

BAB II

LANDASAN TEORI

2.1 Tinjauan Terhadap Penelitian Yang Terkait

Berikut ini adalah beberapa hasil penelitian yang terkait dan menjadi referensi pada penelitian ini :

1. Jumlah kendaraan di kota-kota besar di Indonesia seperti Denpasar dan sekitarnya semakin banyak dan bertambah tiap tahunnya. Bertambahnya jumlah kendaraan tersebut, memiliki dampak terhadap kebutuhan parkir di tempat-tempat umum seperti di kantor, pusat perbelanjaan, sekolah, kampus, tempat rekreasi, dan tempat-tempat umum lainnya yang memiliki area parkir yang cukup luas. Kampus STIKOM Bali yang beralamat di Jalan Raya Puputan No. 86 Renon Denpasar Bali mempunyai beberapa lahan parkir di areal gedung yaitu *basement parking* dan *outdoor parking*. Pada saat beberapa perkuliahan sedang berlangsung bersamaan, seringkali areal parkir menjadi penuh sesak karena kendaraan mahasiswa tidak di tempatkan secara teratur, banyak kendaraan yang menghalangi akses keluar masuk dan tata letak parkir kendaraan yang kurang optimal membuat parkir areal basement terkesan tidak terorganisir, Rekomendasi yang bisa di sarankan untuk mengurangi beban permasalahan parkir kampus adalah dengan penataan areal parkir semaksimal mungkin untuk menampung sebanyak kendaraan mahasiswa sehingga resiko *overcapacity* pada areal parkir dapat dikurangi. Model tata letak parkir yang akan di rancang akan memetakan lahan parkir dimana basement parking tersebut akan di pecah menjadi blok parkir yang kemudian di bagi lagi menjadi *parking lots* untuk menentukan

daya tampung maksimal kendaraan, kemudian model parkir tersebut akan di rancang untuk memenuhi aspek yang di kehendaki. Dalam penentuan model tata letak parkir menggunakan algoritma *Ant Colony System* (ACS) dimana algoritma ini dapat mencari tata letak yang optimal mengenai tujuan dari sistem ini (Jayanti, Ni Ketut Dewi Ari, and Ni Made Dewi Kansa Putri. 2019).

2. Perkembangan yang terjadi pada kota Batam sangat mempengaruhi jumlah kunjungan para wisatawan dari dalam atau luar negeri Mereka memilih berlibur ke Batam dengan alasan Batam merupakan surganya belanja , karena Batam merupakan wilayah FTZ atau *FreeTaxZone* dimana barang yang ada di Batam tidak dikenakan pajak. Wisatawan seringkali tidak mempunyai banyak waktu dan biaya untuk mengunjungi lokasi wisata secara sekaligus, agar tidak mengeluarkan biaya terlalu besar atau memakan waktu banyak di jalan karena tidak mengerti rute atau jalan mana yang akan ditempuh. Pemilihan rute perjalanan wisata dan berbelanja yang baik dapat meminimalisir hal –hal yang tidak diinginkan oleh para wisatawan seperti tersesat atau kehilangan arah. Salah satu cara untuk menanggulangnya, permasalahan akan dibahas menggunakan algoritma *Ant Colony* yang nantinya akan membantu para wisatawan yang akan menuju ketempat wisata atau tempat berbelanja di kota Batam, Kepulauan Riau (Meldra, Delia. 2019).

2.2 Artificial Intelligence (AI)

Menurut (Dr. Hendra, dkk. 2018) Kecerdasan buatan atau *Artificial Intelligence* (AI) merupakan salah satu bagian ilmu komputer yang membuat agar

mesin (komputer) dapat melakukan pekerjaan seperti dan sebaik yang dilakukan oleh manusia. Pada awal diciptakannya, komputer hanya difungsikan sebagai alat hitung saja. Namun seiring dengan perkembangan jaman, maka peran komputer semakin mendominasi kehidupan umat manusia. Komputer tidak lagi hanya digunakan sebagai alat hitung, lebih dari itu, komputer diharapkan untuk dapat diberdayakan untuk mengerjakan segala sesuatu yang bisa dikerjakan oleh manusia.

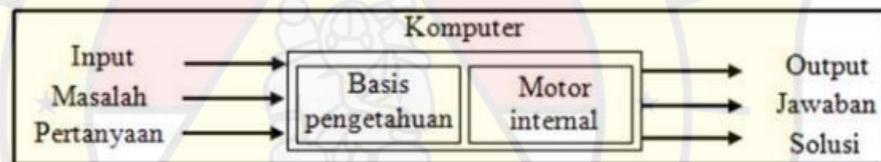
Manusia bisa menjadi pandai dalam menyelesaikan segala permasalahan di dunia ini karena manusia mempunyai pengetahuan dan pengalaman pengetahuan diperoleh dari belajar. Semakin banyak bekal pengetahuan yang dimiliki oleh seseorang tentu saja diharapkan akan lebih mampu dalam menyelesaikan permasalahan. Namun bekal pengetahuan saja tidak cukup, manusia juga diberi akal untuk melakukan penalaran, mengambil kesimpulan berdasarkan pengetahuan dan pengalaman yang mereka miliki. Tanpa memiliki kemampuan untuk menalar dengan baik, manusia dengan segudang pengalaman dan pengetahuan tidak akan dapat menyelesaikan masalah dengan baik. Demikian pula, dengan kemampuan menalar yang sangat baik, namun tanpa bekal pengetahuan dan pengalaman yang memadai, manusia juga tidak akan bisa menyelesaikan masalah dengan baik.

Lebih detilnya, kecerdasan buatan dapat dipandang dari berbagai sudut pandang, antara lain:

1. Sudut pandang kecerdasan Kecerdasan buatan akan membuat mesin menjadi 'cerdas' (mampu berbuat seperti apa yang dilakukan oleh manusia).
2. Sudut pandang penelitian Kecerdasan buatan adalah suatu studi bagaimana membuat agar komputer dapat melakukan sesuatu sebaik yang dikerjakan oleh manusia.

3. Sudut pandang bisnis Kecerdasan buatan adalah kumpulan peralatan yang sangat powerful dan metodologis dalam menyelesaikan masalah-masalah bisnis.
4. Sudut pandang pemrograman Kecerdasan buatan meliputi studi tentang pemrograman simbolik, penyelesaian masalah (*problem solving*) dan pencarian (*searching*). Untuk melakukan aplikasi kecerdasan buatan ada 2 bagian utama yang sangat dibutuhkan, yaitu (Gambar 2.1):

- Basis pengetahuan (*knowledge base*), berisi fakta-fakta, teori, pemikiran dan hubungan antara satu dengan lainnya.
- Motor inferensi (*inference engine*), yaitu kemampuan menarik kesimpulan berdasarkan pengalaman.



Gambar 2.1 Penerapan Konsep Kecerdasan Buatan

2.2.1 Kelebihan *Artificial Intelligence*

Artificial Intelligence mempunyai banyak kelebihan dibandingkan dengan sistem konvensional:

1. Tidak memerlukan persamaan matematik. obyek perancangan sistem konvensional selalu memerlukan persamaan matematik dari objek yang akan diproses, yang mana untuk sistem-sistem linear hal itu masih mudah untuk didapatkan, namun untuk sistem yang non linear maka sangat sulit sekali untuk mendapatkannya. Perancangan sistem kecerdasan buatan cukup memerlukan informasi penalaran dari perilaku proses, yang dapat

berupa informasi dalam bentuk bahasa (linguistic information).

2. Dapat melakukan proses pembelajaran. Memiliki kemampuan untuk mengambil kesimpulan berdasarkan data masukan-keluaran yang terdahulu.
3. Dapat bersifat adaptif. Memiliki kemampuan untuk mengubah parameter atau mengadaptasi parameter interna sistem secara mandiri.
4. Dapat kokoh terhadap perubahan parameter obyek. Kemampuan menantisipasi perubahan parameter obyek dengan mengadaptasi parameter internalnya.

2.3 Algoritma Sistem

2.3.1 *Ant Colony System*

Menurut (Fajrin & Meldra, 2019) ACO adalah teknik pencarian multi agent untuk menyelesaikan permasalahan kombinatorial dan persoalan yang lain yang di inspirasi tingkah laku semut dalam suatu koloni. Algoritma ini di inspirasi oleh tingkah laku koloni semut, bagaimana hewan yang hampir buta dengan kemampuan individu yang sederhana dapat menemukan jalan terpendek (sarang semut dengan sumber makanan) jika bersama dalam suatu koloni, dan *Ant Colony System* (ACS) adalah sebuah metodologi yang dihasilkan melalui pengamatan terhadap semut. Pada algoritma ACS, semut berfungsi sebagai agen yang ditugaskan untuk mencari solusi terhadap suatu masalah optimisasi. ACS telah diterapkan dalam berbagai bidang, salah satunya adalah untuk mencari solusi optimal pada Traveling Salesman Problem (TSP). Dengan memberikan sejumlah n titik, TSP dapat didefinisikan sebagai suatu permasalahan dalam menemukan jalur terpendek dengan mengunjungi setiap titik yang ada hanya sekali.

Ant Colony System (ACS) adalah sebuah metodologi yang dihasilkan melalui pengamatan terhadap semut. Pada algoritma ACS, semut berfungsi sebagai agen yang ditugaskan untuk mencari solusi terhadap suatu masalah optimisasi, Semut adalah serangga sosial yang hidupnya berkoloni, semut secara individu tidaklah begitu berguna. Hal ini telah diamati bahwa pada saat berjalan, semut telah menaruh sejumlah informasi, yang disebut *pheromone* (dalam jumlah tertentu), di tempat yang dilaluinya itu sehingga menandai jalur tersebut. Semut berikutnya yang melalui jalur tersebut dapat mengidentifikasi *pheromone* yang diletakkan oleh semut sebelumnya, memutuskan dengan probabilitas yang tinggi untuk mengikutinya, dan menguatkan jalur yang dipilihnya itu dengan *pheromone* miliknya, ACS memiliki tiga aspek utama [Dorigo dan Gambardella, 1997]: aturan transisi status pada sistem ini memberikan suatu cara langsung untuk menyeimbangkan antara penjelajahan (*exploration*) ruas-ruas yang baru dengan eksploitasi (*exploitation*) dari sebuah priori dan pengetahuan yang dihimpun mengenai masalah tersebut, aturan pembaruan *pheromone global* hanya dilakukan pada ruas-ruas yang merupakan bagian dari tur terbaik, dan disaat ants membangun sebuah solusi, diterapkan suatu aturan pembaruan *pheromone lokal* (*local pheromone updating rule*).

Menurut (Kristina et al., 2020) ACS merupakan metode yang digunakan untuk menyelesaikan masalah optimasi dimana inspirasi dalam memecahkan masalah tersebut berasal dari perilaku kumpulan atau kawan semut dalam mencari sumber makanan dan metode metaheuristic ini dapat diterapkan dengan mudah pada dunia nyata dengan waktu penyelesaian yang lebih cepat dibandingkan dengan metode eksak, Secara informal, ACS bekerja sebagai berikut: pertama kali,

sejumlah m semut ditempatkan pada sejumlah n titik berdasarkan beberapa aturan inisialisasi (misalnya, secara acak). Setiap semut membuat sebuah tur (yaitu, sebuah solusi TSP yang mungkin) dengan menerapkan sebuah aturan transisi status secara berulang kali. Selagi membangun turnya, seekor semut juga memodifikasi jumlah *pheromone* (sejumlah informasi yang ditinggalkan oleh semut di tempat yang dilalui dan menandai jalur tersebut) pada ruas-ruas yang dikunjunginya dengan menerapkan aturan pembaruan *pheromone* lokal. Setelah semua semut mengakhiri tur mereka, jumlah *pheromone* yang ada pada ruas-ruas dimodifikasi kembali (dengan menerapkan aturan pembaruan *pheromone global*). Dalam membuat tur, semut 'dipandu' oleh informasi heuristik (mereka lebih memilih ruas-ruas yang pendek) dan oleh informasi *pheromone*. Sebuah ruas dengan jumlah *pheromone* yang tinggi merupakan pilihan yang sangat diinginkan. Kedua aturan pembaruan *pheromone* itu dirancang agar semut cenderung untuk memberi lebih banyak *pheromone* pada ruas-ruas yang harus mereka lewati.

Terdapat tiga karakteristik utama dari ACS, yaitu : Aturan transisi status, Aturan pembaruan *pheromone* lokal, Aturan pembaruan *pheromone global*.

1. Aturan transisi status.

Aturan transisi status yang berlaku pada ACS adalah sebagai berikut: seekor semut yang ditempatkan pada titik t memilih untuk menuju ke titik v , kemudian diberikan bilangan pecahan acak q dimana $0 \leq q \leq 1$, q_0 adalah sebuah parameter yaitu Probabilitas semut melakukan eksplorasi pada setiap tahapan, dimana ($0 \leq q_0 \leq 1$) dan $p_k(t,v)$ adalah probabilitas dimana semut k memilih untuk bergerak dari titik t ke titik v .

Jika $q \leq q_0$ maka pemilihan titik yang akan dituju menerapkan aturan yang

ditunjukkan oleh persamaan (1)

$$\begin{aligned}
 \text{temporary}(t,u) &= [\tau(t,u_i)] \cdot [\eta(t,u_i)]^\beta, i = 1,2,3,\dots,n \\
 v &= \max \{ [\tau(t,u_i)] \cdot [\eta(t,u_i)]^\beta \} \dots\dots\dots(1)
 \end{aligned}$$

Dengan v = titik yang akan dituju

sedangkan jika q > q0 digunakan persamaan (2)

$$v = p_k(t,v) = \frac{[\tau(t,v)] \cdot [\eta(t,v)]^\beta}{\sum_{i=1}^n [\tau(t,u_i)] \cdot [\eta(t,u_i)]^\beta} \dots\dots\dots(2)$$

dengan

$$\eta(t,u_i) = \frac{1}{\text{jarak}(t,u_i)}$$

dimana $\tau(t,u)$ adalah nilai dari jejak *pheromone* pada titik (t,u), $\eta(t,u)$ adalah fungsi heuristik dimana dipilih sebagai invers jarak antara titik t dan u, β merupakan sebuah parameter yang mempertimbangkan kepentingan relatif dari informasi heuristic, yaitu besarnya bobot yang diberikan terhadap parameter informasi heuristic, sehingga solusi yang dihasilkan cenderung berdasarkan nilai fungsi matematis. Nilai untuk parameter β adalah ≥ 0 . Pheromon adalah zat kimia yang berasal dari kelenjar endokrin dan digunakan oleh makhluk hidup untuk mengenali sesama jenis, individu lain, kelompok, dan untuk membantu proses reproduksi. Berbeda dengan hormon, pheromon menyebar ke luar tubuh dapat mempengaruhi dan dikenali oleh individu lain yang sejenis (satu spesies). Proses peninggalan

pheromon ini dikenal sebagai *stigmergy*, sebuah proses memodifikasi lingkungan yang tidak hanya bertujuan untuk mengingat jalan pulang ke sarang, tetapi juga memungkinkan para semut berkomunikasi dengan koloninya. Seiring waktu, bagaimanapun juga jejak pheromon akan menguap dan akan mengurangi kekuatan daya tariknya, sehingga jejak pheromon harus diperbaharui. Pada ACS pembaruan pheromon dibagi menjadi 2, yaitu: Aturan pembaruan pheromon lokal, Aturan pembaruan *pheromon global*.

2. Aturan pembaruan *pheromon* lokal

Selagi melakukan tur untuk mencari solusi dari TSP, semut mengunjungi ruas-ruas dan mengubah tingkat pheromon pada ruas-ruas tersebut dengan menerapkan aturan pembaruan pheromon lokal yang ditunjukkan oleh persamaan (3)

$$\tau(t,v) \leftarrow (1-\rho) \cdot \tau(t,v) + \rho \cdot \Delta\tau(t,v) \quad (3)$$

$$\Delta\tau(t,v) = \frac{1}{L_{nn} \cdot c}$$

Dimana :

L_{nn} = panjang tur yang diperoleh

C = jumlah lokasi

ρ = parameter dengan nilai 0 sampai 1

$\Delta\tau$ = perubahan pheromon

ρ adalah sebuah parameter (koefisien evaporasi), yaitu besarnya koefisien penguapan pheromon . Adanya penguapan *pheromone* menyebabkan tidak semua semut mengikuti jalur yang sama dengan semut sebelumnya. Hal ini

memungkinkan dihasilkan solusi alternatif yang lebih banyak. Peranan dari aturan pembaruan *pheromone* lokal ini adalah untuk mengacak arah lintasan yang sedang dibangun, sehingga titik-titik yang telah dilewati sebelumnya oleh tur seekor semut mungkin akan dilewati kemudian oleh tur semut yang lain. Dengan kata lain, pengaruh dari pembaruan lokal ini adalah untuk membuat tingkat ketertarikan ruas-ruas yang ada berubah secara dinamis: setiap kali seekor semut menggunakan sebuah ruas maka ruas ini dengan segera akan berkurang tingkat ketertarikannya (karena ruas tersebut kehilangan sejumlah *pheromon*-nya), secara tidak langsung semut yang lain akan memilih ruas-ruas lain yang belum dikunjungi. Konsekuensinya, semut tidak akan memiliki kecenderungan untuk berkumpul pada jalur yang sama. Fakta ini, yang telah diamati dengan melakukan percobaan [Dorigo dan Gambardella, 1997]. Merupakan sifat yang diharapkan bahwa jika semut membuat tur-tur yang berbeda maka akan terdapat kemungkinan yang lebih tinggi dimana salah satu dari mereka akan menemukan solusi yang lebih baik daripada mereka semua berkumpul dalam tur yang sama. Dengan cara ini, semut akan membuat penggunaan informasi *pheromon* menjadi lebih baik tanpa pembaruan lokal, semua semut akan mencari pada lingkungan yang sempit dari tur terbaik yang telah ditemukan sebelumnya.

3. Aturan pembaruan *pheromon global*

Pada sistem ini, pembaruan *pheromon* secara global hanya dilakukan oleh semut yang membuat tur terpendek sejak permulaan percobaan. Pada akhir sebuah iterasi, setelah semua semut menyelesaikan tur mereka, sejumlah *pheromon* ditaruh pada ruas-ruas yang dilewati oleh seekor semut yang telah

menemukan tur terbaik (ruas-ruas yang lain tidak diubah). Tingkat pheromon itu diperbarui dengan menerapkan aturan pembaruan *pheromon global* yang ditunjukkan oleh persamaan (4).

$$\tau(t,v) \leftarrow (1-\alpha) \cdot \tau(t,v) + \alpha \cdot \Delta\tau(t,v) \dots\dots\dots(4)$$

$$\Delta\tau(t,v) = \begin{cases} L_{gb}^{-1} & \text{jika } (t,v) \in \text{tur_terbaik} \\ 0 & \end{cases}$$

Dimana :

$\tau(t,v)$ = nilai *pheromone* akhir setelah mengalami pembaharuan lokal

L_{gb} = panjang jalur terpendek pada akhir siklus

α = parameter dengan nilai antara 0 sampai 1

$\Delta\tau$ = perubahan *pheromone*

$\Delta\tau(t,v)$ bernilai L_{gb}^{-1} jika ruas (t,v) merupakan bagian dari rute terbaik namun jika sebaliknya $\Delta\tau(t,v) = 0$. α adalah tingkat kepentingan relatif dari

pheromon atau besarnya bobot yang diberikan terhadap pheromon, sehingga solusi yang dihasilkan cenderung mengikuti sejarah masa lalu dari semut dari perjalanan sebelumnya, dimana nilai parameter α adalah ≥ 0 , dan L_{gb} adalah panjang dari tur terbaik secara global sejak permulaan percobaan.

Pembaruan *pheromon global* dimaksudkan untuk memberikan pheromon yang lebih banyak pada tur-tur yang lebih pendek. Persamaan (3) menjelaskan bahwa hanya ruas-ruas yang merupakan bagian dari tur terbaik secara global yang akan menerima penambahan *pheromone*, Kelebihan dari metode ACS dibandingkan metode ACO yang lain adalah pada tour construction dimana ACS memilih jalur dengan pseudorandom proportional

rule, sedangkan metode ACO lainnya berdasarkan random proportional rule (A. Leksono, 2009). Berikut adalah tahapan dari ACS:

- Tour Construction

Ketika pada node i , semut k akan memilih node j dengan formulasi:

$$j = \begin{cases} \operatorname{argmax}_{l \in N(s^p)} [\tau_{il} (\eta_{il})^\beta], & \text{if } q \leq q_0 \\ J, & \text{otherwise} \end{cases}$$

Di mana, J merupakan persamaan (30) dengan nilai $\alpha = 1$. Dengan notasi:

- i, j : indeks lokasi
- l : node yang belum dikunjungi oleh semut k
- β : parameter visibilitas jarak antar node
- τ_{il} : intensitas *pheromone* pada node i, l
- $N(s^p)$: kumpulan dari feasible component
- η_{il} : invers dari jarak antar node i, l
- Q : bilangan random yang terdistribusi uniform $[0, 1]$
- Q_0 : parameter ($0 \leq q_0 \leq 1$)

- Global Pheromone Update

Pembaharuan *pheromone* hanya dapat dilakukan oleh semut terbaik dengan formulasi:

$$\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij} + \rho \cdot \Delta\tau_{ij}^{best}$$

Di mana,

$$\Delta\tau_{ij}^{best} = \begin{cases} 1/TC_{best}, & \text{if } (i, j) \text{ belongs to best tour} \\ 0, & \text{otherwise} \end{cases}$$

Dengan notasi:

- i, j : indeks lokasi
- ρ : intensitas penguapan *pheromone*
- τ_{ij} : intensitas *pheromone* pada node i, j
- ρ : intensitas penguapan *pheromone*
- $\Delta\tau_{ij}^{best}$: jumlah *pheromone* pada node i, j oleh semut terbaik
- TC_{best} : Total cost yang dibuat oleh semut terbaik

- *Local Pheromone Update*

Semut menggunakan aturan *local pheromone update* melewati (i,i) dengan formulasi:

$$\tau_{ij} \leftarrow (1 - \rho) \cdot \tau_{ij} + \rho \cdot \tau_0$$

Di mana,

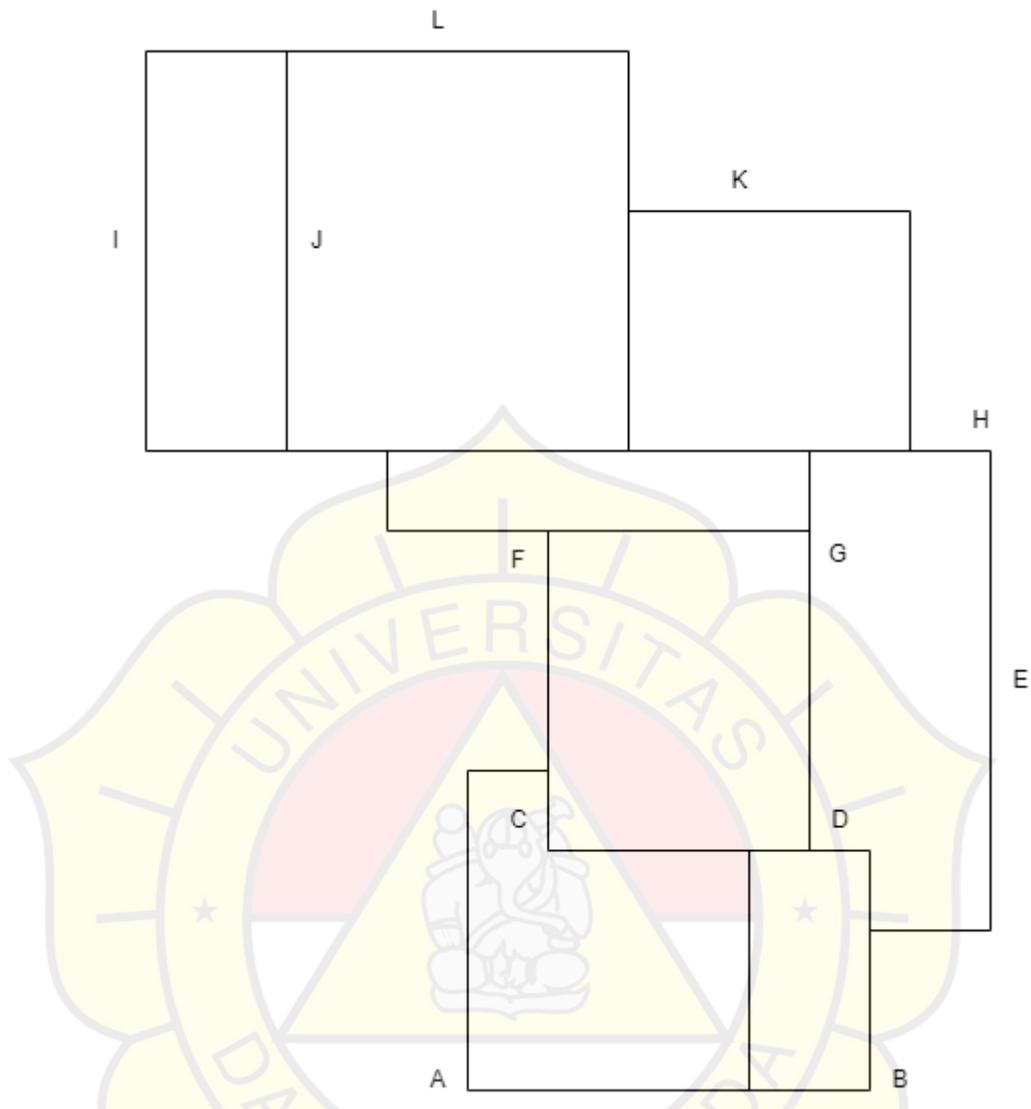
$$\tau_0 = \frac{1}{n \cdot TC_{initial}}$$

Dengan notasi:

i, j	: indeks lokasi
ρ	: intensitas penguapan <i>pheromone</i>
n	: total dari jumlah <i>customer</i>
τ_0	: inisialisasi <i>pheromone</i>
$TC_{initial}$: total initial cost

2.3.2 Model Denah Lokasi Parkir

Pada saat ini kendaraan yang akan mencari parkir harus berjalan berputar putar untuk menemukan posisi parkir yang terbaik, dapat di lihat pada peta parkir arah rute tempuh pada peta sangat tidak beraturan, dalam hal ini berarti kendaraan menghabiskan waktu tempuh yang lebih banyak untuk menemukan lokasi parkir karena tidak adanya ukuran untuk menentukan posisi parkir yang kosong, Berikut adalah model denah yang ada :



Gambar 2.2 Denah Parkir Kendaraan Roda Empat

2.4 Pemodelan UML

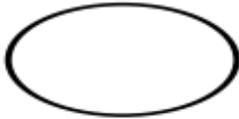
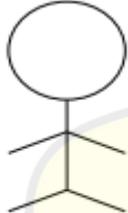
“*Unified Modeling Language (UML)* bukanlah suatu proses melainkan bahasa pemodelan secara grafis untuk menspesifikasikan, memvisualisasikan, membangun, dan mendokumentasikan seluruh artifak sistem perangkat lunak. Penggunaan model ini bertujuan untuk mengidentifikasi bagian-bagian yang termasuk dalam lingkup sistem dibahas dan bagaimana hubungan antara sistem suatu dengan subsistem maupun sistem lain di luarnya.” *Unified Modeling*

Language (UML)” adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasi, menspesifikasikan dari sebuah sistem pengembangan software berbasis object oriented.” Dari Pengertian diatas penulis menyimpulkan bahwa *Unified Modeling Language (UML)* merupakan bahasa pemodelan yang berbentuk grafis yang digunakan untuk memvisualisasi, menspesifikasikan suatu sistem perangkat lunak (Yuhanar, 2018).

2.4.1 Use case Diagram

Diagram *use case* merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam suatu sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.”(Yunahar, 2018). Berikut adalah simbol-simbol yang ada pada diagram *use case*:

Tabel 2.1 Simbol *Use case Diagram* (Yunahar, 2018)

Simbol	Deskripsi
<p><i>Use Case</i></p> 	Fungsional yang disediakan sistem sebagai unit-unit yang saling bertukar antar unit atau aktor biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <u>use case</u>
<p><i>Actor</i></p> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, biasanya dinyatakan menggunakan kata benda di awal <i>frase</i> nama aktor
<p>Asosiasi</p> 	Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor
<p><i>Extensi</i></p> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan
<p>Generalisasi</p> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya
<p>Include</p> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini

2.4.2 Activity Diagram

Activity Diagram menggambarkan *work flow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas dapat dilakukan oleh sistem” (Yunahar, 2018). Simbol-simbol yang digunakan dalam *Activity Diagram* sebagai berikut :

Tabel 2.2 Simbol *Activity Diagram* (Yunahar, 2018)

Simbol	Deskripsi
	Simbol <i>start</i> untuk menyatakan awal dari suatu proses
	Simbol <i>stop</i> untuk menyatakan akhir dari suatu proses
	Simbol <i>decision</i> digunakan untuk menyatakan kondisi dari suatu proses
	Simbol <i>action</i> menyatakan aksi yang dilakukan dalam suatu arsitektur sistem

2.4.3 *Sequence Diagram*

“*Sequence Diagram* adalah *tool* yang sangat populer dalam pengembangan sistem informasi secara *object-oriented* untuk menampilkan interaksi antar objek. Berdasarkan definisi tersebut, dapat disimpulkan bahwa *Sequence Diagram* adalah *tool* yang digunakan dalam pengembangan system” (Yunahar, 2018).

Tabel 2.3 Simbol *Sequence Diagram* (Yunahar, 2018)

Nama Komponen	Keterangan	Simbol
<i>Lifeline</i>	Mengindikasikan keberadaan sebuah objek dalam basis waktu. Notasi untuk <i>lifeline</i> adalah garis putus-putus vertikal yang ditarik dari sebuah objek	
<i>Activation</i>	Dinotasikan sebagai sebuah kotak segi empat digambar pada sebuah lifeline mengindikasikan sebuah objek yang akan melakukan sebuah aksi	
<i>Message</i>	Digambarkan dengan anak panah horizontal antara activation. Message mengindikasikan komunikasi antara objek-objek	
<i>Object</i>	Merupakan <i>instance</i> dari sebuah class dan dituliskan tersusun secara horizontal.	
<i>Actor</i>	<i>Actor</i> juga dapat berkomunikasi	