupakan tahap lanjutan dari proses *communication* (*analysis requirement*). Tahapan ini akan menghasilkan dokumen *user requirement* atau bisa dikatakan sebagai data yang berhubungan dengan keinginan atau kebutuhan *user* dalam *software* yang akan dibuat, seperti data apa saja yang dibutuhkan oleh sistem, termasuk rencana pengerjaan yang akan dilakukan.

C. *Modeling*

Proses *modeling* ini akan menerjemahkan syarat kebutuhan ke sebuah perancangan *software* yang dapat diperkirakan sebelum dibuat *coding*. Proses ini berfokus pada rancangan struktur data, arsitektur *software*, representasi *interface*, dan detail (algoritma) prosedural. Tahapan ini akan menghasilkan dokumen yang disebut *software requirement*.

D. *Construction*

Construction merupakan proses membuat kode. *Coding* atau pengembangan merupakan penerjemahan desain dalam bahasa yang bisa

dikenali oleh komputer. *Programmer* akan menerjemahkan transaksi yang diminta oleh *user*. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu *software*, artinya penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengembangan selesai maka akan dilakukan *testing* terhadap sistem yang telah dibuat tadi. Tujuan *testing* adalah menemukan kesalahan-kesalahan terhadap sistem tersebut untuk kemudian bisa diperbaiki.

E. *Deployment*

Tahapan ini bisa dikatakan final dalam pembuatan sebuah *software* atau sistem. Setelah melakukan analisis, desain dan pengembangan maka sistem yang sudah jadi akan digunakan oleh *user*. Kemudian *software* yang telah dibuat harus dilakukan pemeliharaan secara berkala.

2.5 *Black Box Testing*

Black Box Testing merupakan salah satu metode untuk melakukan pengujian terhadap aplikasi atau sistem yang sudah dibuat. Black Box Testing menekankan pengujian pada spesifikasi fungsional dari aplikasi yang dibuat, penguji dapat menyiapkan beberapa kondisi *Input* dan melakukan pengujian pada fungsi-fungsi yang terdapat di aplikasi. Menurut (Mursyidah & Hidayat, 2017) Black Box Testing sering kali digunakan untuk menemukan:

1. Fungsi atau fitur yang mengalami *error* atau *bug*.
2. Error yang terjadi pada antarmuka aplikasi.
3. Error pada struktur atau rancangan data.
4. Error pada performa aplikasi.

2.6 Penelitian Terdahulu

Setelah melihat dan mempelajari beberapa penelitian terdahulu, maka dapat dilihat perbedaan-perbedaan antara penelitian terdahulu dengan penelitian yang dilakukan oleh penulis dalam Tabel 2.2 berikut ini.

Tabel 2.2 Penelitian Terdahulu

Nama Peneliti	Judul	Hasil Penelitian
Penelitian Terdahulu 1		

Nama Peneliti	Judul	Hasil Penelitian
Yulianto, Ramadiani, Awang Harsa Kridalaksana	Penerapan <i>Haversine</i> Informasi Pencarian Jarak Terdekat Lokasi Lapangan Futsal	Formula Aplikasi untuk mencari lokasi lapangan futsal dengan metode <i>haversine</i> pada platform web.
Persamaan	Topik penelitian antara kedua penelitian sama yaitu keduanya menggunakan metode <i>haversine</i> untuk mencari jarak terdekat dari beberapa lokasi yang berbeda.	
Perbedaan	Penelitian yang dilakukan oleh (Y. Yulianto et al., 2018) menghasilkan aplikasi yang dapat digunakan oleh masyarakat secara umum tanpa <i>login</i> (akun) dan hanya menyediakan informasi mengenai lapangan tersebut saja, sedangkan pada penelitian ini akan menyajikan fitur untuk pemesanan dan <i>login</i> sehingga dari sisi bisnis akan mendapatkan data pengguna yang dapat diolah kembali.	
Penelitian Terdahulu 2		
Whelly Yulianto	Menentukan Jarak Terdekat Hotel dengan Metode <i>Haversine</i> Formula	Aplikasi untuk mencari hotel terdekat menggunakan metode <i>haversine</i> .
Persamaan	Topik penelitian antara kedua penelitian sama yaitu keduanya menggunakan metode <i>haversine</i> untuk mencari jarak terdekat dari beberapa lokasi yang berbeda.	
Perbedaan	Penelitian yang dilakukan oleh (W. Yulianto, 2015) hanya menyajikan informasi mengenai hotel terdekat saja, namun tidak menyajikan layanan pemesanan hotel tersebut, sedangkan pada penelitian ini akan menyajikan fitur atau layanan untuk pemesanan tempat atau reservasi pada restoran dan kafe.	

BAB III METODOLOGI PENELITIAN

Penerapan metode *Haversine* pada penelitian ini akan menggunakan *framework Software Development Life Cycle (SDLC) Waterfall*.

3.1 Communication

Pada tahap *communication*, akan dilakukan observasi, studi literatur dan analisis proses bisnis untuk mencari informasi lebih detail tentang proses bisnis yang dilakukan, kebutuhan data, kebutuhan pengguna, kebutuhan fungsional dan kebutuhan non-fungsional dari sistem yang akan dibuat.

3.1.1 Observasi

Langkah pertama yang dilakukan pada tahap *communication* adalah tahap observasi, observasi dilakukan untuk mengidentifikasi dan memastikan masalah dan kebutuhan pengguna aplikasi Picnicker. Observasi dilakukan untuk mengamati kondisi dan lingkungan sekitar yang terdampak Covid-19 terutama pada segmen pariwisata kuliner pada sosial media dan portal berita.

3.1.2 Survei dan Kuisisioner

Setelah tahap observasi, dilakukanlah survei untuk memperkuat dugaan terhadap hasil dari observasi yang dilakukan. Untuk memperkuat hasil observasi dilakukanlah survei kepada 43 partisipan dengan kriteria yang sesuai dengan *user persona* Picnicker yang terdapat pada Lampiran 3. Tujuan dari proses survei ini adalah untuk mencari tahu bagaimana pendapat, permasalahan dan kebutuhan target pengguna Picnicker mengenai objek wisata kuliner restoran dan kafe. Hasil dari survei yang dilakukan dapat dilihat pada Lampiran 1.

Selain itu juga dilakukan survei setelah aplikasi Picnicker selesai, survei ini dilakukan untuk memastikan apakah masalah yang diidentifikasi sebelumnya sudah terselesaikan dengan menggunakan aplikasi Picnicker. Survei ini diikuti oleh 38 partisipan, dan hasil dari survei ini dapat dilihat pada Lampiran 9.

3.1.3 Studi Literatur

Langkah selanjutnya adalah melakukan studi literatur untuk mengumpulkan informasi seputar topik penelitian dengan tujuan untuk mendalami teori serta penerapan dari sistem yang dibuat. Studi literatur dilakukan dengan cara mengumpulkan informasi dan data dari berbagai macam sumber pustaka seperti artikel, buku, jurnal ilmiah, laporan penelitian dan internet.

3.1.4 Identifikasi Masalah

Setelah melakukan studi literatur selanjutnya adalah melakukan identifikasi masalah terhadap hasil dari observasi yang telah dilakukan, setelah melakukan observasi didapatkan permasalahan sebagai berikut yang tertera pada Tabel 3.1.

Tabel 3.1 Identifikasi Masalah

No	Permasalahan	Dampak	Alternatif Solusi
1	Belum adanya platform untuk melakukan pemesanan tempat pada restoran atau kafe.	Masyarakat menjadi kesusahan atau kerepotan untuk melakukan pemesanan tempat pada restoran atau kafe.	Pembuatan platform Picnicker untuk melakukan pencarian dan pemesanan restoran dan kafe terdekat menggunakan metode <i>haversine</i> .
2	Platform penyedia informasi saat ini belum dapat menyajikan fitur pencarian restoran atau kafe secara mendetail hanya pencarian sederhana saja.	Masyarakat memerlukan waktu lebih untuk mencari restoran atau kafe yang sesuai dengan kebutuhan mereka.	

3.2 Perencanaan

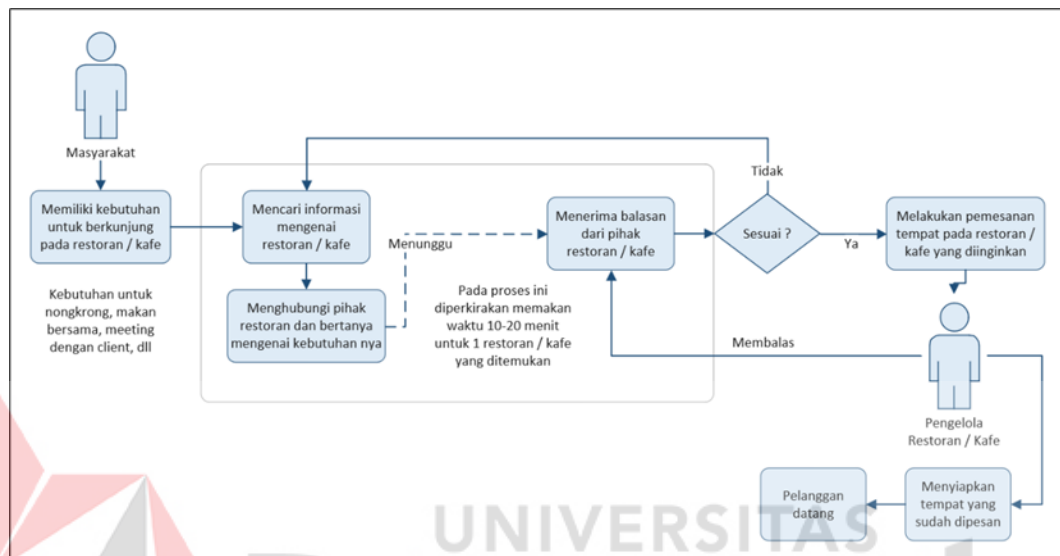
Tahapan berikut pada metode *waterfall* adalah tahap perencanaan atau *planning*. Pada Tahap perencanaan akan menjelaskan langkah-langkah teknis yang dilakukan dan estimasi waktu yang dibutuhkan dalam pengembangan sistem dalam bentuk jadwal kerja. Jadwal kerja pada pengembangan sistem Picnicker dapat dilihat pada Lampiran 2.

3.3 Analisis

3.3.1 Analisis Pengguna

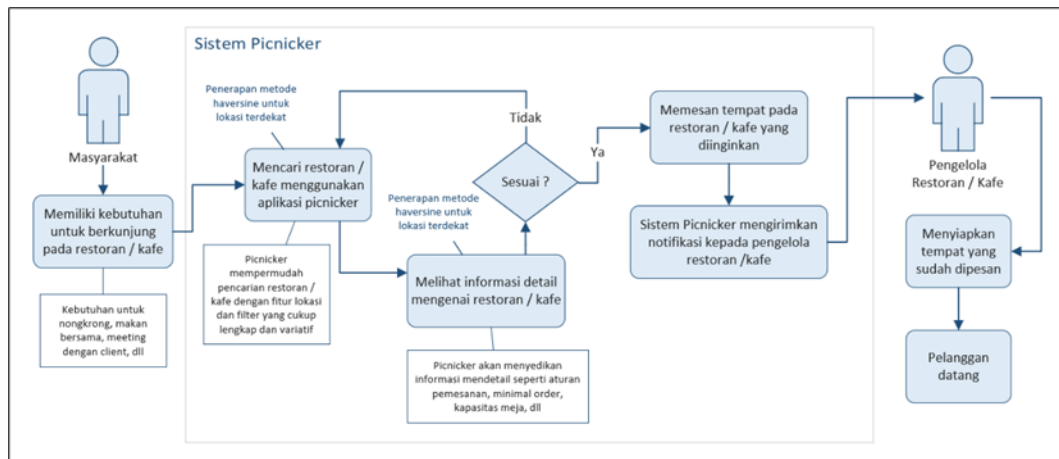
A. Analisis Proses Bisnis

Langkah terakhir pada tahap *communication* adalah melakukan analisis proses bisnis, pada tahap ini akan menjelaskan bagaimana proses bisnis saat ini dan proses bisnis usulan.



Gambar 3.1 Proses bisnis awal

Proses bisnis di atas merupakan gambaran proses bisnis saat ini di mana semua proses yang dilakukan masih konvensional dan belum ada sistem yang mampu menjembatani semua prosesnya. Seperti pada proses pencarian informasi yang harus dilakukan menggunakan beberapa alat seperti Google Maps, Instagram, *website* (blog / artikel), dan aplikasi pengiriman makanan. Kemudian ketika calon pelanggan ingin menghubungi pihak restoran atau kafe juga mengalami kendala seperti pihak restoran atau kafe yang tidak dapat dihubungi, respon yang lambat, dan lainnya. Sehingga berdasarkan proses bisnis tersebut maka akan diusulkan solusi alternatif berupa platform yang mampu menangi kebutuhan dan permasalahan yang ada. Proses bisnis usulan tersebut dapat dilihat pada Gambar 3.2 di bawah.



Gambar 3.2 Proses bisnis usulan

Dengan adanya proses bisnis usulan di atas proses yang dilakukan oleh masyarakat atau pengguna menjadi lebih ringkas, efisien dan efektif, karena semua proses yang dilakukan sudah diwadahi oleh satu platform yang sama. Sehingga pengguna tidak perlu bergonta-ganti platform untuk mencari informasi yang dibutuhkan untuk melakukan pemesanan tempat. Melalui aplikasi Picnicker pengguna dapat mencari restoran atau kafe terdekat dengan fitur lokasi Picnicker, mencari lebih spesifik dengan fitur filter dan langsung memesan dengan fitur pemesanan. Dengan adanya aplikasi Picnicker juga akan membantu pemilik atau pengelola restoran dan kafe dalam membagikan informasi mengenai usaha mereka dan juga dapat menangani pemesanan tempat atau reservasi.

B. Identifikasi Pengguna

Identifikasi pengguna merupakan tahapan untuk mendefinisikan pengguna yang akan menggunakan aplikasi yang akan dibuat. Berdasarkan dari tahap sebelumnya pengguna yang akan menggunakan aplikasi ini adalah pelanggan atau pengguna akhir, pemilik atau pengelola restoran atau kafe sebagai *merchant*, dan karyawan sebagai admin Picnicker. Dari ketiga pengguna tersebut dapat digambarkan karakteristiknya melalui *user persona* pada Lampiran 3.

C. Identifikasi Data

Identifikasi data dilakukan untuk mengetahui kebutuhan terkait data apa saja yang diperlukan dalam pengembangan sistem untuk memenuhi kebutuhan

pengguna yang sudah diidentifikasi sebelumnya pada tahapan identifikasi pengguna. Data yang diperlukan dalam pengembangan sistem antara lain :

1. Data Master, meliputi :
 - a. Data Wilayah (Provinsi, Kota)
 - b. Data Bank
 - c. Data Restoran & Kafe
 - d. Data Fasilitas Restoran & Kafe
 - e. Data Pemilik Restoran & Kafe
 - f. Data Rekening Pemilik Restoran & Kafe
 - g. Data Pengguna
 - h. Data Karyawan (Admin Picnicker)
 - i. Data Kategori & Jenis Restoran

2. Data Transaksi, meliputi :
 - a. Data Pemesanan / Reservasi Restoran & Kafe
 - b. Data Pembayaran Reservasi
 - c. Data Saldo Restoran & Kafe
 - d. Data Arus Kas Saldo Restoran & Kafe
 - e. Data *Rating* Restoran & Kafe
 - f. Data *Review* Restoran & Kafe

D. Analisis Kebutuhan Pengguna

Tahap lanjutan dalam pengumpulan kebutuhan adalah Identifikasi kebutuhan pengguna yang dilakukan untuk mengidentifikasi siapa saja pengguna yang nantinya akan menggunakan sistem yang dibuat. Data yang dibutuhkan meliputi:

1. Pengguna Akhir / Pelanggan

Tabel 3.2 Analisis Kebutuhan Pengguna - Pelanggan

Aktivitas	Kebutuhan Data	Kebutuhan Informasi
Pendaftaran akun pengguna	1. Data pengguna 2. Data wilayah	Daftar pengguna
Pencarian restoran / kafe	1. Data wilayah 2. Data restoran & kafe	Daftar dan informasi restoran / kafe

Aktivitas	Kebutuhan Data	Kebutuhan Informasi
	3. Data <i>rating & review</i> restoran & kafe	
Pemesanan tempat	1. Data pengguna 2. Data restoran & kafe 3. Data pemesanan tempat 4. Data (bukti) pembayaran reservasi 5. Data <i>rating & review</i> restoran & kafe	Daftar pemesanan tempat
Pemberian <i>Rating & Review</i>	1. Data pemesanan tempat 2. Data restoran & kafe 3. Data pengguna 4. Data <i>rating</i> 5. Data <i>review</i>	<i>Rating & review</i> pengguna terhadap restoran / kafe

2. Pemilik Restoran / Kafe (*Merchant*)

Tabel 3.3 Analisis Kebutuhan Pengguna - *Merchant*

Aktivitas	Kebutuhan Data	Kebutuhan Informasi
Pendaftaran dan pengelolaan data <i>merchant</i>	1. Data restoran & kafe 2. Data fasilitas restoran & kafe 3. Data pemilik restoran & kafe 4. Data rekening pemilik restoran & kafe	Daftar dan informasi restoran / kafe
Verifikasi dan pengelolaan data pemesanan tempat	1. Data pengguna 2. Data restoran & kafe 3. Data pemesanan tempat	Daftar dan informasi pemesanan tempat
Laporan transaksi pemesanan bulanan <i>merchant</i>	1. Data pemesanan 2. Data pengguna	Laporan transaksi pemesanan bulanan <i>merchant</i>
Pencairan dana	1. Data pemesanan 2. Data pembayaran	Informasi dana yang didapatkan dan akan dicairkan

Aktivitas	Kebutuhan Data	Kebutuhan Informasi
	3. Data saldo restoran & kafe	
	4. Data arus kas restoran & kafe	

3. Admin Picnicker

Tabel 3.4 Analisis Kebutuhan Pengguna - Admin

Aktivitas	Kebutuhan Data	Kebutuhan Informasi
Pembuatan akun admin	Data admin	Data diri admin
Pengelolaan data master	<ol style="list-style-type: none"> 1. Data wilayah 2. Data restoran dan kafe 3. Data pemilik restoran dan kafe 4. Data pengguna 5. Data kategori dan jenis restoran 6. Data karyawan 7. Data fasilitas 8. Data bank 	<ol style="list-style-type: none"> 1. Daftar restoran dan kafe 2. Daftar pemilik restoran dan kafe 3. Daftar karyawan (admin) 4. Daftar pengguna 5. Daftar kategori dan jenis restoran 6. Daftar fasilitas 7. Daftar bank
Verifikasi pendaftaran <i>merchant</i>	<ol style="list-style-type: none"> 1. Data restoran dan kafe 2. Data pemilik restoran dan kafe 3. Data rekening pemilik restoran dan kafe 	Daftar restoran dan kafe
Verifikasi pencairan dana <i>merchant</i>	<ol style="list-style-type: none"> 1. Data restoran dan kafe 2. Data pemilik restoran dan kafe 3. Data rekening pemilik restoran dan kafe 4. Data pemesanan 5. Data pembayaran 6. Data saldo restoran & kafe 	<ol style="list-style-type: none"> 1. Daftar restoran dan kafe 2. Daftar pemesanan dan pembayaran 3. Data saldo dan arus kas restoran dan kafe

Aktivitas	Kebutuhan Data	Kebutuhan Informasi
	7. Data arus kas restoran & kafe	
Laporan transaksi pemesanan bulanan	1. Data pemesanan 2. Data pengguna 3. Data <i>merchant</i>	Laporan transaksi pemesanan bulanan

E. Analisis Kebutuhan Non-Fungsional

Kebutuhan non-fungsional adalah suatu kebutuhan lain yang bukan merupakan fungsi utama sistem namun masih tetap dibutuhkan. Berikut analisis kebutuhan non-fungsional yang digunakan pada aplikasi ini:

1. Sistem Keamanan

Sistem keamanan merupakan salah satu sistem yang dibutuhkan untuk mengurangi resiko terjadinya kebocoran data atau penyalahgunaan hak akses, maka dari itu pada aplikasi Picnicker terdapat sistem keamanan menggunakan email dan *password* untuk setiap pengguna aplikasi.

2. *Responsive Design*

Responsive design merupakan suatu paham dimana *design* yang dibuat harus bisa menyesuaikan dengan perangkat yang digunakan. *Responsive design* dibutuhkan pada aplikasi Picnicker, agar pengguna Picnicker dapat mengakses halaman web aplikasi melalui bermacam-macam perangkat, seperti PC, *laptop*, *tablet* dan *smartphone*.

F. Analisis Sistem Aplikasi

Spesifikasi sistem yang dibutuhkan untuk mengembangkan aplikasi bagi pengguna (pelanggan, *merchant* dan admin Picnicker) dibagi menjadi dua yaitu spesifikasi perangkat keras dan spesifikasi perangkat lunak. Spesifikasi perangkat keras diperlukan untuk menjalankan sistem Picnicker yang dikembangkan. Sedangkan spesifikasi perangkat lunak digunakan untuk mengembangkan sistem yang dibangun dalam penelitian ini. Kebutuhan perangkat keras dan perangkat lunak dapat dilihat Tabel 3.5 berikut ini.

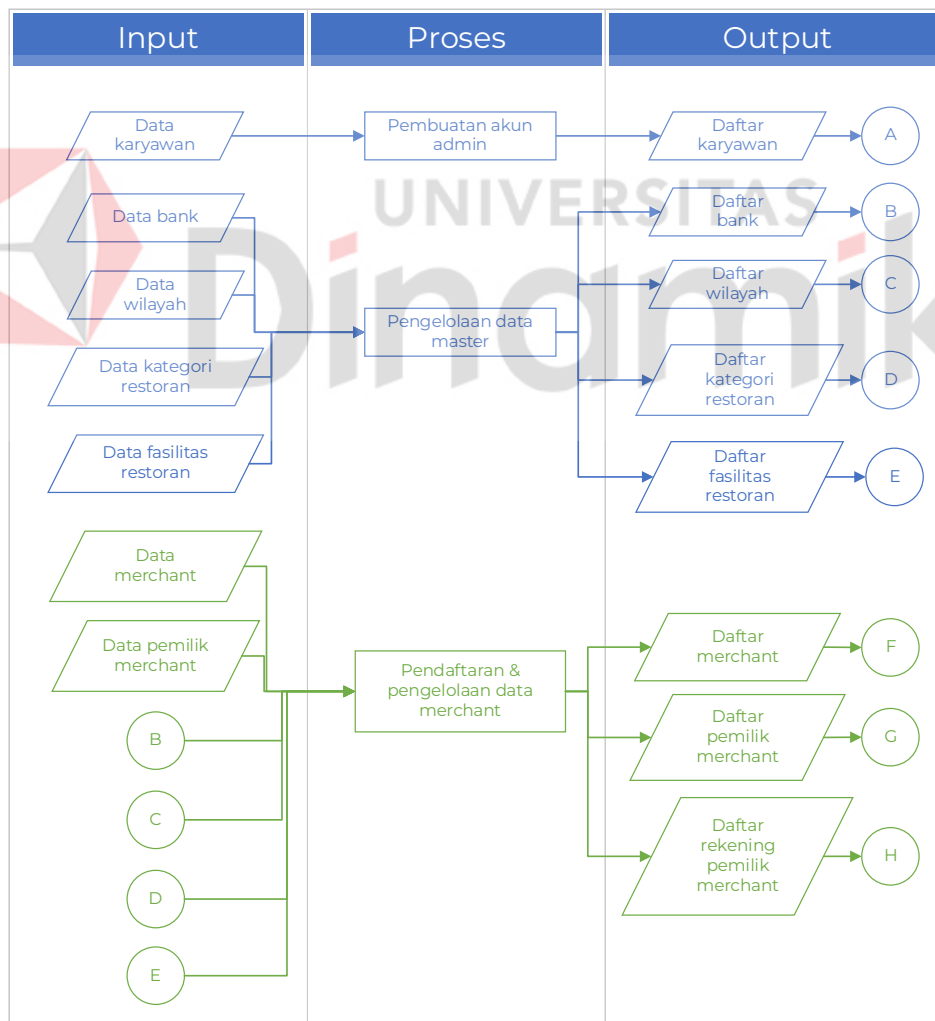
Tabel 3.5 Analisis Kebutuhan Sistem Aplikasi

Keterangan	Keterangan	Spesifikasi
------------	------------	-------------

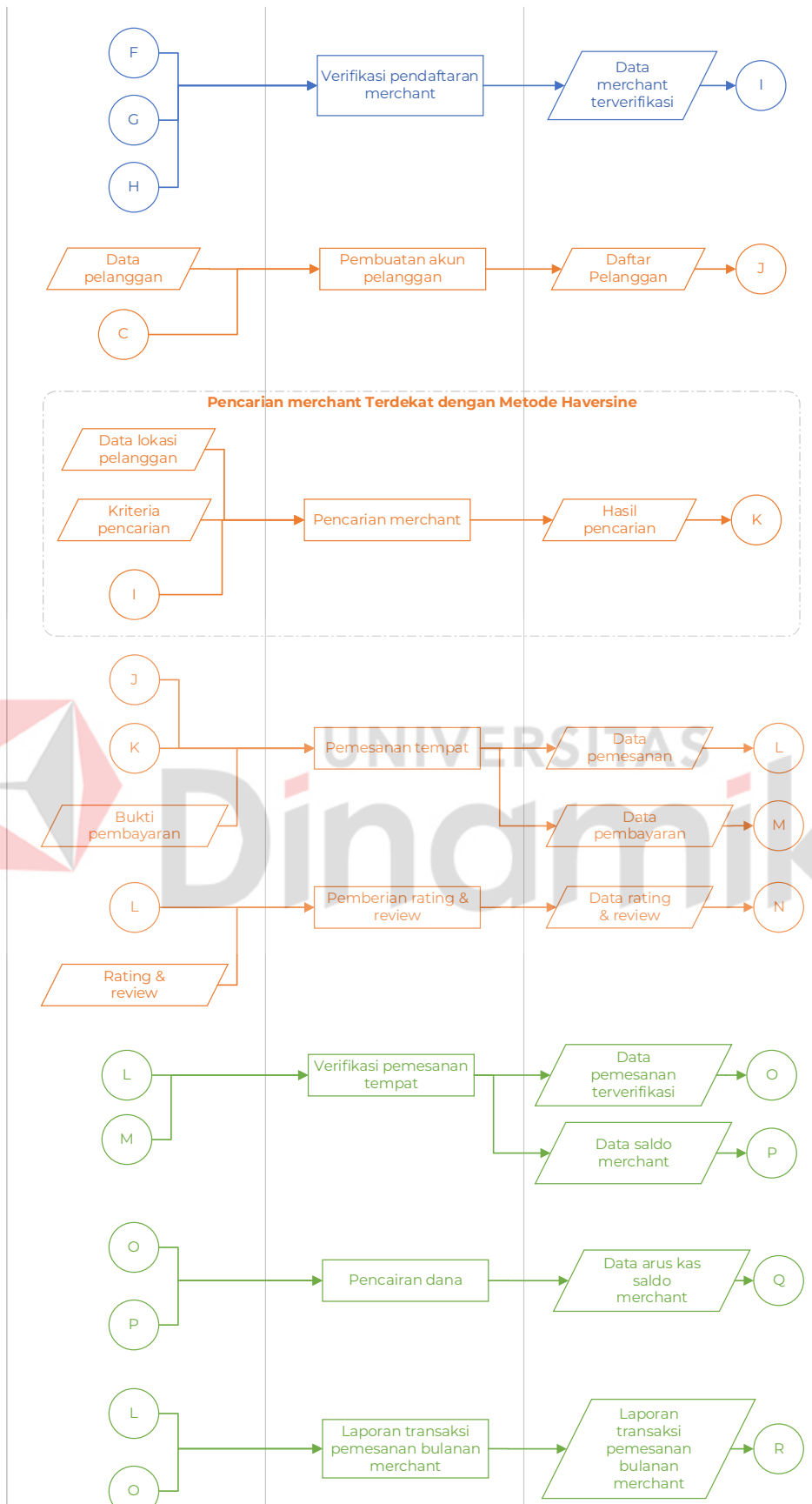
<i>Hardware</i>	<i>Processor</i>	Intel Pentium 4
	RAM	4 GB
	<i>Harddisk</i>	<i>Free Space</i> 40GB
<i>Software</i>	Sistem Operasi (OS)	Windows 7,8,10,11
	Web Browser	Chrome / Firefox / Edge
	Database	Maria DB / My SQL (XAMPP)
	Web Server	Apache (XAMPP)
	Text Editor	Visual Studio Code
	Lain-lain	Menginstall <i>dependency manager</i> “Composer” & <i>Framework</i> Laravel 9 (PHP 8.x)

3.3.2 Analisis Sistem

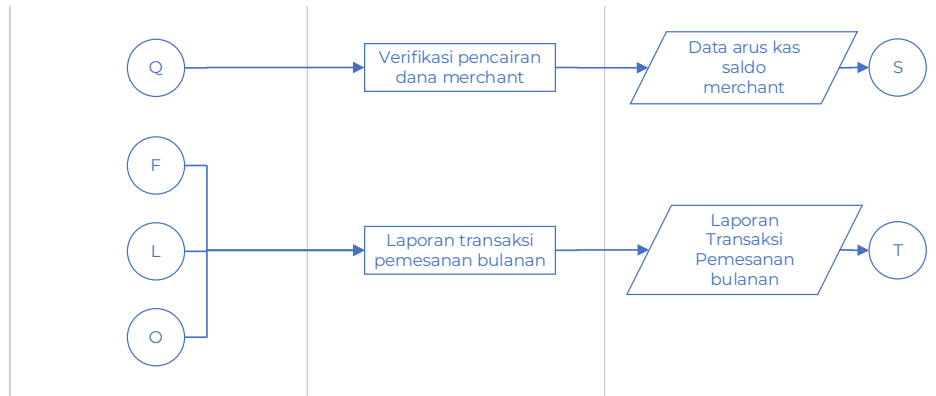
A. Diagram Input-Process-Output (IPO)



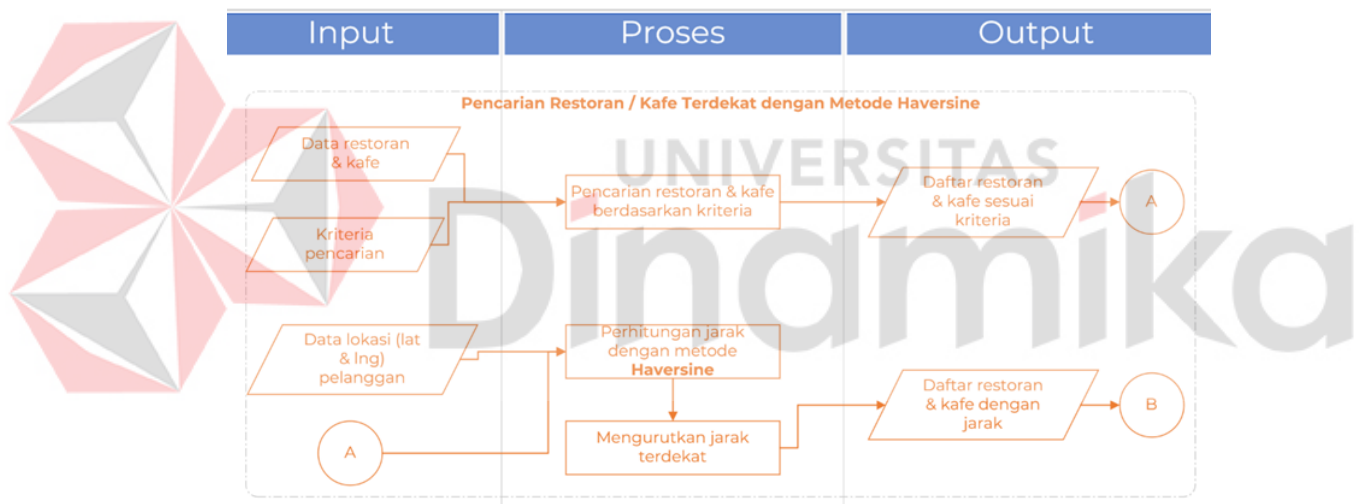
Gambar 3.3 Diagram IPO



Gambar 3.4 Diagram IPO Lanjutan



Gambar 3.5 Diagram IPO Lanjutan 2

Gambar 3.6 Diagram IPO Penerapan Metode *Haversine*

Berikut adalah penjelasan dari Gambar 3.6 diatas.

Input

1. Data restoran / kafe

Data ini berisi daftar restoran dan kafe yang sudah tercatat dalam sistem Picnicker.

2. Data kriteria *pencarian*

Data ini berisi kriteria pencarian yang dimasukkan atau dipilih oleh pengguna, kriteria ini dapat berupa kata kunci seperti nama restoran, ataupun kategori atau jenis restoran dan kafe.

3. Data lokasi

Data ini berisi data lokasi yang dipilih oleh pengguna berupa koordinat geografis *latitude* dan *longitude*.

4. Data restoran & kafe sesuai kriteria

Data ini berisi data restoran dan kafe yang sudah difilter atau dicari berdasarkan dengan kriteria yang telah dimasukkan atau dipilih oleh pengguna.

Proses

1. Pencarian restoran dan kafe berdasarkan kriteria

Proses ini melakukan pencarian menggunakan *query* pada *database* untuk menampilkan data restoran dan kafe yang sesuai dengan kriteria inputan pengguna.

2. Perhitungan jarak dengan metode *haversine*

Proses ini melakukan perhitungan metode *haversine* antara data lokasi (*latitude* & *longitude*) yang sudah dipilih oleh pengguna dengan data lokasi milik restoran / kafe yang didapatkan dari data restoran dan kafe berdasarkan kriteria.

3. Mengurutkan jarak terdekat

Proses ini akan mengurutkan hasil yang telah diperoleh dari proses sebelumnya (perhitungan jarak dengan metode *haversine*), data akan diurutkan berdasarkan jarak terdekat (kecil) ke jarak terjauh (besar).

Output

1. Daftar restoran & kafe sesuai kriteria

Data ini berisi data restoran dan kafe yang sudah difilter atau dicari berdasarkan dengan kriteria yang telah dimasukkan atau dipilih oleh pengguna.

2. Daftar restoran & kafe dengan jarak

Data ini berisi data restoran dan kafe yang sudah melalui proses perhitungan jarak dengan metode *haversine*, sehingga terdapat atribut baru

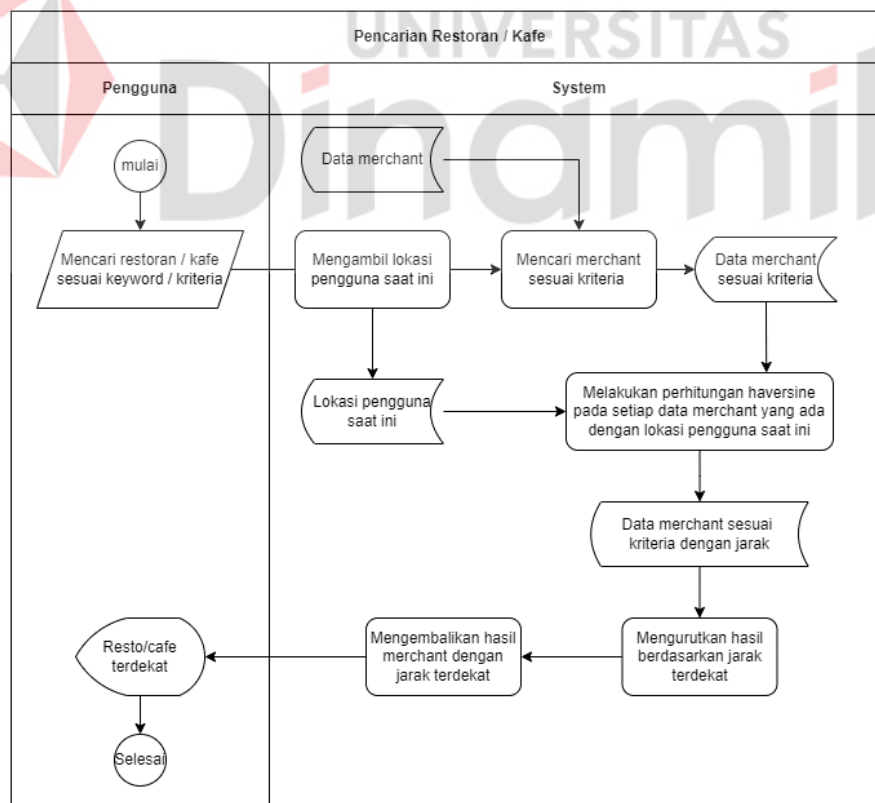
(jarak) pada data yang ditampilkan, dan data ini yang akan ditampilkan kepada pengguna.

B. System Flowchart (Sysflow)

System flow (*sysflow*) merupakan diagram alir yang menggambarkan kaitan antar proses dari awal hingga akhir yang dilakukan oleh pengguna / aktor tertentu pada sistem aplikasi yang dikembangkan.

Sysflow Pencarian Restoran / Kafe dengan Metode *Haversine*

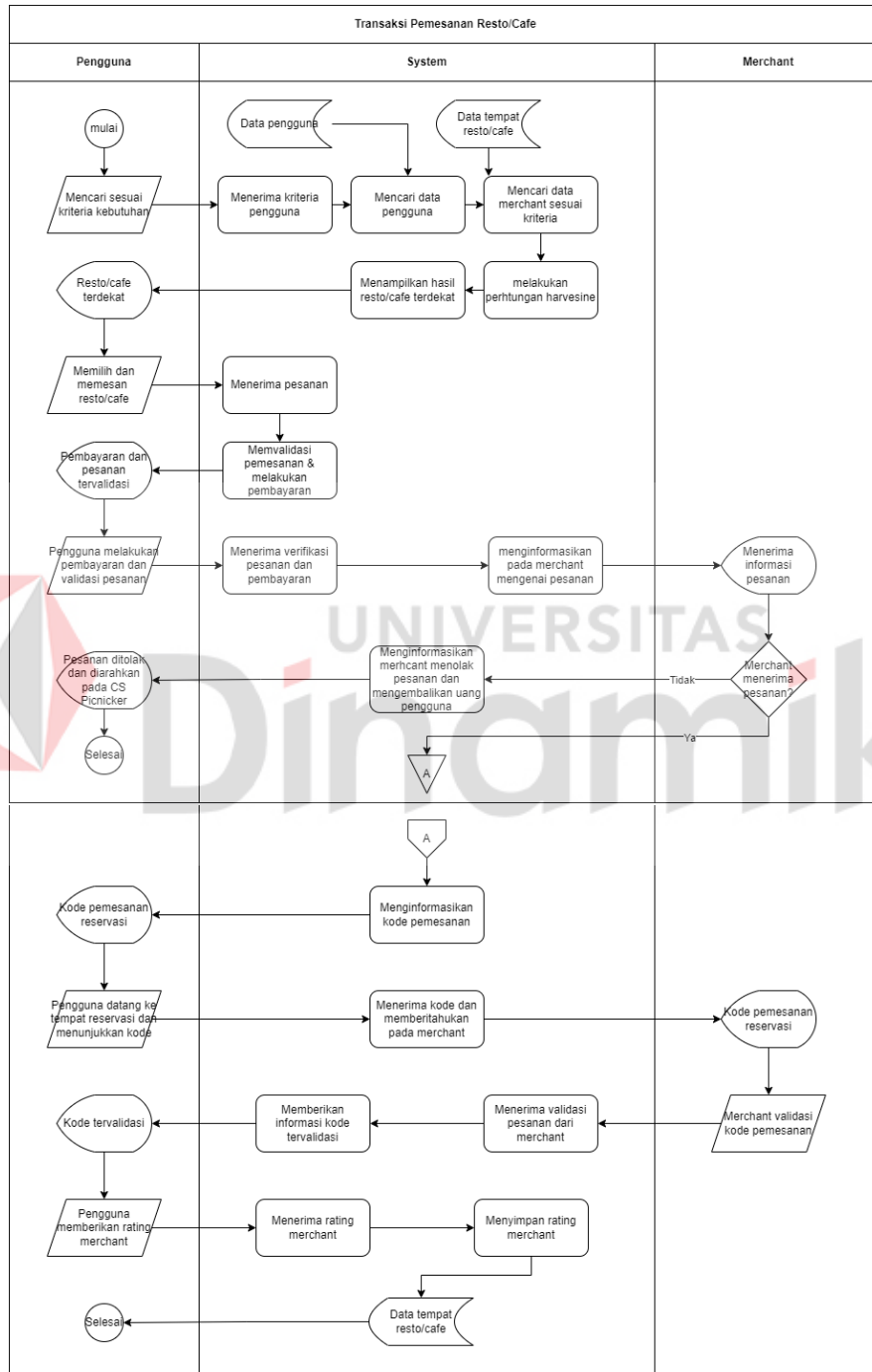
Sysflow pencarian restoran / kafe dengan metode *haversine* menggambarkan bagaimana proses atau alur dari fungsi perhitungan metode *haversine* yang digunakan pada saat pencarian restoran atau kafe terdekat berdasarkan kriteria atau kata kunci yang diinputkan pelanggan. *Sysflow* ini dapat dilihat pada Gambar 3.7 dibawah.



Gambar 3.7 *Sysflow* Pencarian Restoran / Kafe Dengan Metode *Haversine*

Sysflow Pencarian & Pemesanan Restoran / Kafe

Sysflow pencarian & pemesanan restoran / kafe menggambarkan bagaimana proses atau alur dari fitur pencarian dan pemesanan yang ada pada aplikasi Picnicker. Sysflow ini dapat dilihat pada Gambar 3.8 dibawah.

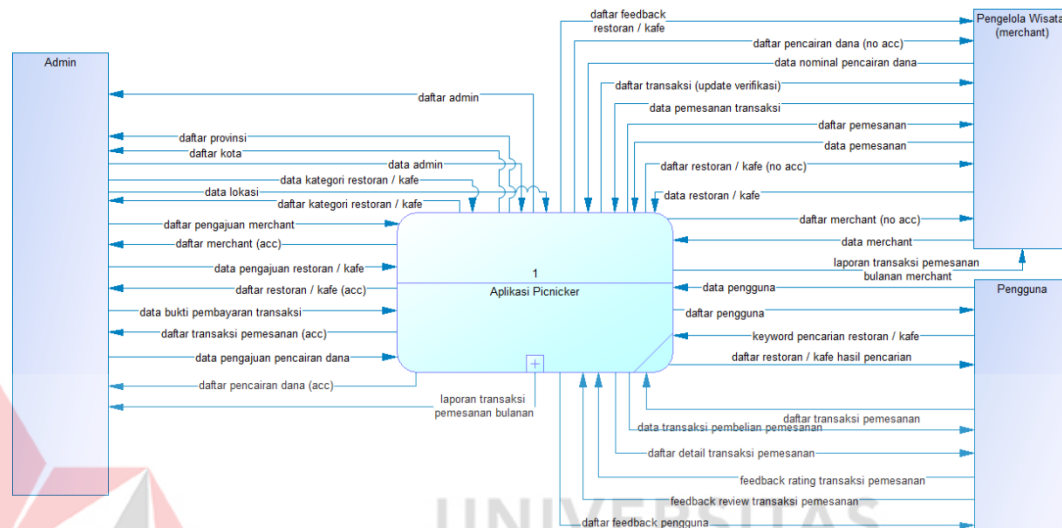


Gambar 3.8 Sysflow Pencarian & Pemesanan Restoran / Kafe

C. Data Flow Diagram (DFD)

1. Context Diagram

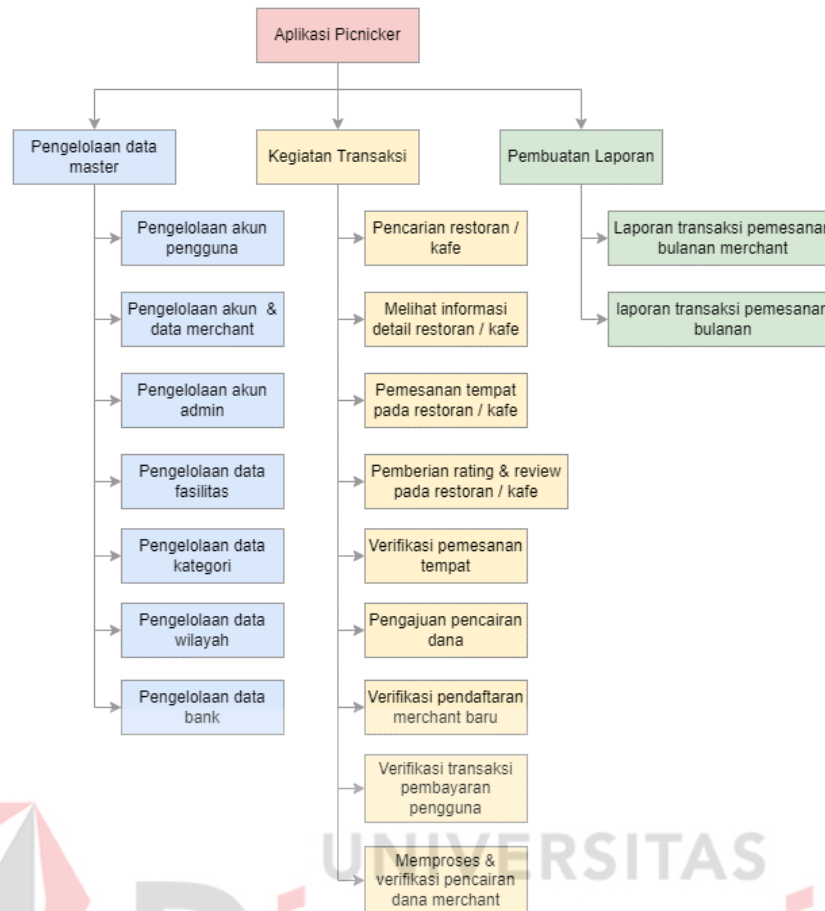
Context diagram atau diagram konteks menggambarkan keterlibatan *stakeholder* dalam kesatuan sistem yang dikembangkan dengan aliran lalu lintas data sebagai media komunikasi. Diagram konteks yang ada dalam penelitian ini dapat dilihat pada Gambar 3.9 dibawah ini.



Gambar 3.9 Context Diagram

2. HIPO Diagram

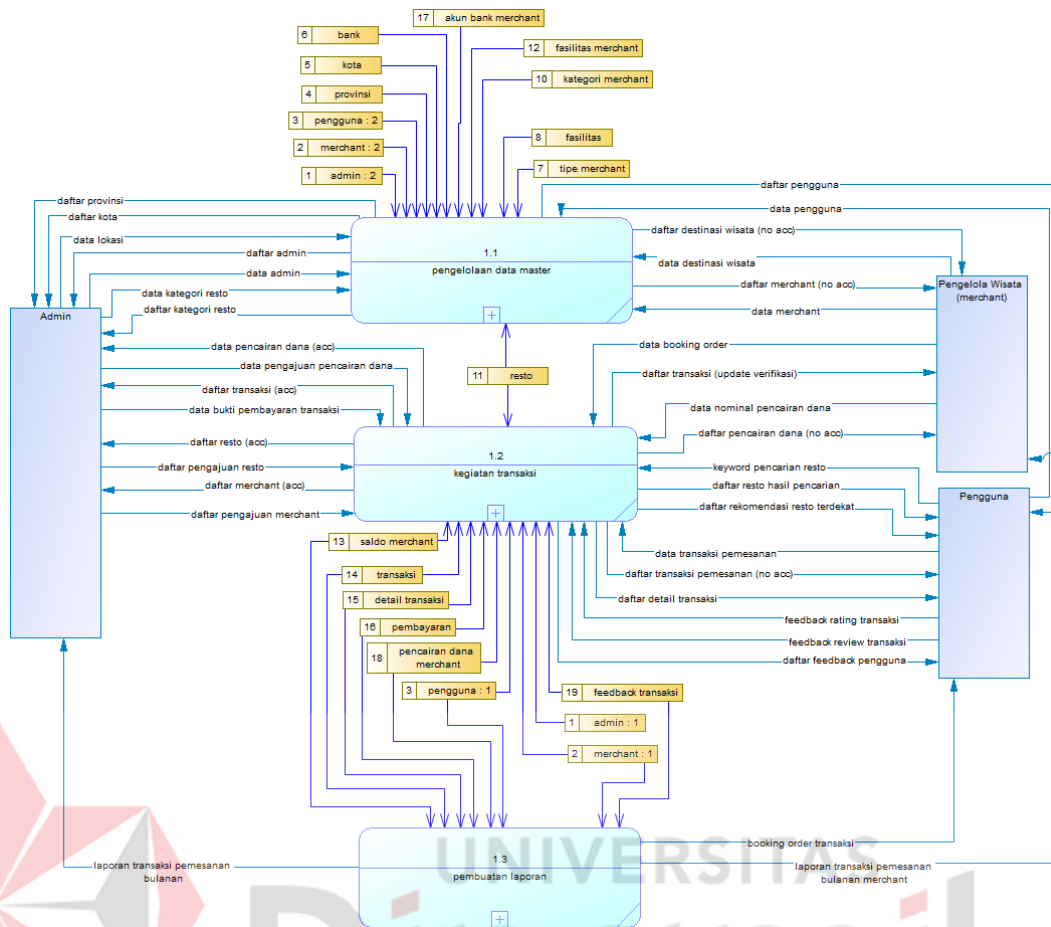
Diagram berjenjang atau *Hierarchical Input-Process-Output* (HIPO) menggambarkan *input*, proses dan *output* pada fungsional sistem yang dikembangkan melalui gambaran hierarki. Diagram HIPO yang menggambarkan sistem Picnicker terdiri atas pengelolaan data master, kegiatan transaksi dan pembuatan laporan dapat dilihat pada Gambar 3.10 dibawah ini.



Gambar 3.10 Diagram Berjenjang

3. Data Flow Diagram (DFD) Level 0

Data flow diagram (DFD) Level 0 merupakan diagram yang menggambarkan aliran data masuk dan keluar pada proses yang dilakukan oleh *stakeholder* dalam sebuah sistem informasi. Diagram DFD terkait dengan *context diagram* yang sudah dibuat sebelumnya namun memiliki level lebih detail. Dalam penelitian ini dibuat 2 level DFD yaitu level 0 dan 1. DFD level 0 menjelaskan gambaran besar aliran data yang terdiri dari proses pengelolaan data master, kegiatan transaksi dan pembuatan laporan. DFD level 1 menjelaskan lebih mendetail mengenai masing-masing proses secara spesifik yang dilakukan oleh masing-masing *stakeholder*. Diagram DFD level 0 dapat dilihat pada Gambar 3.11 berikut ini, dan untuk DFD level 1 dapat dilihat pada Lampiran 5.



Gambar 3.11 Data Flow Diagram

D. Entity Relationship Diagram (ERD)

Entity Relationship Diagram atau juga yang dikenal sebagai ERD merupakan model yang digunakan untuk menggambarkan relasi atau keterkaitan data dalam pengembangan aplikasi. ERD dibagi menjadi dua yaitu *Conceptual Data Model* berisikan notasi yang menggambarkan entitas yang digunakan dalam aplikasi dan kedua adalah *Physical Data Model* (PDM) yang menggambarkan model fisik dari penyimpanan data pada aplikasi.

1. *Conceptual Data Model* (CDM)

Model data konseptual atau *Conceptual Data Model* (CDM) merupakan diagram yang menggambarkan hubungan antar entitas data secara konsep. Dalam penelitian ini, entitas yang digunakan yaitu 19 entitas terbagi atas 12 tabel master dan 7 tabel transaksi. Diagram CDM dapat dilihat pada Lampiran 5.

2. *Physical Data Model (PDM)*

Model data fisik atau *Physical Data Model (PDM)* merupakan diagram yang menggambarkan implementasi hubungan antar entitas data yang sebelumnya sudah dibuat dalam bentuk model konseptual. Model data fisik yang dihasilkan dari model data konseptual adalah 21 tabel yang terdiri dari 12 tabel master dan 9 tabel transaksi. Diagram PDM dapat dilihat pada Lampiran 6.

E. *Desain User Interface*

Desain Antarmuka Pengguna / *User Interface (UI)* merupakan desain tampilan aplikasi yang dibuat untuk nantinya diimplementasikan pada penelitian ini. Adapun gambaran mengenai desain tampilan aplikasi Picnicker dapat dilihat pada Lampiran 8.

F. *Desain Testing*

Testing atau Uji coba merupakan salah satu tahap yang dilakukan dalam penelitian ini untuk mengetahui hasil dari solusi yang dikembangkan untuk menyelesaikan masalah. Dalam penelitian ini menggunakan metode *black box* dengan parameter sebagai berikut:

BAB IV

HASIL DAN IMPLEMENTASI

4.1 Implementasi Sistem

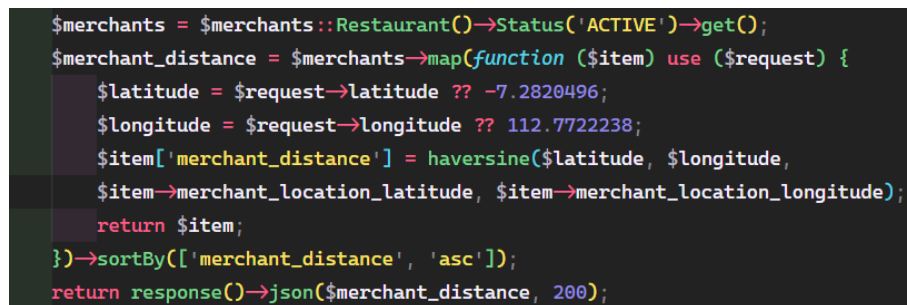
4.1.1 Penerapan Metode *Haversine*

Penerapan metode *haversine* dalam penelitian ini dilakukan saat pengguna mencari atau melihat daftar restoran atau kafe yang ada, kemudian sistem akan menampilkan daftar restoran atau kafe terdekat yang berada di sekitar pengguna berdasarkan kriteria yang di pilih atau di-inputkan oleh pengguna. Implementasi metode metode *haversine* dilakukan dengan contoh data dapat dilihat pada di Lampiran 4.



```
function haversine($lat1, $lon1, $lat2, $lon2)
{
  // current_pos
  $lat1 = deg2rad($lat1);
  $lon1 = deg2rad($lon1);
  // destination
  $lat2 = deg2rad($lat2);
  $lon2 = deg2rad($lon2);
  // earth radius const
  $earth_radius = 6371;
  // differences
  $diff_lat = $lat2 - $lat1;
  $diff_lon = $lon2 - $lon1;
  // calculation
  $distance_1 = sin($diff_lat / 2) ** 2 +
    cos($lat1) *
    cos($lat2) *
    sin($diff_lon / 2) ** 2;
  $distance = round($earth_radius * (2 * asin(sqrt($distance_1))), 3);
  return $distance;
};
```

Gambar 4.2 Fungsi Metode *Haversine*



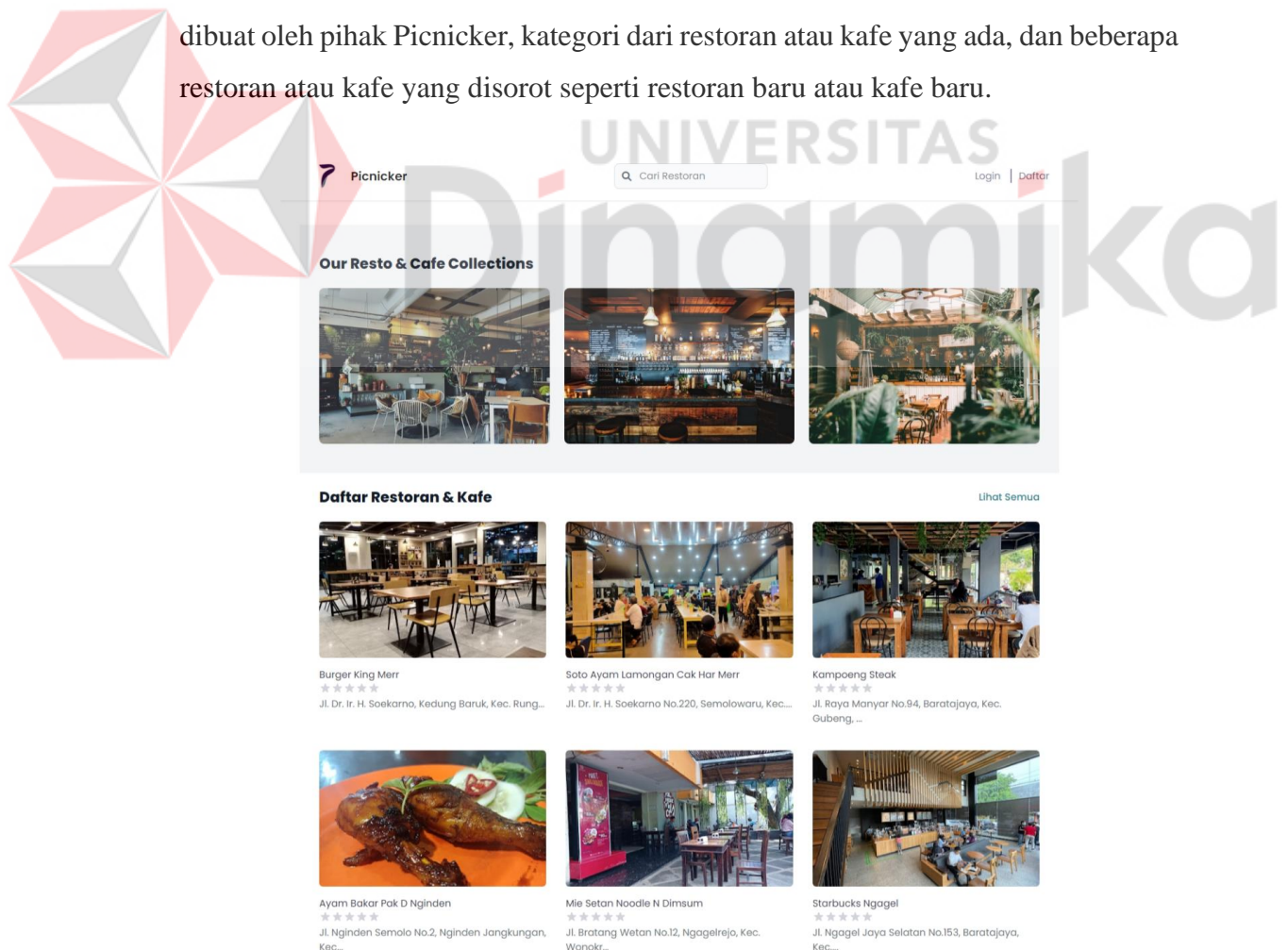
```
$merchants = $merchants::Restaurant()->Status('ACTIVE')->get();
$merchant_distance = $merchants->map(function ($item) use ($request) {
  $latitude = $request->latitude ?? -7.2820496;
  $longitude = $request->longitude ?? 112.7722238;
  $item['merchant_distance'] = haversine($latitude, $longitude,
    $item->merchant_location_latitude, $item->merchant_location_longitude);
  return $item;
})->sortBy(['merchant_distance', 'asc']);
return response()->json($merchant_distance, 200);
```

Gambar 4.1 Penerapan Metode *Haversine*

Penerapan metode *haversine* dilakukan setelah pengguna memilih lokasi saat ini dan memberi inputan kriteria atau kata kunci yang dicari. Kemudian sistem akan melakukan pencarian melalui *query* kedalam *database* berdasarkan kriteria atau kata kunci tersebut, seperti yang dapat dilihat pada Gambar 4.1. Hasil dari pencarian tersebut akan diolah oleh sistem menggunakan fungsi pada Gambar 4.2, kemudian sistem akan mengembalikan atau mengarahkan pengguna ke halaman hasil pencarian atau daftar restoran dan kafe dengan data hasil olahan yang sudah diurutkan berdasarkan jarak terdekat antara lokasi pengguna dengan restoran dan kafe yang ada.

4.1.2 Halaman Awal Pada Sisi Pelanggan

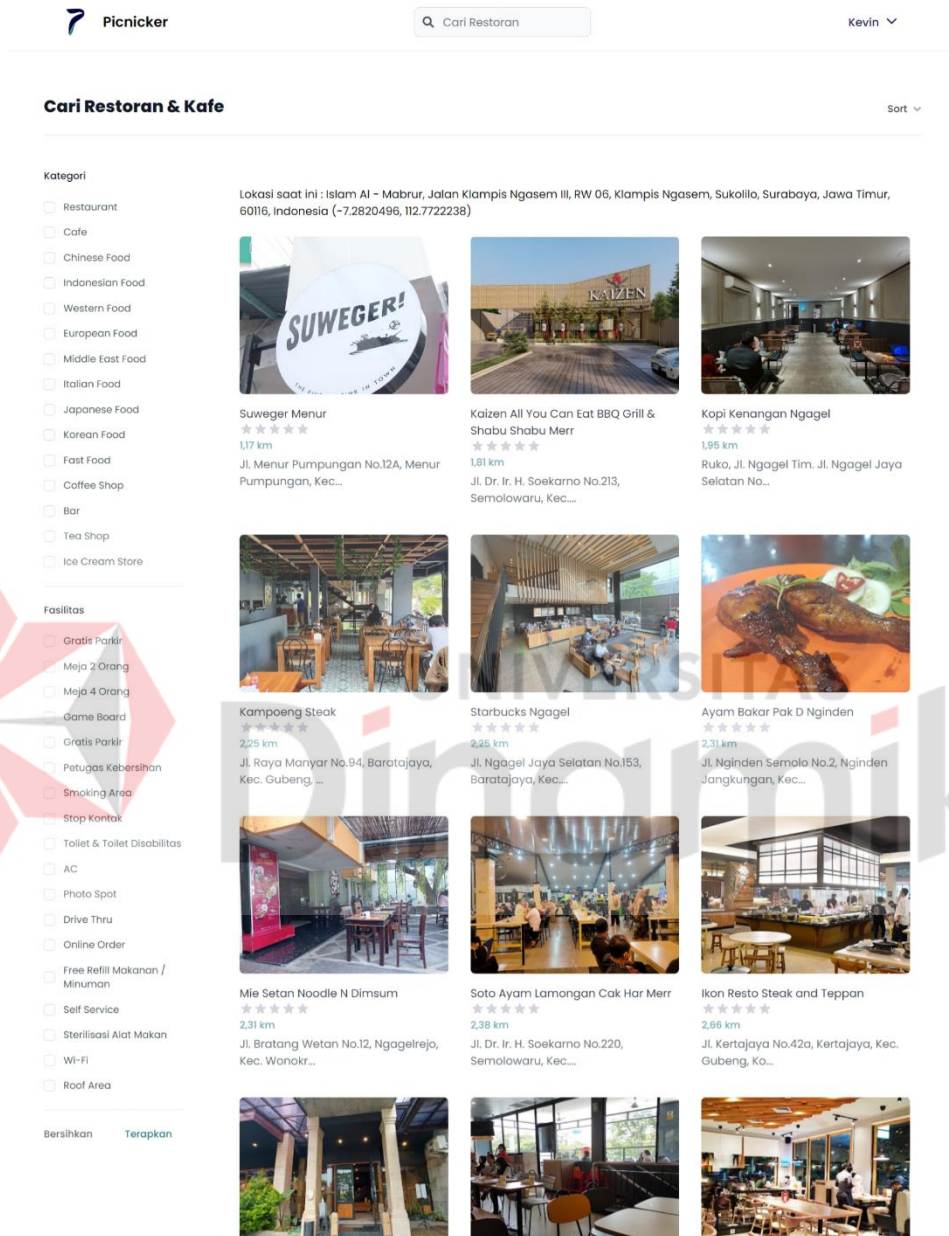
Halaman awal aplikasi memuat beberapa komponen inti dari aplikasi, yang tertera pada Gambar 4.3 dibawah ini. Halaman awal memuat *banner-banner* yang dibuat oleh pihak Picnicker, kategori dari restoran atau kafe yang ada, dan beberapa restoran atau kafe yang disorot seperti restoran baru atau kafe baru.



Gambar 4.3 Halaman Utama Sisi Pelanggan

4.1.3 Halaman Pencarian *Merchant* Pada Sisi Pelanggan

Halaman pencarian *merchant* memuat daftar *merchant* sesuai dengan hasil pencarian dan filter yang diterapkan, yang tertera pada Gambar 4.4 dibawah ini.

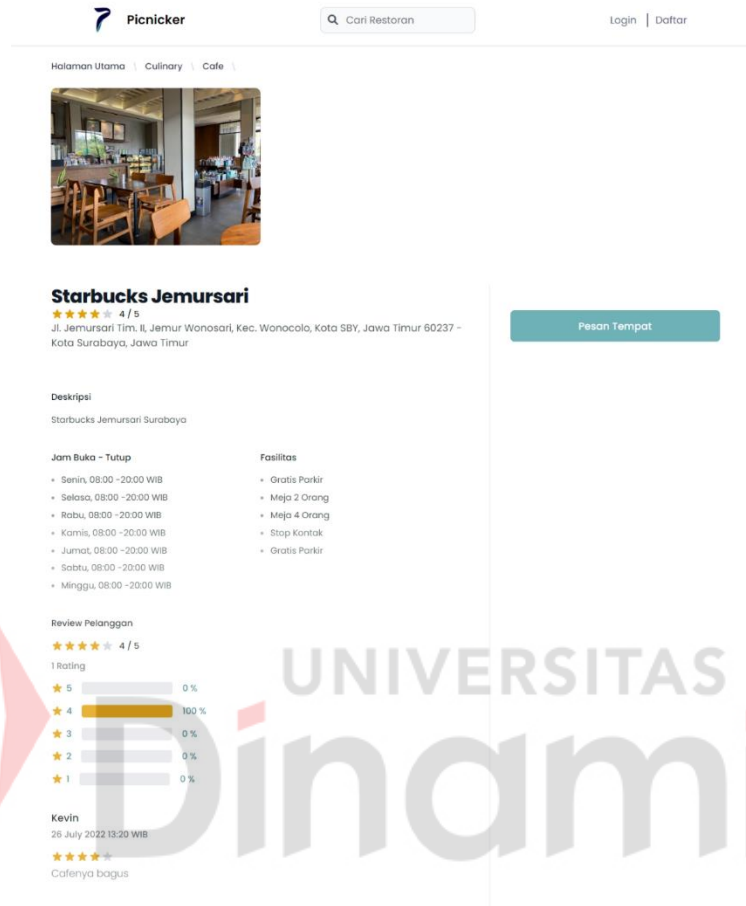


Gambar 4.4 Halaman Pencarian Sisi Pelanggan

4.1.4 Halaman *Merchant* Pada Sisi Pelanggan

Halaman *merchant* memuat detail dari *merchant* (restoran & kafe) yang ada pada aplikasi, yang tertera pada Gambar 4.5 dibawah ini. Pada halaman ini akan informasikan data-data yang dimiliki oleh *merchant*, seperti fasilitas yang

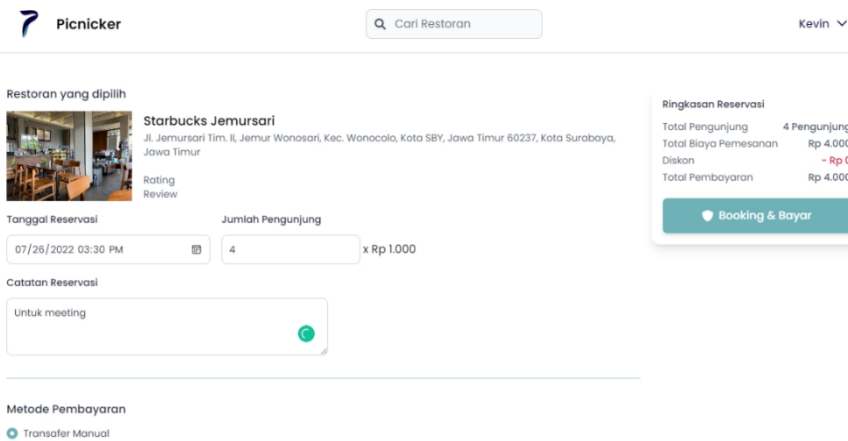
disediakan oleh *merchant*, kategori *merchant*, foto *merchant*, jam buka-tutup *merchant* yang ada, jarak dari lokasi pengguna saat ini dengan *merchant*, *rating* dan *review* dari pelanggan, dan tombol untuk melakukan pemesanan.



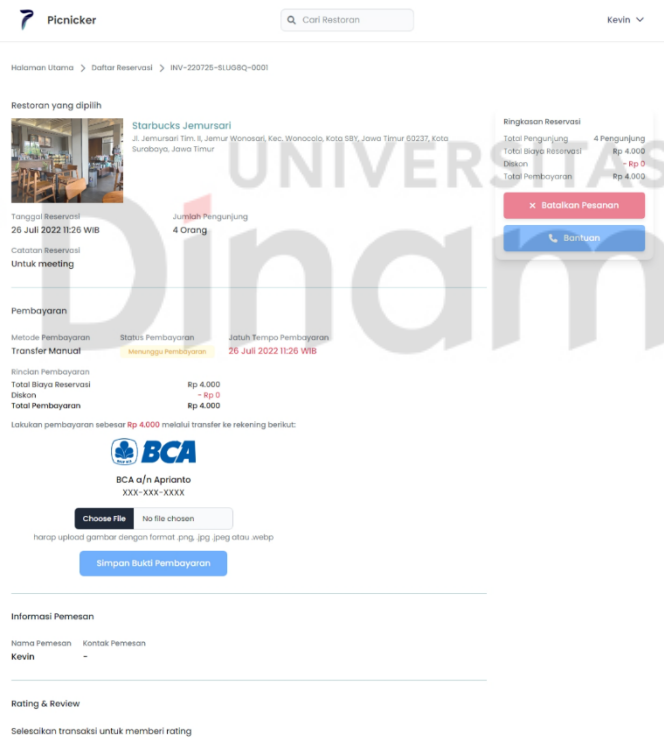
Gambar 4.5 Halaman Merchant Sisi Pelanggan

4.1.5 Halaman Pemesanan Pada Sisi Pelanggan

Halaman pemesanan merupakan halaman untuk melakukan reservasi kepada *merchant* yang dipilih oleh pengguna, dimana pada halaman ini pengguna akan menginputkan data seperti tanggal reservasi, jumlah pengunjung, catatan pemesanan dan *upload* bukti pembayaran, serta apabila transaksi sudah selesai maka pengguna dapat memberikan *rating* & *review* kepada *merchant* yang di kunjungi. Halaman pemesanan dapat dilihat pada Gambar 4.6 dan 4.7 dibawah.



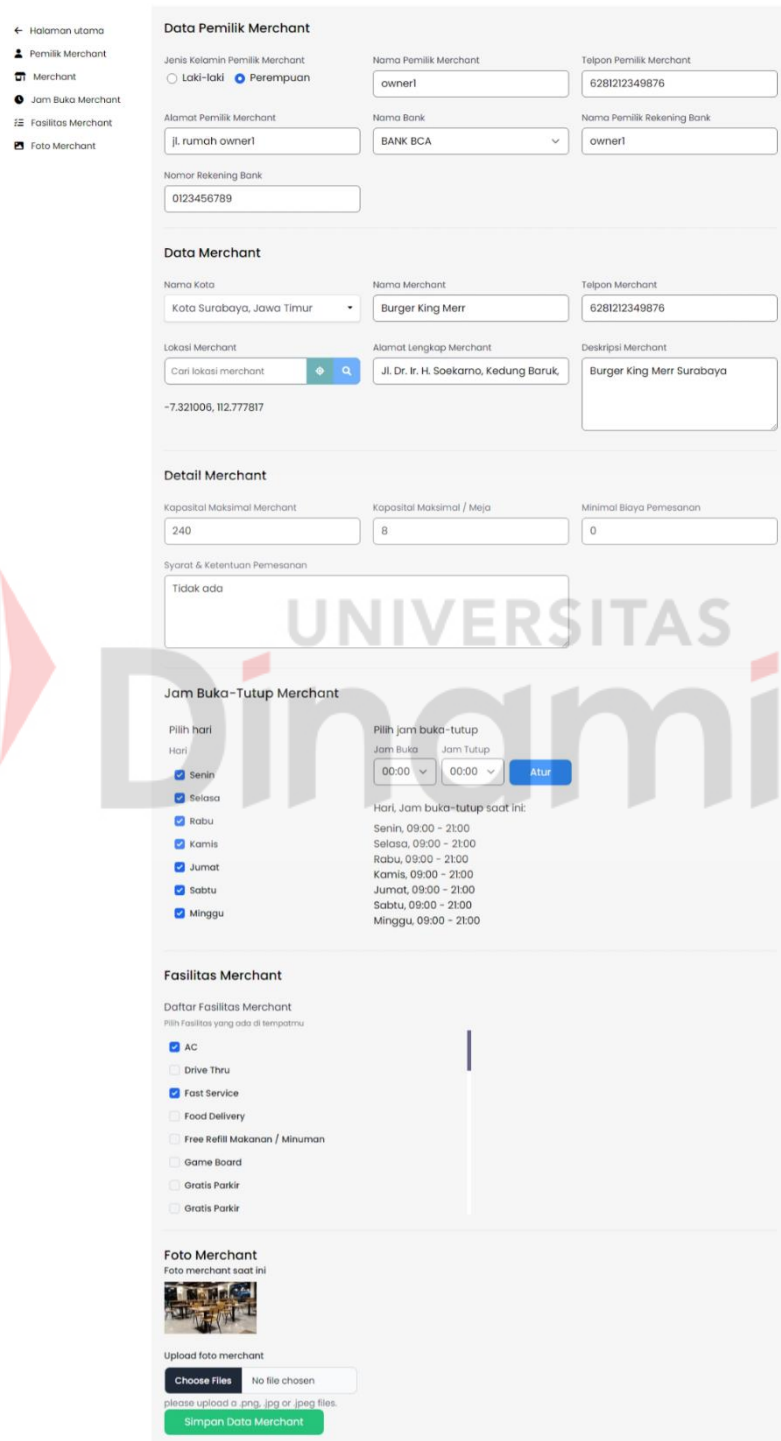
Gambar 4.6 Halaman Pemesanan Sisi Pelanggan



Gambar 4.7 Halaman Detail Pemesanan Sisi Pelanggan

4.1.6 Halaman Pendaftaran *Merchant* Pada Sisi *Merchant*

Halaman pendaftaran *merchant* memuat atribut-atribut data yang perlu *merchant* isi untuk melengkapi informasi detail *merchant*, yang tertera pada Gambar 4.8 dibawah ini.



Data Pemilik Merchant

Jenis Kelamin Pemilik Merchant
 Laki-laki Perempuan

Nama Pemilik Merchant: owner1
Telpon Pemilik Merchant: 6281212349876

Alamat Pemilik Merchant: Jl. rumah owner1
Nama Bank: BANK BCA
Nama Pemilik Rekening Bank: owner1

Nomor Rekening Bank: 0123456789

Data Merchant

Nama Kota: Kota Surabaya, Jawa Timur
Nama Merchant: Burger King Merr
Telpon Merchant: 6281212349876

Lokasi Merchant: Cari lokasi merchant
Alamat Lengkap Merchant: Jl. Dr. Ir. H. Soekarno, Kedung Baruk
Deskripsi Merchant: Burger King Merr Surabaya

-7.321006, 112.777817

Detail Merchant

Kapasitas Maksimal Merchant: 240
Kapasitas Maksimal / Meja: 8
Minimal Biaya Pemesanan: 0

Syarat & Ketentuan Pemesanan: Tidak ada

Jam Buka-Tutup Merchant

Pilih hari: Senin, Selasa, Rabu, Kamis, Jumat, Sabtu, Minggu

Pilih jam buka-tutup: Jam Buka: 00:00, Jam Tutup: 00:00, Atur

Hari, Jam buka-tutup saat ini:
Senin, 09:00 - 21:00
Selasa, 09:00 - 21:00
Rabu, 09:00 - 21:00
Kamis, 09:00 - 21:00
Jumat, 09:00 - 21:00
Sabtu, 09:00 - 21:00
Minggu, 09:00 - 21:00

Fasilitas Merchant

Daftar Fasilitas Merchant
Pilih Fasilitas yang ada di tempatmu

AC
 Drive Thru
 Fast Service
 Food Delivery
 Free Refill Makanan / Minuman
 Game Board
 Gratis Parkir
 Gratis Parkir

Foto Merchant

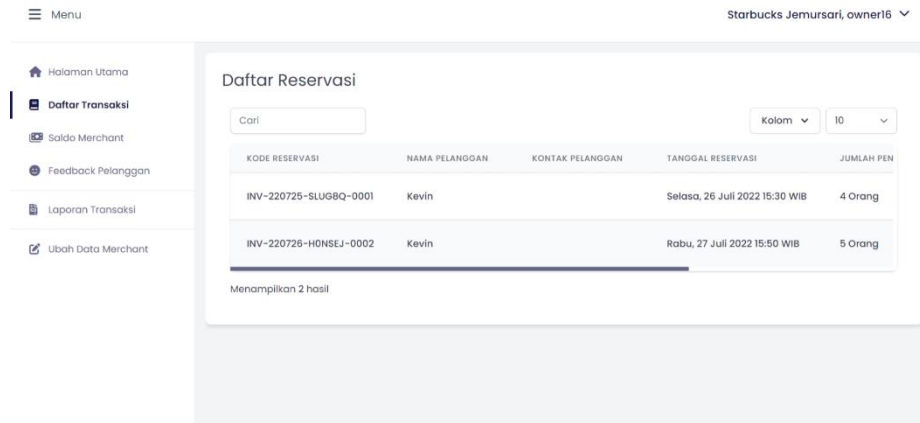
Foto merchant saat ini

Upload foto merchant
Choose Files No file chosen
please upload a .png, .jpg or .jpeg files.
Simpan Data Merchant

Gambar 4.8 Halaman Transaksi Pemesanan Sisi *Merchant*

4.1.7 Halaman Transaksi Pemesanan Pada Sisi Merchant

Halaman transaksi pemesanan *merchant* memuat daftar transaksi pemesanan yang terjadi pada *merchant* tersebut, yang tertera pada Gambar 4.9 dibawah ini.

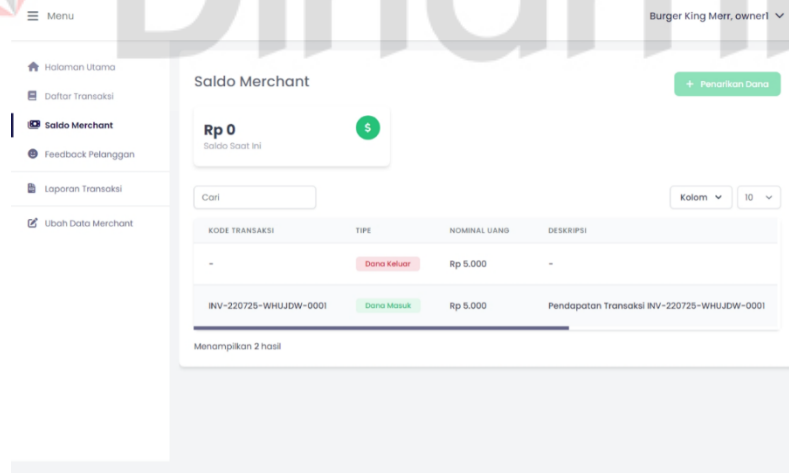


KODE RESERVASI	NAMA PELANGGAN	KONTAK PELANGGAN	TANGGAL RESERVASI	JUMLAH PEN
INV-220725-SLUG8Q-0001	Kevin		Selasa, 26 Juli 2022 15:30 WIB	4 Orang
INV-220726-HNSEJ-0002	Kevin		Rabu, 27 Juli 2022 15:50 WIB	5 Orang

Gambar 4.9 Halaman Transaksi Pemesanan Sisi Merchant

4.1.8 Halaman Pencairan Dana Pada Sisi Merchant

Halaman pencairan dana *merchant* memuat arus kas dan jumlah dana yang *merchant* dapatkan dan dapat di cairkan, yang tertera pada Gambar 4.10 dibawah ini.



KODE TRANSAKSI	TIFE	NOMINAL UANG	DESKRIPSI
-	Dana Keluar	Rp 5.000	-
INV-220725-WHUJDW-0001	Dana Masuk	Rp 5.000	Pendapatan Transaksi INV-220725-WHUJDW-0001

Gambar 4.10 Halaman Pencairan Dana Sisi Merchant

4.1.9 Halaman Laporan Pemesanan Pada Sisi Merchant

Halaman laporan pemesanan *merchant* memuat rangkuman / laporan dari transaksi yang terjadi pada *merchant* dalam periode bulan, yang tertera pada Gambar 4.11 dibawah ini.

CUSTOMER QUANTITY	TRANSACTION QUANTITY	TRANSACTION ITEM QUANTITY	TRANSACTION GRAND TOTAL
1 Customer	1 Pemesanan	5 Items	Rp 5.000

Gambar 4.11 Halaman Laporan Pemesanan Sisi *Merchant*

4.1.10 Halaman Pengelolaan Data Master Pada Sisi Admin

Halaman pengelolaan data master admin memuat fitur-fitur yang dibutuhkan untuk mengelola data master yang digunakan pada aplikasi Picicker sesuai dengan identifikasi data yang sudah dilakukan. Salah satu halaman pengelolaan data master dapat dilihat pada Gambar 4.12 dibawah ini.

FACILITY NAME	MERCHANT TYPE	STATUS	ACTION
AC	RESTAURANT	ACTIVE	[Edit] [Delete]
Drive Thru	RESTAURANT	ACTIVE	[Edit] [Delete]
Fast Service	RESTAURANT	ACTIVE	[Edit] [Delete]
Food Delivery	RESTAURANT	ACTIVE	[Edit] [Delete]

Gambar 4.12 Halaman Pengelolaan Data Master Sisi Admin

4.1.11 Halaman Transaksi Pemesanan Pada Sisi Admin

Halaman transaksi pemesanan admin memuat daftar transaksi pemesanan yang terjadi, yang tertera pada Gambar 4.13 dibawah ini.

KODE RESERVASI	JUMLAH PENUNJUNG	STATUS RESERVASI	STATUS PEMBAYARAN	ACTION
INV-220708-66DP-0001	5 Orang	Reservasi Baru	Sudah Terbayar	[Edit] [Delete]
INV-220708-JVBA-0003	2 Orang	Reservasi Baru	Sudah Terbayar	[Edit] [Delete]
INV-220708-KLIG-0002	8 Orang	Sudah Terverifikasi	Sudah Terbayar	[Edit] [Delete]
INV-220713-JIQD-0001	5 Orang	Reservasi Baru	Sudah Terbayar	[Edit] [Delete]

Gambar 4.13 Halaman Transaksi Pemesanan Sisi Admin

4.1.12 Halaman Pencairan Dana Pada Sisi Admin

Halaman pencairan dana admin memuat daftar arus kas dan permintaan pencairan dana dari *merchant*, yang tertera pada Gambar 4.14 dibawah ini.

KODE TRANSAKSI	TIFE	NOMINAL UANG	DESKRIPSI
-	Dana Keluar	Rp 5.000	-
INV-220725-WHUJDW-0001	Dana Masuk	Rp 5.000	Pendapatan Transaksi INV-220725-WHUJDW-0001

Gambar 4.14 Halaman Pencairan Dana Sisi Admin

4.1.13 Halaman Laporan Pemesanan Pada Sisi Admin

Halaman laporan pemesanan admin memuat rangkuman / laporan dari transaksi yang terjadi dalam periode bulan, yang tertera pada Gambar 4.15 dibawah ini.

MERCHANT QUANTITY	CUSTOMER QUANTITY	TRANSACTION QUANTITY	TRANSACTION ITEM QUANTITY	TRANSA
4 Merchant	1 Customer	7 Transaction	31 Items	Rp 31,010

Gambar 4.15 Halaman Laporan Pemesanan Sisi Admin

4.2 Hasil Pengujian

Pengujian sistem dalam penelitian ini dilakukan dengan melakukan metode *black box*.

4.2.1 Hasil Pengujian Metode *Haversine*

Pada pengujian metode *haversine*, lokasi pengguna saat ini disimulasikan pada jl. Mleto no 12, dengan nilai *latitude* sebesar -7.2820496 dan nilai *longitude* sebesar 112.7722238 kemudian dikonversikan dalam satuan radians maka akan

didapatkan nilai *latitude* sebesar -0.127095742 dan nilai *longitude* sebesar 1.96824661. Pengujian akan dilakukan terhadap 4 destinasi dengan detail pada Tabel 4.1.

Tabel 4.1 Pengujian Metode *Haversine*

Nama / Keterangan	Latitude	Longitude	Latitude (Radians)	Longitude (Radians)
Lokasi saat ini	-7.2820496	112.7722238	-0.127095742	1.96824661
Starbucks Ngagel	-7.2948409	112.7563840	-0.127318992	1.967970153
Coffee Toffee Merr	-7.2599110	112.7824945	-0.12670935	1.968425868
Starbucks Jemursari	-7.3207121	112.7450169	-0.12777053	1.96777176
Excelso Jemursari	-7.3141664	112.7554457	-0.127656286	1.967953777

Dengan menggunakan metode *haversine* berikut, maka didapatkan hasil:

$$d = 6371 * 2 * \text{asin}(\sqrt{\sin^2((\text{lat}2 - \text{lat}1)/2) + \cos(\text{lat}1) * \cos(\text{lat}2) * \sin^2((\text{lon}2 - \text{lon}1)/2)})$$

Lokasi saat ini → Starbucks Ngagel

$$= 6371 * 2 * \text{asin}(\sqrt{\sin^2((-0.127319 - -0.127096)/2)^2 + \cos(-0.127096) * \cos(-0.127319) * \sin^2((1.96797 - 1.968247)/2)^2})$$

$$= 2.253 \text{ km}$$

Lokasi saat ini – Coffee Toffee Merr

$$= 6371 * 2 * \text{asin}(\sqrt{\sin^2((-0.126709 - -0.127096)/2)^2 + \cos(-0.127096) * \cos(-0.126709) * \sin^2((1.968426 - 1.968247)/2)^2})$$

$$= 2.710 \text{ km}$$

Lokasi saat ini – Starbucks Jemursari

$$= 6371 * 2 * \text{asin}(\sqrt{\sin^2((-0.127771 - -0.127096)/2)^2 + \cos(-0.127096) * \cos(-0.127771) * \sin^2((1.967772 - 1.968247)/2)^2})$$

$$= 5.243 \text{ km}$$

Lokasi saat ini – Excelso Jemursari

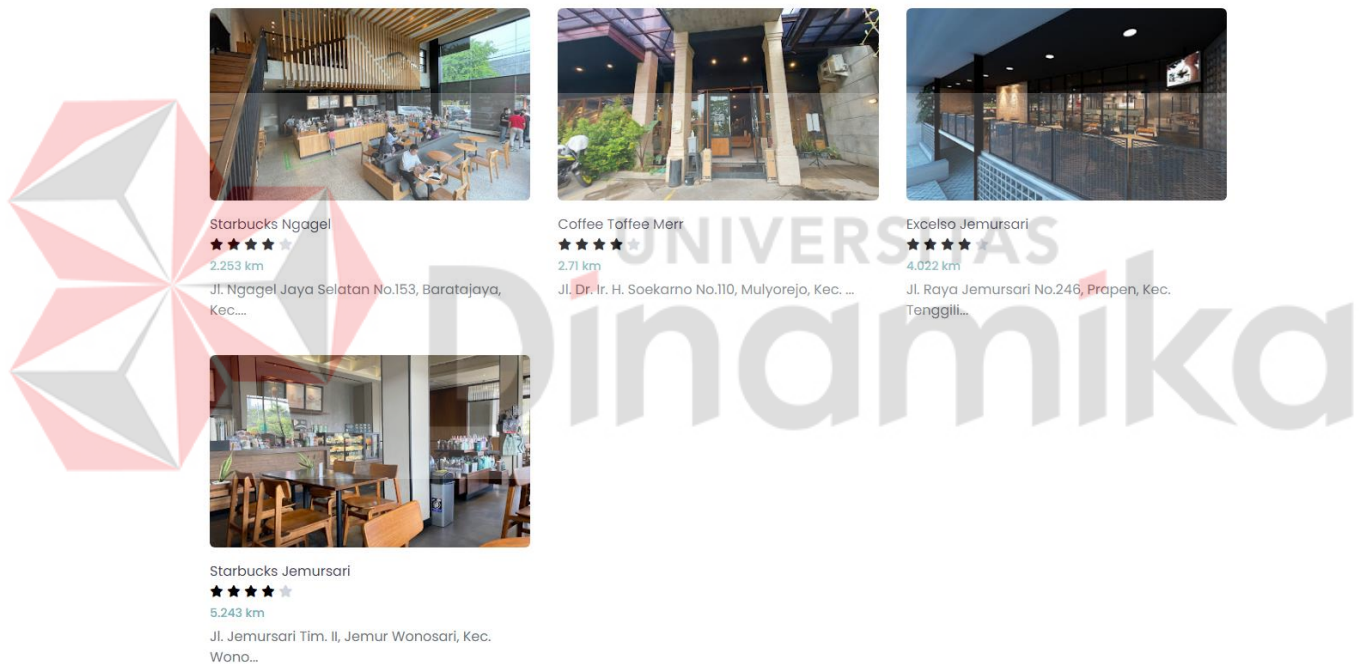
$$= 6371 * 2 * \text{asin}(\text{sqrt}(\sin((-0.127656 - -0.127096)/2)^2 + \cos(-0.127096) * \cos(-0.127656) * \sin((1.967954 - 1.968247)/2)^2))$$

$$= 4.022 \text{ km}$$

Berdasarkan pengujian diatas, didapatkan hasil sebagai berikut:

- Lokasi Terdekat : Starbucks Ngagel 2.253 km
 Lokasi Terjauh : Starbucks Jemursari 5.243 km

Seperti yang dapat dilihat pada Gambar 4.16 hasil dari pengujian diatas sama dengan hasil pada aplikasi Picnicker, dimana aplikasi Picnicker sudah mengurutkan data tersebut sesuai dengan jarak terdekat hingga terjauh.



Gambar 4.16 Hasil Pengujian Metode haversine

4.3 Evaluasi Sistem

Evaluasi pada sistem dalam penelitian ini dilakukan untuk menentukan tingkat akurasi atau ketepatan dari metode *haversine* yang digunakan untuk mencari restoran atau kafe terdekat dari lokasi pengguna.

4.3.1 Evaluasi Penerapan Metode *Haversine*

Berdasarkan hasil pengujian yang dilakukan sebelumnya antara perhitungan manual dengan fungsi yang di implementasikan pada sistem didapati bahwa hasil dari empat sampel yang digunakan sama. Selain itu juga dilakukan evaluasi perbandingan dengan menggunakan jarak tempuh terdekat yang disarankan oleh Google Maps pada sampel data yang tertera pada Lampiran 4 dengan lokasi pengguna saat ini yang disimulasikan pada jl. Mleto no 12, dengan nilai *latitude* sebesar -7.2820496 dan nilai *longitude* sebesar 112.7722238 kemudian dikonversikan dalam satuan radians maka akan didapatkan nilai *latitude* sebesar -0.127095742 dan nilai *longitude* sebesar 1.96824661.

Dengan pengujian menggunakan metode *haversine* didapatkan hasil sebagai berikut:

Jarak tempuh terdekat metode *haversine* : 1.2 km
Jarak tempuh terjauh metode *haversine* : 5.2 km
Rata-rata jarak tempuh metode *haversine* : 3.2 km

Dengan menggunakan layanan Google Maps jarak tempuh yang didapatkan adalah sejauh:

Jarak tempuh terdekat Google Maps : 1.4 km
Jarak tempuh terjauh Google Maps : 6.1 km
Rata-rata jarak tempuh Google Maps : 3.9 km

Setelah mendapatkan kedua hasil tersebut, maka akan dicari selisihnya dan mendapatkan selisih sebagai berikut:

Selisih jarak tempuh terdekat : 0.1 km
Selisih jarak tempuh terjauh : 1.3 km
Rata-rata selisih : 0.7 km

Berdasarkan selisih yang didapatkan, dapat disimpulkan bahwa penerapan metode *haversine* yang digunakan sudah cukup akurat, hal ini dibuktikan dengan tingkat akurasi (rata-rata jarak tempuh metode *haversine* / rata-rata jarak tempuh Google Maps) sebesar 83% dan selisih yang didapatkan dimana selisih antara jarak

perhitungan *haversine* dengan jarak yang direkomendasikan Google Maps masih cukup kecil (0.1 – 1.3km) dengan rata-rata sejauh 0.7km atau 700m yang masih dapat di toleransi. Dengan demikian penerapan metode *haversine* dapat membantu menyelesaikan permasalahan jarak terdekat yang dialami oleh pengguna

4.3.2 Evaluasi Pembangunan Sistem

Dengan adanya aplikasi Picnicker, permasalahan yang dialami oleh pengguna dapat teratasi, yang sebelumnya pengguna harus melakukan pencarian dan pemesanan restoran atau kafe secara manual, rumit dan membutuhkan waktu yang lama, sekarang dapat melakukan secara *online*, mudah, dan cepat seperti yang digambarkan pada Tabel 4.2 dibawah ini:

Tabel 4.2 Evaluasi Sistem

Cara Konvensional	Menggunakan Aplikasi Picnicker
1. Membuka aplikasi Google Maps	1. Membuka aplikasi Picnicker
2. Memasukkan kata kunci atau kriteria pencarian pada kolom pencarian	2. Memasukkan kata kunci atau kriteria pencarian pada kolom pencarian (nama tempat, kota, nama kategori)
3. Memilih salah satu tempat yang muncul dari hasil pencarian	3. Memilih salah satu tempat yang muncul dari hasil pencarian
4. Melihat informasi yang ada pada tempat tersebut.	4. Melihat informasi detail seperti fasilitas-fasilitas yang disediakan oleh restoran / kafe yang ada pada tempat tersebut. Pengguna dapat kembali ke halaman hasil pencarian dan memilih tempat yang lain apabila merasa kurang sesuai.
5. Apabila merasa kurang puas dengan informasi yang disediakan serta tidak menemukan kontak tempat tersebut, sehingga perlu mencari pada platform lain seperti Instagram, <i>official website</i> , blog <i>review</i> makanan, dan aplikasi yang lain.	5. Melakukan pemesanan kepada tempat yang dipilih, dan membayar biaya pemesanan.
6. Mendapatkan kontak tempat tersebut dan menanyakan fasilitas yang disediakan, serta informasi mengenai pemesanan	6. Datang ke tempat tersebut dan menunjukkan kode pemesanan untuk menyelesaikan pemesanan.
7. Mengulangi proses 4-6 apabila informasi dan hasil dari proses 6 kurang memuaskan.	
8. Menghubungi kembali tempat yang diinginkan dan melakukan pemesanan.	

9. Datang ke tempat tersebut dan memvalidasi pemesanan.	
Estimasi waktu: 10-30 menit	Estimasi waktu: 4-8 menit

Hasil pada Tabel 4.2 diatas menunjukan bahwa metode pemesanan konvensional membutuhkan proses dan estimasi waktu yang lebih lama dibandingkan dengan menggunakan aplikasi Picnicker. Dengan menggunakan aplikasi Picnicker diperkirakan waktu pencarian dapat 2.5x – 3.75x lebih cepat dibandingkan dengan metode pemesanan konvensional.

Selain evaluasi diatas juga dilakukan evaluasi menggunakan survei kepada 38 responden dengan hasil yang ada pada Lampiran 9 hasil survei ini akan digunakan untuk menentukan apakah aplikasi Picnicker bermanfaat bagi penggunanya. Hasil dari survei ini membuktikan bahwa 84% pengguna atau responden merasa terbantu dan puas dalam menggunakan aplikasi Picnicker dengan detail sebagai berikut:

1. Sebesar 81% responden merasa fitur pencarian (nama, kategori, filter, fasilitas) informasi restoran / kafe lebih mudah dan cepat dibandingkan cara konvensional.
2. Sebesar 85% responden merasa fitur pemesanan pada aplikasi Picnicker sangat mudah dan membantu responden dalam pemesanan restoran / kafe.
3. Sebesar 87% responden merasa informasi restoran / kafe cukup detail dan menjawab kebutuhan / pertanyaan mereka.
4. Sebesar 81% responden merasa informasi jarak pada aplikasi Picnicker membantu responden dalam memilih lokasi restoran / kafe yang ingin dikunjungi.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan dari hasil, implementasi, dan evaluasi yang sudah dilakukan pada pembuatan sistem Picnicker, dapat disimpulkan sebagai berikut:

1. Dengan menggunakan metode *haversine* pengguna dapat lebih mudah dalam mencari dan memesan restoran dan kafe yang ada di sekitar pengguna. Metode *haversine* juga dapat diandalkan karena setelah dilakukan evaluasi perbandingan antara metode *haversine* dengan jarak menggunakan Google Maps didapatkan tingkat akurasi sebesar 83% dan selisih jarak yang ada cukup kecil yaitu 0.1km hingga rata-rata sejauh 0.7km sehingga nilai ini masih dapat ditoleransi.
2. Berdasarkan dari hasil evaluasi pembangunan sistem yang dilakukan, aplikasi Picnicker dapat menyelesaikan masalah pengguna dengan menjadikan pengalaman pencarian dan pemesanan restoran / kafe menjadi lebih cepat hingga 3.5x lipat dibandingkan dengan cara pemesanan konvensional.
3. Fitur-fitur yang terdapat pada aplikasi Picnicker terbukti mampu membantu mengatasi kebutuhan pengguna berdasarkan hasil kuisisioner dan percobaan yang diikuti oleh 38 pengguna, dimana 84% pengguna yang mencoba merasa terbantu dan puas dengan adanya aplikasi Picnicker untuk melakukan pencarian dan pemesanan pada restoran / kafe yang ada.

5.2 Saran

Aplikasi Picnicker yang dikembangkan tentu saja masih belum sempurna dan memiliki banyak kekurangan terutama pada bagian tampilan atau *UI-UX* yang digunakan, sehingga ke depannya untuk meningkatkan kualitas dari aplikasi Picnicker maka diperlukan desain tampilan atau *UI-UX* yang lebih baik agar dapat menarik lebih banyak pengguna dan meningkatkan kenyamanan pengguna dalam menggunakan aplikasi Picnicker.

DAFTAR PUSTAKA

- Hakim, A. R. (2022, March 7). *Menko Luhut: Jabodetabek dan Surabaya Raya Turun Jadi PPKM Level 2* | merdeka.com.
<https://www.merdeka.com/uang/menko-luhut-jabodetabek-dan-surabaya-raja-turun-jadi-ppkm-level-2.html>
- Hartanto, S., Furqan, M., Siahaan, A. P. U., & Fitriani, W. (2017). Haversine Method in Looking for the Nearest Masjid. *International Journal of Recent Trends in Engineering and Research*, 3(8), 187–195.
<https://doi.org/10.23883/ijrter.2017.3402.pd61h>
- Haryanto, A. (2022, March). *Aturan Perjalanan Dilonggarkan, Pariwisata Lembang KBB Kembali Menggeliat* : Okezone Travel.
<https://travel.okezone.com/read/2022/03/23/406/2566167/aturan-perjalanan-dilonggarkan-pariwisata-lembang-kbb-kembali-menggeliat>
- Miftahuddin, Y., Umaroh, S., & Karim, F. R. (2020). Perbandingan Metode Perhitungan Jarak Euclidean, Haversine, Dan Manhattan Dalam Penentuan Posisi Karyawan. *Jurnal Tekno Insentif*, 14(2), 69–77.
<https://doi.org/10.36787/jti.v14i2.270>
- Mursyidah, M., & Hidayat, H. T. (2017). Pengujian Sistem Informasi Akuntansi Biaya Operasional Sekolah Dengan Black Box Testing. *Jurnal Infomedia: Teknik Informatika, Multimedia & Jaringan*, 2(2). <http://e-jurnal.pnl.ac.id/infomedia/article/view/512>
- Nursyafitri, G. D. (2021, October). *Kenali Tentang Statistik Parametrik: Pengertian, Syarat, Kel...* Dqlab. <https://dqlab.id/kenali-tentang-statistik-parametrik-pengertian-syarat-kelebihan-dan-kekurangannya>
- Pratama, A. M. (2022, February 14). *Asosiasi Pengusaha Kafe & Restoran: Penerapan PPKM Semestinya Sudah Tidak Diterapkan Halaman all* - Kompas.com. KOMPAS.
<https://money.kompas.com/read/2022/02/14/211456926/asosiasi-pengusaha-kafe-restoran-penerapan-ppkm-semestinya-sudah-tidak?page=all>

- Rafie, B. T. (2021, November 8). *WFO mulai kembali diterapkan, catat 5 persiapan yang harus dilakukan*. <https://newssetup.kontan.co.id/news/wfo-mulai-kembali-diterapkan-catat-5-persiapan-yang-harus-dilakukan>
- Steven Dharmawan, W., Purwaningtias, D., Risdiansyah, D., Studi Sistem Informasi, P., PSDKU Pontianak, U., Abdurrahman Saleh No, J., & Barat, K. (2018). *Penerapan Metode SDLC Waterfall Dalam Perancangan Sistem Informasi Administrasi Keuangan Berbasis Desktop*. VI(2).
- Suwarna, I. K., & Widyatmaja, I. G. N. (2017). *Pengetahuan Dasar Ilmu Pariwisata*.
- Utami, S. (2018). *Kuliner Sebagai Identitas Budaya: Perspektif Komunikasi Lintas Budaya*. *CoverAge: Journal of Strategic Communication*.
- Yulianto, W. (2015). *Menentukan Jarak Terdekat Hotel Dengan Metode Haversine Formula*.
- Yulianto, Y., Ramadiani, R., & Kridalaksana, A. H. (2018). *Penerapan Formula Haversine Pada Sistem Informasi Geografis Pencarian Jarak Terdekat Lokasi Lapangan Futsal* (Vol. 13, Issue 1).



UNIVERSITAS
Dinamika