

## **BAB II**

### **LANDASAN TEORI**

Bab landasan teori membahas mengenai dasar teori-teori pendukung dalam penelitian dan membahas tinjauan pustaka penelitian sebelumnya.

#### **2.1 Dasar teori**

##### **2.1.1 Sistem Pendukung Keputusan**

Sistem Pendukung Keputusan (SPK) adalah sebuah sistem yang dapat mendukung pengambil keputusan dalam situasi semi terstruktur yang dimaksudkan sebagai tambahan penilaian atau keputusan yang tidak dapat didukung oleh algoritma, sistem tersebut untuk mendukung solusi dari masalah tertentu atau untuk mengevaluasi sebuah peluang (Turban et al., 2005). Sedangkan menurut setiawan (Angeline & Astuti, 2018) Sistem pendukung keputusan merupakan penggabungan sumber-sumber kecerdasan individu dengan kemampuan komponen untuk memperbaiki kualitas keputusan. Menurut Subakti (2002) menyatakan bahwa *Decision Support System* (DSS) adalah sistem berbasis komputer yang terdiri 3 komponen interaktif:

1. sistem bahasa – mekanisme yang menyediakan komunikasi diantara *user* dan berbagai komponen dalam DSS
2. *knowledge system* – penyimpanan *knowledge domain* permasalahan yang ditanamkan dalam DSS, baik sebagai data ataupun prosedur
3. sistem pemrosesan permasalahan – *link* diantara dua komponen, mengandung satu atau lebih kemampuan memanipulasi masalah yang dibutuhkan untuk pengambilan keputusan.

##### **2.1.2 Badan Kepegawaian dan Pengembangan Sumber Daya Manusia (BKPSDM)**

Badan Kepegawaian dan Pengembangan Sumber Daya Manusia (BKPSDM) adalah sebuah badan pengelola urusan Pegawai Negeri Sipil (PNS) sesuai dengan daerah otonom. Penelitian ini berfokus pada instansi BKPSDM di

Kabupaten Kayong Utara. BKPSDM Kayong Utara dibentuk untuk mengelola urusan kepegawaian termasuk kenaikan jabatan struktural bagi PNS yang ada di wilayah Kayong Utara. Secara umum, BKPSDM memiliki tugas pokok yaitu membuat dan melaksanakan penyusunan serta pelaksanaan kebijakan daerah yang bersifat spesifik di bidang pelayanan kepegawaian, pengembangan pegawai, mutasi, pendidikan dan pelatihan serta melaksanakan ketatausahaan badan.

### **2.1.3 Jabatan Struktural Pegawai Negeri Sipil**

Jabatan struktural merupakan jabatan yang secara tegas terdapat dalam struktur organisasi. Jabatan struktural hanya dapat diduduki oleh pegawai yang sudah berstatus pegawai negeri sipil atau PNS. Kedudukan dalam jabatan struktural memiliki tingkatan dari tingkat yang terendah yaitu eselon IV/b hingga yang tertinggi yaitu eselon 1/a. Contoh jabatan struktural PNS di daerah yaitu Kepala Dinas/Badan/Kantor, Kepala Bagian, Kepala Bidang, Sekretaris Daerah, Kepala Seksi, Camat, Sekretaris Camat, Lurah, dan Sekretaris Lurah. Sedangkan contoh jabatan struktural PNS di pusat yaitu Direktur Jendral, Sekretaris Jendral, Kepala Biro dan Staf Ahli (Fitri, 2010).

### **2.1.4 Profile Matching**

Dwijaya (2010) mendefinisikan *profile matching* secara garis besar merupakan proses membandingkan antara kompetensi individu kedalam kompetensi jabatan sehingga dapat diketahui perbedaan kompetensinya (disebut juga gap), semakin kecil gap yang dihasilkan maka bobot nilainya semakin besar yang berarti memiliki peluang lebih besar untuk karyawan menempati posisi tersebut. Metode *profile matching* merupakan sebuah mekanisme dalam proses pengambilan keputusan dengan mengansumsikan bahwa terdapat tingkat *variable* prediktor ideal yang harus dimiliki oleh individu, bukan tingkat minimal yang harus dipenuhi atau dilewati. Berikut prosedur metode *profile matching* (Atmanegara dkk, 2017):

### 1. Pemetaan Gap

Gap adalah perbedaan antara profil jabatan dengan nilai profil individu. Perhitungan Gap terdapat pada persamaan 2.1 sebagai berikut.

$$Gap = Nilai - Nilai Standar \quad (2.1)$$

Contoh rentang Gap dapat dilihat pada Tabel 2.1. *Range* penilaian dapat disesuaikan.

**Tabel 2.1** Nilai Gap

Range Penilaian	Kategori	Nilai
0-49	Sangat Kurang	1
50-59	Kurang	2
60-74	Cukup	3
75-84	Baik	4
84-100	Sangat Baik	5

### 2. Penentuan Bobot Nilai Gap

Pada tahap ini akan ditentukan bobot nilai masing-masing atribut. Berikut perbandingan bobot dengan nilai gap dapat dilihat pada tabel 2.2.

**Tabel 2.2** Perbandingan Bobot dengan Nilai Gap

Selisih	Bobot	Keterangan
0	5	Tidak ada selisih (Kompetensi sesuai dengan yang dibutuhkan)
1	4,5	Kompetensi kelebihan 1 tingkat
-1	4	Kompetensi kekurangan 1 tingkat
2	3,5	Kompetensi kelebihan 2 tingkat
-2	3	Kompetensi kekurangan 2 tingkat
3	2,5	Kompetensi kelebihan 3 tingkat
-3	2	Kompetensi kekurangan 3 tingkat
4	1,5	Kompetensi kelebihan 4 tingkat
-4	1	Kompetensi kekurangan 4 tingkat

### 3. Perhitungan Nilai *Core Factor* dan *Secondary Factor*

*Core factor* merupakan faktor yang paling menonjol atau yang paling dibutuhkan dari suatu jabatan yang diperkirakan akan menghasilkan kinerja yang optimal. Sehingga nilai persenan pada *core factor* pada penelitian ini ditentukan sebesar 60%. Sedangkan *secondary factor* merupakan faktor

pendukung dari suatu jabatan yang memiliki nilai persenan yaitu 40% (Sutoyo, 2015). Berikut rumus dari *core factor* dan *secondary factor*.

a. *Core Factor*

Perhitungan *core factor* dapat dirumuskan pada persamaan 2.2

$$NCF = \frac{\sum NC}{\sum IC} \quad (2.2)$$

Keterangan:

NCF = Nilai Rata-Rata *Core Factor*

NC = Jumlah Total Nilai *Core Factor* (atribut 1, atribut 2, dst)

IC = Jumlah aspek *Core Factor*

b. *Secondary Factor*

Perhitungan *secondary factor* dapat dilihat pada persamaan 2.3.

$$NSF = \frac{\sum NS}{\sum IC} \quad (2.3)$$

Keterangan:

NSF = Nilai Rata-Rata *Secondary Factor*

NS = Jumlah Total Nilai *Secondary Factor* (atribut 1, atribut 2, dst)

IC = Jumlah aspek *Secondary Factor*

4. Perhitungan Nilai Total Aspek

Perhitungan nilai total aspek dapat dirumuskan pada persamaan 2.4.

$$\text{Nilai Total} = (x)\% NCF + (x)\% NSF \quad (2.4)$$

Keterangan:

NCF = Nilai Rata-Rata *Core Factor*

NSF = Nilai Rata-Rata *Secondary Factor*

(x)% = Nilai Persen Yang Diinputkan

## 5. Perhitungan Hasil Akhir

Hasil akhir dari metode *profile matching* adalah perurutan kandidat yang akan menempati posisi jabatan tertentu. Penentuan ranking mengacu pada hasil perhitungan, perhitungan tersebut dapat ditunjukkan pada persamaan 2.5.

$$\text{Hasil Akhir} = \sum_{i=0} (x)\%Ni \quad (2.5)$$

Keterangan:

Ni = Nilai Total Aspek *i*

(*x*)% = Nilai Persen Rumus Hasil Akhir (total 100%)

### 2.1.5 Analisis PIECES

Analisis PIECES merupakan akronim dari *Performance, Information, Economy, Controll, Efeciency, Service* dimana menurut fatta (2007) digunakan unuk mengidentifikasi masalah yang meliputi analisis terhadap kinerja, informasi, ekonomi, keamanan aplikasi, efesiensi dan pelayanan pelanggan. Panduan ini dikenal dengan istilah analisis PIECES. Adapun penjelasan secara rinci mengenai analisis PIECES adalah sebagai berikut:

#### 1. *Performance* (Analisis Kinerja)

Analisis kinerja diukur berdasarkan jumlah produksi dan waktu tanggap. Jumlah produksi adalah jumlah pekerjaan yang bisa diselesaikan selama jangka waktu tertentu. Sedangkan waktu tanggap adalah keterlambatan rata-rata antara suatu transaksi dengan tanggapan yang diberikan kepada transaksi tersebut.

#### 2. *Information* (Analisis Informasi)

Analisis informasi merupakan evaluasi terhadap kemampuan sistem informasi dalam menghasilkan informasi yang bermanfaat. Dalam hal ini meningkatkan jumlah informasi tidak dengan menambah jumlah informasi, karena terlalu banyak informasi akan menimbulkan masalah baru. Situasi yang membutuhkan peningkatan informasi meliputi:

- a. Kurangnya informasi mengenai keputusan atau situasi yang sekarang.
- b. Kurangnya informasi yang relevan mengenai situasi sekarang.
- c. Kurangnya informasi yang tepat waktu.

- d. Terlalu banyak informasi.
- e. Informasi tidak akurat.

Sementara analisis informasi memeriksa keluaran dari sistem, analisis data, meneliti data yang disimpan dalam sebuah sistem. Permasalahan yang dihadapi meliputi:

- a. Data yang berlebihan, data yang sama ditangkap dan disimpan di dalam banyak tempat.
- b. Kekuatan data, data yang ditangkap dan disimpan, tetapi diorganisasikan sedemikian rupa sehingga laporan dan pengujian tidak dapat atau sulit untuk dilakukan.

### 3. *Economy* (Analisis Ekonomi)

Analisis ekonomi meliputi hal-hal sebagai berikut:

- a. Biaya
  - 1. Biaya yang tidak diketahui.
  - 2. Biaya tidak dapat dilacak ke sumber.
  - 3. Biaya terlalu tinggi.
- b. Keuntungan
  - 1. Pasar-pasar baru dapat dieksplorasikan.
  - 2. Pemasaran saat ini dapat diperbaiki.
  - 3. Pesanan-pesanan dapat ditingkatkan.

### 4. *Control* (Analisis Keamanan)

Analisis keamanan meliputi hal-hal sebagai berikut:

- a. Keamanan atau kontrol yang lemah
  - 1. Input data tidak diedit dengan cukup.
  - 2. Kejahatan.
  - 3. Pelanggaran etika pada data atau informasi.
  - 4. Data tersimpan secara berlebihan, tidak konsisten pada file-file atau database yang berbeda.
  - 5. Pelanggaran peraturan atau privasi data.
  - 6. Terjadi kesalahan saat pemrosesan (oleh manusia, mesin, atau perangkat lunak).

7. Terjadi kesalahan saat membuat keputusan.
  - b. Kontrol atau keamanan berlebihan
    1. Prosedur birokrasi memperlambat sistem.
    2. Pengendalian yang berlebihan.
    3. Pengendalian yang berlebihan menyebabkan penundaan pemerosesan.
5. *Efficiency* (Analisis Efisiensi)

Analisis efisiensi menyangkut bagaimana cara menghasilkan keluaran sebanyak-banyaknya dengan masukan yang seminimal mungkin. Berikut ini merupakan indikasi bahwa suatu sistem dapat dikatakan tidak efisien:

- a. Banyaknya waktu yang terbuang pada aktivitas sumberdaya manusia, mesin dan komputer.
  - b. Data dimasukan atau disalin secara berlebihan.
  - c. Informasi yang dihasilkan secara berlebihan.
  - d. Usaha yang dibutuhkan untuk tugas-tugas terlalu berlebihan.
  - e. Material yang dibutuhkan untuk tugas-tugas terlalu berlebihan.
6. *Service* (Analisis Layanan)

Berikut ini merupakan kriteria penilaian di mana kualitas sistem dapat dikatakan buruk:

- a. Sistem menghasilkan produk yang tidak akurat.
- b. Sistem menghasilkan produk yang tidak konsisten.
- c. Sistem menghasilkan produk yang tidak dapat dipercaya.
- d. Sistem tidak mudah untuk dipelajari.
- e. Sistem tidak mudah digunakan.
- f. Sistem canggung untuk digunakan.
- g. Sistem tidak fleksibel.

### **2.1.6 Unified Modeling Language**

*Unified Modelling Language* (UML) menurut Kendall dan Kendall (2014) menyediakan seperangkat alat standar yang dapat digunakan sebagai dokumentasi analisis dan desain sebuah sistem *software*. Sedangkan menurut Booch, dkk (2005) UML adalah bahasa standar untuk menulis *blueprint software*. UML dapat

digunakan untuk memvisualisasikan, menspesifikan, membangun dan mendokumentasikan. Menurut Dennis dkk (2012) *Unified Modelling Language* (UML) didefinisikan sebagai seperangkat empat belas teknik diagram yang digunakan untuk memodelkan sistem. Diagram dibagi menjadi dua kelompok utama yaitu pertama untuk melakukan pemodelan *structure* sistem dan satu lagi untuk pemodelan *behavior*. *structure diagram* menyediakan cara untuk mewakili data dan hubungan statis dalam sistem informasi dan *behavior diagram* menyediakan analisis dengan cara untuk menggambarkan hubungan dinamis antara contoh atau objek yang mewakili sistem informasi bisnis. Adapun penjelasan diagram UML secara jelas terdapat pada Tabel .3 sebagai berikut.

**Tabel 2.3** Penjelasan Diagram UML

<b>Nama Diagram</b>	<b>Kegunaan</b>	<b>Fase Primer</b>
<i>Class</i>	Mengilustrasikan hubungan antara model <i>class</i> dalam sistem	<i>Analysis, Design</i>
<i>Activity</i>	Mengilustrasikan alur kerja bisnis yang tidak tergantung pada <i>class</i> , aliran dari aktifitas dalam <i>use case</i> atau desain rinci suatu metode	<i>Analysis, Design</i>
<i>Sequence</i>	Model <i>behavior</i> dari objek dalam <i>use case</i> , berfokus pada pemesanan suatu aktifitas berdasarkan waktu	<i>Analysis, Design</i>
<i>Use Case</i>	Menangkap kebutuhan bisnis untuk sistem dan mengilustrasikan interaksi antara sistem dan lingkungannya	<i>Analysis</i>

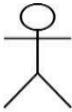
Sumber: (Dennis dkk, 2009)

a. *Use Case Diagram*

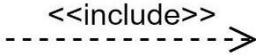
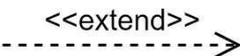
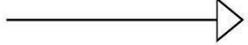
*Use case diagram* merupakan diagram yang mengekspresikan interaksi antara aktor dan *use case* di dalam sistem. *Use case diagram* digunakan untuk membuat daftar aktor dan *use case* untuk menunjukkan keterlibatan antara aktor dan *use case* di dalam sistem (Rumbaugh dkk, 2005). Berikut notasi yang digunakan dalam *use case diagram* dapat dilihat pada tabel 2.4.

**Tabel 2.4** Notasi *Use Case Diagram*

<b>Notasi</b>	<b>Nama</b>	<b>Keterangan</b>
---------------	-------------	-------------------

 Actor	<i>Actor</i>	Aktor adalah pengguna yang berinteraksi langsung dengan <i>use case</i>
--	--------------	---

**Tabel 2.4** Notasi *Use Case Diagram* (Lanjutan)

Notasi	Nama	Keterangan
 Use Case	<i>Use Case</i>	<i>Use case</i> merupakan bagian utama dari fungsionalitas sistem yang terlibat dengan aktor
 <<include>>	<i>Include</i>	Relasi <i>include</i> merupakan relasi yang merepresentasikan penyertaan fungsionalitas dari suatu <i>use case</i> di dalam <i>use case</i> lainnya.
 <<extend>>	<i>Extend</i>	Relasi <i>extend</i> adalah perpanjangan dari <i>use case</i> .
 Generalization	<i>Generalization</i>	Relasi <i>generalization</i> merupakan relasi yang merepresentasikan hubungan generalisasi antara <i>use case</i> umum dan <i>use case</i> spesifik yang mewarisi dan menambahkan fitur ke dalamnya
 Association	<i>Association</i>	Relasi asosiasi merupakan relasi yang merepresentasikan hubungan aktor yang berinteraksi dengan <i>use case</i> .

Sumber: (Dennis dkk, 2009)

b. *Activity Diagram*

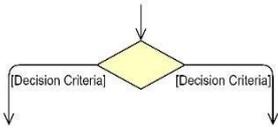
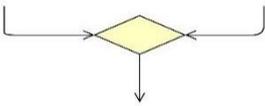
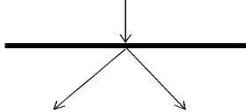
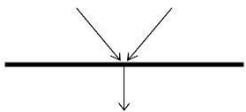
*Activity diagram* merupakan diagram yang menggambarkan serangkaian aktivitas yang menampilkan alur kontrol diantara aktivitas komputasi yang didalamnya terlibat aktivitas berupa *workflow* (Rumbaugh dkk, 2005). Berikut notasi yang digunakan dalam *activity diagram* dapat dilihat pada tabel 2.5.

**Tabel 2.5** Notasi *Activity Diagram*

Notasi	Nama	Keterangan
 Activity	<i>Activity</i>	<i>Activity</i> digunakan untuk merepresentasikan suatu serangkaian tindakan atau aktivitas.
 Control Flow	<i>Control Flow</i>	<i>Control flow</i> menunjukkan urutan eksekusi.

	<i>Initial Node</i>	<i>Initial node</i> menggambarkan mula dari suatu tindakan dan aktivitas.
---	---------------------	---

**Tabel 2.5** Notasi *Activity Diagram* (Lanjutan)

Notasi	Nama	Keterangan
	<i>Final Activity Node</i>	<i>Final activity node</i> digunakan untuk menyelesaikan aliran kontrol maupun objek dalam tindakan dan aktivitas.
	<i>Decision</i>	<i>Decision</i> digunakan untuk merepresentasikan kondisi pengujian untuk memastikan bahwa aliran kontrol atau aliran objek hanya turun satu jalan
	<i>Merge</i>	<i>Merge</i> digunakan untuk menyatukan kembali jalur keputusan yang berbeda yang dibuat menggunakan sebuah <i>node</i> keputusan.
	<i>Fork</i>	<i>Fork</i> digunakan untuk menunjukkan bahwa dua proses paralel yang bersamaan harus dieksekusi.
	<i>Join</i>	<i>Join</i> digunakan hanya untuk menyatukan kembali paralel yang terpisah atau aliran bersamaan dalam proses bisnis menjadi proses tunggal

Sumber: (Dennis dkk, 2009)

c. *Sequence Diagram*

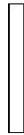
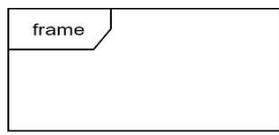
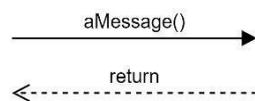
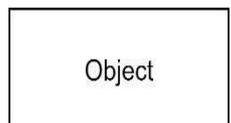
*Sequence Diagram* merupakan model dinamis yang menunjukkan urutan eksplisit pesan yang dikirimkan oleh objek dalam interaksi yang ditentukan. *Sequence diagram* mengilustrasikan objek yang berperan dalam *use case* dan pesan yang lewat diantara objek dari waktu ke waktu untuk satu kasus penggunaan (Dennis dkk, 2009). Berikut notasi yang digunakan dalam *sequence diagram* dapat dilihat pada tabel 2.6.

**Tabel 2.6** Notasi *Sequence Diagram*

Notasi	Nama	Keterangan
--------	------	------------

	<i>Actor</i>	<i>Actor</i> adalah orang yang memperoleh manfaat dari sistem dan berada diluar sistem. Aktor berpartisipasi secara berurutan dengan mengirim atau menerima pesan.
---	--------------	--

**Tabel 2.6** Notasi *Sequence Diagram* (Lanjutan)

Notasi	Nama	Keterangan
	<i>Execution Occurrence</i>	<i>Execution Occurrence</i> menunjukkan ketika suatu objek mengirim atau menerima pesan.
	<i>Frame</i>	<i>Frame</i> menunjukkan konteks <i>sequence diagram</i> .
	<i>Lifeline</i>	<i>Lifeline</i> menunjukkan alur kehidupan suatu objek berdasarkan urutan
	<i>Message</i>	<i>Message</i> menyampaikan informasi atau pesan dari satu objek ke objek lainnya
	<i>Object</i>	<i>Object</i> berpartisipasi secara berurutan dengan mengirim atau menerima pesan

Sumber: (Dennis dkk, 2009)

#### d. *Class Diagram*

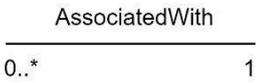
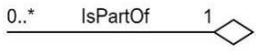
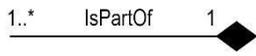
*Class diagram* merupakan model statis yang menunjukkan kelas-kelas dan relasi antar kelas di dalam sistem dari waktu ke waktu. *Class diagram* menggambarkan kelas kelas yang didalamnya mencakup perilaku dengan relasi antara kelas (Dennis dkk, 2009). Berikut notasi yang digunakan dalam *class diagram* dapat dilihat pada tabel 2.7.

**Tabel 2.7** Notasi *Class Diagram*

Notasi	Nama	Keterangan
--------	------	------------

<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center; padding: 2px;"><b>Classname</b></td> </tr> <tr> <td style="padding: 2px;">- attribute</td> </tr> <tr> <td style="padding: 2px;">+ operation()</td> </tr> </table>	<b>Classname</b>	- attribute	+ operation()	<i>Class</i>	<i>Class</i> merepresentasikan jenis orang, tempat, atau sesuatu yang perlu diambil dan disimpan oleh sistem informasi yang memiliki daftar atribut dan daftar operasi.
<b>Classname</b>					
- attribute					
+ operation()					

**Tabel 2.7** Notasi *Use Case Diagram* (Lanjutan)

Notasi	Nama	Keterangan
<i>Attribute name/ derived attribute name</i>	<i>Attribute</i>	<i>Attribute</i> merepresentasikan properti yang mendeskripsikan keadaan suatu objek dan dapat diturunkan dari atribut lain.
<i>Operation name()</i>	<i>Operation</i>	<i>Operation</i> merepresentasikan tindakan atau fungsi yang dapat kelas lakukan. Dapat dikelompokkan sebagai <i>constructor</i> , <i>query</i> atau operasi pembaruan.
<i>Operation name()</i>	<i>Operation</i>	<i>Operation</i> merepresentasikan tindakan atau fungsi yang dapat kelas lakukan. Dapat dikelompokkan sebagai <i>constructor</i> , <i>query</i> atau operasi pembaruan.
	<i>Association</i>	Relasi <i>association</i> merepresentasikan hubungan antara beberapa kelas atau kelas lainnya. Relasi ini dapat digunakan oleh satu atau banyak kelas.
	<i>Generalization</i>	Relasi <i>generalization</i> menyatakan hubungan <i>inheritance</i> dengan makna umum-khusus.
	<i>Aggregation</i>	Relasi <i>aggregation</i> merepresentasikan hubungan bagian-bagian yang logis diantara beberapa kelas dan kelas lainnya.
	<i>Composition</i>	Relasi <i>composition</i> merepresentasikan hubungan bagian-bagian yang fisikal antara beberapa kelas atau kelas lainnya.

Sumber: (Dennis dkk, 2009)

### 2.1.7 PHP

PHP atau *Hypertext Preprocessor* merupakan bahasa pemrograman yang dirancang dan dibangun untuk menghasilkan halaman *web* secara interaktif di

komputer yang disebut dengan *web server*. PHP membuat pengembangan *web* menjadi mudah karena semua kode yang dibutuhkan ada didalam kerangka PHP. Berbeda dengan HTML dimana menggunakan *tag* dan *markup* pada *web browser* untuk menghasilkan halaman *web*, pada *code* PHP dapat berjalan di antara halaman yang diminta dan *web server* dengan menambahkan dan mengubah *output* HTML dasar. PHP merupakan bahasa pemrograman yang bersifat *open source* sehingga siapa saja dapat menggunakannya karena tidak perlu membeli lisensi pengguna (Davis & Phillips, 2007).

### **2.1.8 Codeigniter**

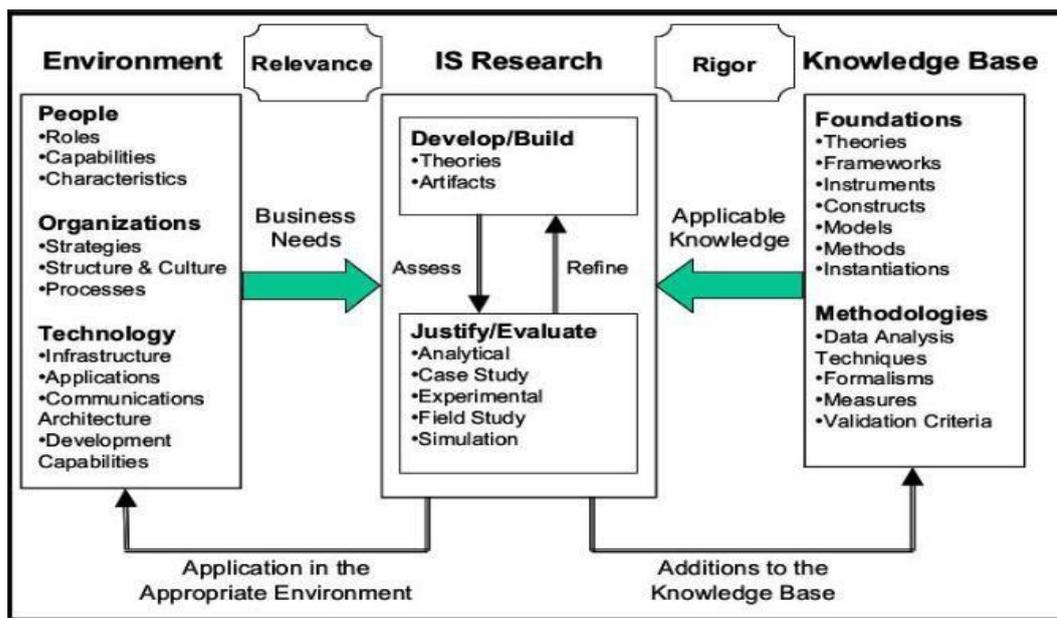
*Codeigniter* pertama kali dikembangkan oleh Rick Ellis pada tahun 2006 dengan logo api yang menyala. *Codeigniter* merupakan kerangka kerja dari sebuah *web application network* yang bersifat *open source* yang digunakan untuk pengembangan aplikasi berbasis PHP dinamis. *Codeigniter* adalah kerangka kerja PHP dengan menggunakan konsep MVC (*Model, View, Controller*) untuk membangun *web* dinamis menggunakan bahasa pemrograman PHP. *Codeigniter* memiliki dokumentasi yang lengkap disertai dengan contoh pengimplementasian kodenya. Adapun fungsi-fungsi *Codeigniter* diantaranya yaitu (idCloudhouse, 2017):

- a. Mempercepat dan memudahkan pengembang dalam pembuatan aplikasi berbasis *web*.
- b. Menghasilkan struktur pemrograman yang rapi baik dari segi *code* maupun struktur *file* PHPnya.
- c. Memberikan standar *coding* sehingga dapat memudahkan pengembang untuk mempelajari kembali aplikasi yang dibangun.

### **2.1.9 Kerangka Kerja IS Research**

Sistem informasi dan organisasi yang didukung bersifat kompleks, *artificial*, dan dirancang berdasarkan tujuan. Kerangka kerja IS *research* terdiri dari orang, struktur, teknologi, dan sistem kerja. Kerangka kerja IS *research* menyajikan kerangka kerja konseptual untuk memahami, melaksanakan dan mengevaluasi

penelitian sistem informasi yang mengkombinasikan paradigma *behavioral science* dan paradigma *design science*. Kontribusi dari *behavioral science* dan *design science* dalam penelitian sistem informasi dinilai saat mereka diterapkan pada kebutuhan bisnis di lingkungan yang sesuai dan saat menambah konten basis pengetahuan untuk penelitian dan praktik lebih lanjut (Hevner dkk, 2004).



**Gambar 2.1** Kerangka Kerja *Information System Research*

Sumber: (Hevner dkk, 2004)

### 2.1.10 Skala Likert

Skala likert menggunakan beberapa butir pertanyaan dan memiliki lima pilihan untuk setiap butir pertanyaan seperti "sangat setuju", "setuju", "tidak memutuskan", "tidak setuju", "sangat tidak setuju" untuk mengukur perilaku individu. Kemudahan dalam penggunaan skala Likert memungkinkan untuk menggunakannya secara lebih luas oleh peneliti. Kemudahan-kemudahan yang ada pada saat penyusunan skala likert perlu diperhatikan dengan seksama agar analisis lebih lanjut terhadap butir-butir respon dari setiap pertanyaan menjadi benar (Budiaji, 2013).

Skala likert merupakan skala pengukuran yang digunakan untuk mengukur persepsi, sikap maupun pendapat seseorang ataupun individu terkait suatu peristiwa

atau fenomena sosial. Terdapat 2 bentuk pertanyaan dalam skala likert, yaitu bentuk pertanyaan positif untuk mengukur skala positif dengan diberi skor 5,4,3,2,1 dan pertanyaan negatif untuk mengukur skala negatif dengan diberi skor 1,2,3,4,5 (Pranatawijaya dkk, 2019). Berikut tabel skala likert beserta skor dari masing-masing butir jawaban.

**Tabel 2.8** Skala Likert Beserta Skor Jawaban

Jawaban	Skor
Sangat Tidak Setuju (STS)	1
Tidak Setuju (TS)	2
Netral (N)	3
Setuju (S)	4
Sangat Setuju (SS)	5

## 2.2 Tinjauan Pustaka

Pada tinjauan pustaka bagian ini akan membandingkan penelitian yang sudah ada dan penelitian saat ini.

**Tabel 2.9** Tinjauan Pustaka Peneliti Pertama

<b>Judul Penelitian</b>	Sistem Pendukung Keputusan Kenaikan Jabatan Pegawai Negeri Sipil Pada Pusat Penelitian dan Pengembangan Minyak dan Gas Bumi Lemigas Kementerian ESDM Dengan Metode <i>Profile Matching</i>
<b>Penulis</b>	Harriyansah, Taufik Badawi. Tahun 2018
<b>Hasil Penelitian</b>	Pada penelitian ini dilakukan pembuatan sistem pendukung keputusan untuk menentukan kenaikan jabatan PNS pada kementerian ESDM menggunakan metode <i>profile matching</i> . Pada penelitian ini metode yang digunakan diantaranya observasi, wawancara dan studi pustaka. Dalam penelitian ini kriteria dan variabel penilaian untuk setiap profil pegawai sangat berpengaruh pada proses pengangkatan jabatan. Kriteria yang dihitung meliputi 4 aspek yaitu aspek kapasitas intelektual, Aspek sikap kerja, Aspek perilaku kerja, Aspek kecerdasan. Penelitian ini berhasil mengetahui hasil hitung penilaian untuk kenaikan jabatan yang dinilai dengan metode <i>profile matching</i>

**Tabel 2.10** Tinjauan Pustaka Peneliti Kedua

<b>Judul Penelitian</b>	Penerapan <i>Profile Matching</i> Untuk Penilaian Kinerja Pegawai Negeri Sipil (PNS)
<b>Penulis</b>	Titin Kristiana. Tahun 2015
<b>Hasil Penelitian</b>	Pada penelitian ini dilakukan pembuatan sebuah Sistem Pendukung Keputusan (SPK) dengan metode <i>profile matching</i> pada badan

	penelitian dan pengembangan ESDM. Pada penelitian ini metode yang digunakan diantaranya observasi, wawancara, studi pustaka dan kuisisioner. Penentuan standar nilai untuk profil penerimaan penghargaan sangat berpengaruh juga dalam proses pemilihan penerimaan abdi karya. Terdapat 7 kriteria yang digunakan. Perhitungan menggunakan metode <i>profile matching</i> untuk pemilihan penerima penghargaan menghasilkan satu pemenang.
--	--

**Tabel 2.11** Tinjauan Pustaka Peneliti Ketiga

<b>Judul Penelitian</b>	SPK Rekomendasi Pemilihan Kandidat Pejabat Struktural Menggunakan Metode <i>Profile Matching</i>
<b>Penulis</b>	Ria Astriratma, Retantyo Wardoyo, Aina Musdholifah. Tahun 2017
<b>Hasil Penelitian</b>	Pada penelitian ini dilakukan pembuatan sebuah Sistem Pendukung Keputusan (SPK) dengan metode <i>profile matching</i> pada Badan Keuangan Daerah (BKD). Dalam penelitian ini Sistem Pendukung Keputusan (SPK) yang diusulkan dapat menghasilkan urutan rekomendasi kandidat terbaik dalam pemilihan jabatan struktural di lingkungan Pemerintahan Kota Tarakan. Kriteria yang digunakan terdapat 6 aspek. Hasil penelitian membantu mengatasi masalah yang muncul karena sebelumnya dilakukan secara manual sehingga sistem ini mampu meminimalisir terjadinya keputusan yang subjektif

Setelah melakukan tinjauan pustaka terhadap penelitian yang serupa diatas, terdapat persamaan dengan penelitian yang dilakukan berkaitan dengan tema secara umum yaitu merancang sebuah Sistem Pendukung Keputusan (SPK) dengan menggunakan metode *profile matching*. Perbedaan yang pada penelitian terdahulu adalah studi kasus yang merupakan instansi pemerintah, sedangkan pada penelitian ini studi kasus dilakukan pada Badan Kepegawaian dan Pengembangan Sumber Daya Manusia (BPKSDM) pada wilayah Kayong Utara. Perbedaan pada penelitian terdahulu adalah kriteria yang digunakan untuk menentukan kenaikan jabatan struktural bagi Pegawai Negeri Sipil.