

## **BAB II**

### **TINJAUAN PUSTAKA DAN LANDASAN TEORI**

#### **2.1 Tinjauan Pustaka**

Dalam penelitian yang berjudul **“Retail Store Location Selection Problem with Multiple Analytical Hierarchy Process of Decision Making on Application in Turkey”** peneliti mampu memodelkan *Multiple Analytical Hierarchy Process(AHP)* dengan 5 kriteria utama yaitu biaya, kondisi persaingan, keadaan lalu lintas, fitur fisik, dan lokasi ritel untuk membandingkan tiga lokasi ritel di Turki. Berdasarkan derajat kepentingan terhadap kriteria utama, dan relevansi terhadap sub-kriteria, maka didapatkan hasil bahwa dari lokasi yang ada yaitu Bosnak Hersek, Turmak Lisesi, dan Toki Konutlari yang paling unggul adalah Bosnak Hersek dengan rasio 56%, diikuti Turmak Lisesi dengan Rasio 25% lalu yang terakhir Toki Konutlari dengan rasio 19% (Hikmek Erbiyik dkk, 2012).

Dari penelitian yang berjudul **“Selecting Location of Retail Stores Using Artificial Neural Network and Google Places API”**. Dalam penelitian tersebut penulis menggunakan metode *Artificial Neural Network* yang dikombinasikan dengan *Google Places API* untuk memproses 123 data toko ritel sebagai data learning, dan 18 data sebagai data testing untuk membuat keputusan pembukaan 3 toko baru. Sistem disimulasikan dengan data yang telah dikumpulkan dari lokasi acak. Hasil yang ditinjau ulang oleh para ahli menunjukkan bahwa sistem menghasilkan estimasi ranking yang masuk akal terhadap kandidat yang ada. Hasil dari sistem juga menunjukkan adanya persamaan antara data kandidat toko dan data toko yang sudah ada. Algoritma yang sama lalu diaplikasikan pada perusahaan, dan dapat menghasilkan ranking pada lokasi yang diproses sesuai yang diperhitungkan oleh sistem (Satman, 2014).

Dalam penelitian yang berjudul **“Rancang Bangun Aplikasi Location Based Service Wisata Populer di Jawa Timur Berbasis Android Dengan Metode Extreme Programming”** peneliti membangun aplikasi *location base services* berbasis android dengan *Google Maps API* dan metode *extreme programming* untuk menampilkan lokasi wisata populer di Jawa Timur. Dari hasil

pengujian sistem *Alpha* maupun *Beta* dengan mengikuti alur metode *extreme programming* didapatkan bahwa aplikasi yang dibangun berhasil mengimplementasikan metode *extreme programming* dan *location based services* berbasis android untuk menampilkan wisata populer di Jawa Timur (Nasrulloh, 2017).

Dalam penelitian yang berjudul **“Pembangunan Aplikasi Android Rekomendasi Tempat Rental Motor Di Kota Malang Dengan Metode AHP TOPSIS Berbasis Location Based Services”** peneliti mampu menerapkan metode *AHP TOPSIS* pada aplikasi *Location Based Services* berbasis android untuk menentukan tempat rental motor terbaik sesuai bobot kriteria yang diinputkan pengguna. Kriteria ini mencakup harga sewa motor 125cc per hari, lokasi tempat rental motor, popularitas dan rating tempat rental motor. Hasil dari pengujian penelitian ini mengatakan bahwa nilai usability mencapai nilai 82.82%, dimana nilai ini termasuk dalam kualifikasi baik, kemudian hasil pengujian fungsional bernilai 100%, lalu pengujian kesesuaian juga menyatakan bahwa aplikasi ini memiliki nilai hasil pengujian 90% (Jeriko Hosea Julanto, 2018).

Dalam penelitian yang berjudul **“Sistem Pendukung Keputusan Rekomendasi Restaurant Location Based pada Perangkat Mobile dengan Menggunakan Metode Weighted Product(WP)”** peneliti mampu membangun sebuah aplikasi *location based service* dengan *Unity3D* untuk perangkat Android dengan metode *Weighted Product(WP)*. Dari perhitungan nilai preferensi dengan kriteria selisih jarak, rating, jumlah kategori, jumlah kriteria, harga rata-rata yang telah diberi bobot masing-masing, aplikasi dapat menampilkan rekomendasi restoran yang ada di Surabaya dan Malang berdasarkan rangking. Perangkingan dilakukan untuk menentukan pilihan terbaik yang akan direkomendasikan kepada user terhadap alternatif yang memiliki nilai preferensi relatif yang paling tinggi (Nazzun Hanif Ahsani, 2015).

## **2.2 Landasan Teori**

### **2.2.1 Sistem Pendukung Keputusan**

#### **2.2.1.1 Definisi Sistem Pendukung Keputusan**

Sistem Pendukung Keputusan adalah barmasi berbasis komputer termasuk sistem berbasis pengetahuan (manajemen pengetahuan) yang dipakai untuk mendukung pengambilan keputusan dalam suatu organisasi atau perusahaan. Dan juga dikatan sebagai sistem komputer yang mengolah data menjadi informasi untuk pengambilan keputusan dari masalah semi-terstruktur yang spesifik.

SPK dapat digambarkan dengan sistem yang berkemampuan mendukung analisis *ad hoc* data, dan pemodelan keputusan, berorientai keputusan, orientasi perencanaan masa depan, dan digunakan pada saat-saat yang tidak biasa (Moore, 1980).

Sistem Pendukung Keputusan sebagai sistem berbasis komputer yang terdiri dari tiga komponen yang saling berinteraksi, yaitu (1) Sistem Bahasa merupakan mekanisme untuk memberikan komunikasi antara pengguna dan komponen sistem pendukung keputusan lainnya, (2) Sistem Pengetahuan merupakan repositori pengetahuan domain masalah yang ada pada Sistem Pendukung Keputusan baik sebagai data maupun sebagai prosedur, dan (3) Sistem Pemrosesan Masalah merupakan hubungan antara dua komponen lainnya, terdiri dari satu atau lebih kapabilitas manipulasi masalah umum yang diperlukan untuk mengambil keputusan. Konsep-konsep yang diberikan oleh definisi tersebut sangat penting untuk memahami hubungan antara SPK dan pengetahuan (Manurung, 2010).

#### **2.2.1.2 Tahapan Pengambilan Keputusan**

Dalam keputusan ada beberapa keadaan yang mungkin dialami oleh pengambilan keputusan ketika mengambil keputusan, yaitu:

1. Pengambilan keputusan dalam kepastian, semua alternatif diketahui secara pasti.
2. Pengambilan keputusan dalam berbagai tingkat resiko yang dipilih.
3. Pengambilan keputusan dalam kondisi ketidakpastian, ada alternatif yang tidak diketahui dengan jelas.

Tentu saja, pengambilan keputusan akan menjadi mudah jika dilakukan dengan suatu kepastian. Tahap-tahap pengambilan keputusan yaitu :

1. Identifikasi masalah
2. Pemilihan metode
3. Pengumpulan data yang dibutuhkan untuk melaksanakan model keputusan.

Adapun teknik pengumpulan data sebagai berikut:

- a. Kuesioner.
  - b. Observasi.
  - c. Wawancara.
  - d. Dokumen.
4. Mengimplementasikan model.
  5. Mengevaluasi sisi positif dari setiap alternatif yang ada.
  6. Melaksanakan solusi terpilih.

### **2.2.2 Location Based Service (LBS)**

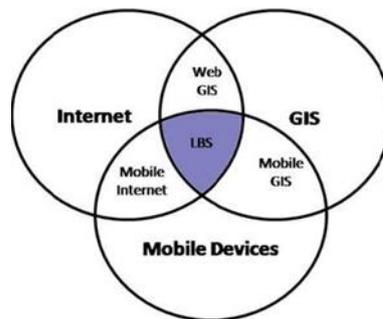
Layanan Berbasis Lokasi atau lebih dikenal dengan *Location Based Service* (LBS) istilah umum yang digunakan untuk menggambarkan teknologi yang digunakan untuk menemukan lokasi perangkat yang kita gunakan. LBS adalah layanan informasi yang dapat diakses melalui *mobile device* dengan menggunakan *mobile network*, yang dilengkapi kemampuan untuk memanfaatkan lokasi dari *mobile device* tersebut . Terdapat dua unsur utama pada LBS (Safaat H, 2013) yaitu:

a) *Location Manager (API Maps)*

Menyediakan *tools/source* untuk LBS, *Application Programming Interface (API) Maps* menyediakan fasilitas untuk menampilkan, memanipulasi maps/peta beserta fitur-fitur lainnya seperti tampilan satelit, *street* (jalan), maupun gabungannya. Paket ini berada pada [com.google.adroid.maps](http://com.google.adroid.maps).

b) *Location Provider (API Location)*

Menyediakan teknologi pencarian lokasi yang digunakan oleh device/perangkat. *API Location* berhubungan dengan data GPS (*Global Positioning System*) dan data lokasi real-time. *API Location* berada pada paket android yaitu dalam paket `android.location`. Dengan *Location Manager*, kita dapat menentukan lokasi kita saat ini, track gerakan/perpindahan, serta kedekatan dengan lokasi tertentu dengan mendeteksi perpindahan.



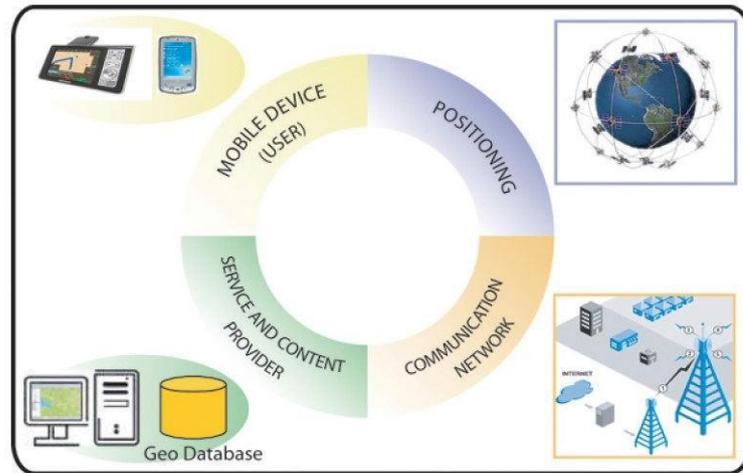
Gambar 2. 1 Teknologi Location based Services

*Location Based Service* dapat digambarkan sebagai suatu layanan yang berada pada pertemuan tiga teknologi yaitu: *Geographic Information System*, *Internet Service*, dan *Mobile Devices*. Teknologi *Location Based Services* berfokus bagaimana menentukan posisi dari peralatan yang Anda gunakan atau disebut dengan metode *positioning*. (Safaat H, 2013)

### 2.2.2.1 Komponen Location Based Services (LBS)

Dalam menggunakan LBS elemen yang diperlukan antara lain :

- Mobile Device* yaitu sebuah alat yang digunakan untuk meminta informasi yang dibutuhkan.
- Communication Network* adalah jaringan selular yang mengirimkan data pengguna dan permintaan layanan.
- Positioning Component* untuk pengolahan layanan biasanya posisi pengguna harus ditentukan
- Service and Application Provider* adalah penyedia layanan pengguna selular yang bertanggung jawab untuk memproses layanan.
- Data and Content Provider* yaitu penyedia layanan informasi data yang dapat diminta oleh pengguna.



Gambar 2. 2 Komponen Location Based Services

### 2.2.3 Google Maps API

*Google Maps API* adalah suatu *library* yang berbentuk JavaScript. Cara membuat *Google Maps* untuk ditampilkan pada suatu web atau blog sangat mudah hanya dengan membutuhkan pengetahuan mengenai HTML serta JavaScript, serta koneksi Internet yang sangat stabil. Dengan menggunakan *Google Maps API*, kita dapat menghemat waktu dan biaya untuk membangun aplikasi peta digital yang handal, sehingga kita dapat fokus hanya pada data-data yang akan ditampilkan. Dengan kata lain, kita hanya membuat suatu data sedangkan peta yang akan ditampilkan adalah milik Google sehingga kita tidak dipusingkan dengan membuat peta suatu lokasi, bahkan dunia. (Swastikayana, 2011)

Dalam pembuatan program *Google Maps API* menggunakan urutan sebagai berikut:

1. Memasukkan *Maps API JavaScript* ke dalam HTML.
2. Membuat element div dengan nama *map\_canvas* untuk menampilkan peta.
3. Membuat beberapa objek literal untuk menyimpan properti-properti pada peta.
4. Menuliskan fungsi *JavaScript* untuk membuat objek peta.
5. Meng-inisiasi peta dalam tag body HTML dengan event onload.

Kode program dasar:

```

<html>
<head>
<meta name="viewport" content="initial-scale=1.0, user-
scalable=no" />
<!-- Langkah 1 -->
<script type="text/javascript"
src="http://maps.google.com/maps/api/js?sensor=true&key=ABQIA
AAA8tt4eKT
uBZMVnLJfP2BZrBT2yXp_ZAY8_ufC3CFXhHIE1NvwkxS4Rz1LFzG0odNPtk8VLkdr
QF5grA"></ script>
<script type="text/javascript">
// Langkah 4
function initialize() {
var latlng = new google.maps.LatLng(-6.4, 106.8186111); var
myOptions = {
zoom: 13, center: latlng,
mapTypeId: google.maps.MapTypeId.ROADMAP
};
// Langkah 3
var map = new
google.maps.Map(document.getElementById("map_canvas"),
myOptions);
}
</script>
</head>

```

### Model Program 2.1 Contoh Program Dasar *Google Maps API*

Pada *Google Maps API* terdapat 4 jenis pilihan model peta yang disediakan oleh Google, diantaranya adalah:

1. *ROADMAP*, ini yang saya pilih, untuk menampilkan peta biasa 2 dimensi
2. *SATELLITE*, untuk menampilkan foto satelit
3. *TERRAIN*, untuk menunjukkan relief fisik permukaan bumi dan menunjukkan seberapa tingginya suatu lokasi, contohnya akan menunjukkan gunung dan sungai
4. *HYBRID*, akan menunjukkan foto satelit yang di atasnya tergambar pula apa yang tampil pada *ROADMAP* (jalan dan nama kota)

#### 2.2.4 *Simple additive weighting (SAW)*

Metode SAW sering juga dikenal istilah metode penjumlahan terbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut. Langkah-langkah metode dalam metode SAW adalah (H.S. Wibowo, 2008) :

1. Membuat matriks keputusan  $Z$  berukuran  $m \times n$ , dimana  $m =$

alternatif yang akan dipilih dan  $n =$  kriteria.

2. Memberikan nilai  $x$  setiap alternatif ( $i$ ) pada setiap kriteria ( $j$ ) yang sudah ditentukan, dimana,  $i=1,2,\dots,m$  dan  $j=1,2,\dots,n$  pada matriks keputusan  $Z$ .

$$Z = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1j} \\ \vdots & & & \vdots \\ x_{i1} & x_{i2} & \dots & x_{ij} \end{bmatrix} \quad (1)$$

3. Memberikan nilai bobot preferensi ( $W$ ) oleh pengambil keputusan untuk masing-masing kriteria yang sudah ditentukan.

$$W = [W_1 \quad W_2 \quad W_3 \quad \dots \quad W_j] \quad (2)$$

4. Melakukan normalisasi matriks keputusan  $Z$  dengan cara menghitung nilai rating kinerja ternormalisasi ( $r_{ij}$ ) dari alternatif  $A_i$  pada atribut  $C_j$ .

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\text{MAX}_i(x_{ij})} & \text{Jika } j \text{ adalah atribut keuntungan} \\ \frac{\text{MIN}_i(x_{ij})}{x_{ij}} & \text{Jika } j \text{ adalah atribut biaya} \end{cases} \quad (3)$$

Dengan ketentuan :

- a. Dikatakan atribut keuntungan apabila atribut banyak memberikan keuntungan bagi pengambil keputusan, sedangkan atribut biaya merupakan atribut yang banyak memberikan pengeluaran jika nilainya semakin besar bagi pengambil keputusan.
  - b. Apabila berupa atribut keuntungan maka nilai ( $x_{ij}$ ) dari setiap kolom atribut dibagi dengan nilai ( $\text{MAX } x_{ij}$ ) dari tiap kolom, sedangkan untuk atribut biaya, nilai ( $\text{MIN } x_{ij}$ ) dari tiap kolom atribut dibagi dengan nilai ( $x_{ij}$ ) setiap kolom.
5. Hasil dari nilai rating kinerja ternormalisasi ( $r_{ij}$ ) membentuk matriks ternormalisasi ( $N$ )

$$N = \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1j} \\ \vdots & & & \vdots \\ r_{i1} & r_{i2} & \dots & r_{ij} \end{bmatrix}$$

6. Melakukan proses perankingan dengan cara mengalikan matriks ternormalisasi (N) dengan nilai bobot preferensi (W).
7. Menentukan nilai preferensi untuk setiap alternatif ( $V_i$ ) dengan cara menjumlahkan hasil kali antara matriks ternormalisasi (N) dengan nilai bobot preferensi(W).

$$V_i = \sum_{j=1}^n W_j r_{ij}$$

Nilai  $V_i$  yang lebih besar mengindikasikan bahwa alternatif  $A_i$  merupakan alternatif terbaik