

BAB II

TINJAUAN PUSTAKA

2.1 Kajian Pustaka

Tabel 2.1. Kajian Pustaka

1.	<p>Nama Penulis</p> <p>Tahun</p> <p>Judul</p> <p>Nama Jurnal</p> <p>Metode</p> <p>Hasil</p>	<p>Afshari A.R</p> <p>2012</p> <p><i>Project Manager Selection by Using Fuzzy simple Additive Weighting Method</i></p> <p><i>IEEE Xplore Digital Library</i></p> <p>Topik: <i>Multi Criteria Decision Making</i> ; Metode SPK: <i>Fuzzy Simple Additive Weighting</i>; Variabel-variabel yang digunakan: <i>Basic Requirements, Project Management Skills, Management skills, Interpersonal skills</i>; Metode pengumpulan data: Kuesioner;</p> <p>Membahas tentang SPK pemilihan manajer proyek menggunakan metode SAW, penelitian ini membangun model <i>hybrid</i> untuk pemilihan manajer proyek menggunakan sistematik baru yaitu metode Delphi. Di penelitiannya belum membahas tentang implementasi ke dalam sistem yang bisa digunakan oleh perusahaan.</p>
----	---	---

2.	<p>Nama Penulis</p> <p>Tahun</p> <p>Judul</p> <p>Nama Jurnal</p> <p>Metode</p> <p>Hasil</p>	<p>Denay Islam Sabanayo</p> <p>2014</p> <p><i>Sistem Pendukung Keputusan Pemilihan Karyawan Terbaik Menggunakan Metode SAW Pada PT. Berkah Cahaya Muria Kudus</i></p> <p>UDINUS Repository</p> <p>Topik: <i>Decision Support System</i> Karyawan Terbaik; Metode SPK: <i>Simple Additive Weighting (SAW)</i>; Variabel-variabel yang digunakan: Kedisiplinan, Kreativitas, Alpha, Kerjasama, Ketertiban; Metode pengumpulan data: Observasi, Wawancara, Studi Pustaka.</p> <p>Merupakan penelitian yang membuat SPK dalam penentuan karyawan terbaik pada PT Berkah Cahaya Muria Kudus. Sistem ini hanya dapat menggunakan kriteria-kriteria yang sudah ditentukan dan untuk bobot setiap kriteria bersifat dinamis (bisa berubah sesuai kebijakan perusahaan).</p>
3.	<p>Nama Penulis</p> <p>Tahun</p> <p>Nama Jurnal</p> <p>Judul</p>	<p>Bambang Efiriyanto</p> <p>2016</p> <p>Electronic Theses and Dissertations Universitas Muhamadiyah Surakarta</p> <p><i>Sistem Pendukung Keputusan Menggunakan Metode Simple Additive Weighting (SAW) Untuk Menentukan Karyawan Terbaik Pada Dealer Motor Berbasis Web</i></p>

	Metode	Topik: <i>Decision Support System</i> Pemilihan Karyawan Terbaik ; Metode SPK: <i>Simple Additive Weighting (SAW)</i> ; Variabel-variabel yang digunakan: Kedisiplinan, kreatifitas, alpa, kerjasama, dan ketertiban; Metode pengumpulan data: Observasi dan Wawancara;
	Hasil	Merupakan penelitian yang membuat SPK dalam penentuan karyawan terbaik pada Dealer Motor. Penelitian ini dirancang untuk menghitung bobot dari setiap alternatif dan kriteria yang ada.
4.	Nama Penulis	Raden Ajeng Yosua Ariane Amos Wiseso
	Tahun	2018
	Judul	<i>Sistem Pendukung Keputusan Dalam Penentuan Penerima Beasiswa PT BFI Finance Indonesia Tbk Menggunakan Metode Profile Matching.</i>
	Nama Jurnal	<i>Ultima Infosys: Jurnal Sistem Informasi</i>
	Metode	Topik: <i>Decision Support System</i> Penerima Beasiswa; Metode SPK: <i>Profile Matching</i> ; Variabel-variabel yang digunakan: IPK, Prestasi, Psikotes, Wawancara; Metode pengumpulan data: Observasi, Wawancara, Studi Pustaka.
	Hasil	Merupakan penelitian yang membuat SPK dalam penentuan penerima beasiswa pada PT BFI Finance Indonesia Tbk. Penelitian ini dirancang untuk menentukan variabel gap kompetensi, menghitung hasil pemetaan gap kompetensi, perhitungan & pengelompokan <i>core</i> dan <i>secondary factor</i> kemudian perhitungan nilai total.

Berdasarkan Tabel 2.1 digunakan 4 referensi sebagai kajian pustaka. Melalui jurnal nomor 1 yang berjudul *Project Manager Selection by Using Fuzzy simple Additive Weighting Method* diambil penjelasan latar belakang mengenai bagaimana seharusnya sistem dibuat.

Melalui jurnal nomor 2 yang berjudul *Sistem Pendukung Keputusan Pemilihan Karyawan Terbaik Menggunakan Metode SAW Pada PT. Berkah Cahaya Muria Kudus* diambil penjelasan mengenai setiap metode pengumpulan data.

Melalui jurnal nomor 3 yang berjudul *Sistem Pendukung Keputusan Menggunakan Metode Simple Additive Weighting (SAW) Untuk Menentukan Karyawan Terbaik Pada Dealer Motor Berbasis Web* diambil penjelasan latar belakang mengenai apa fungsi suatu sistem pendukung keputusan

Melalui jurnal nomor 4 yang berjudul *Sistem Pendukung Keputusan Dalam Penentuan Penerima Beasiswa PT BFI Finance Indonesia Tbk Menggunakan Metode Profile Matching* diambil metode pengembangan sistem yang sama yaitu metode *Waterfall*.

2.2 Landasan Teori

2.2.1 Sistem Pendukung Keputusan (SPK)

Sistem Pendukung Keputusan merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan, dan pemanipulasian data. Sistem ini digunakan untuk membantu pengambilan keputusan dalam situasi yang semi terstruktur dan situasi yang tidak terstruktur, di mana tidak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat (Hidayat & Baihaqi, 2016). SPK dirancang untuk menunjang seluruh tahapan pembuatan keputusan yang dimulai dari tahap mengidentifikasi masalah, memilih data yang relevan, menentukan pendekatan yang digunakan dalam proses pembuatan keputusan, sampai pada kegiatan mengevaluasi pemilihan alternatif. Berikut beberapa contoh metode yang digunakan dalam Sistem Pendukung Keputusan:

a. Metode *Simple Additive Weighting (SAW)*

Metode *Simple Additive Weighting (SAW)* atau yang sering juga dikenal dengan istilah metode penjumlahan terbobot. Konsep dasar metode *SAW* adalah mencari penjumlahan terbobot dari *rating* kinerja pada setiap alternatif pada semua atribut. Metode *SAW* membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua *rating* alternatif yang ada. Metode ini merupakan metode yang paling terkenal dan paling banyak digunakan dalam menghadapi situasi *Multiple Attribute Decision Making (MADM)*. *MADM* itu sendiri merupakan suatu metode yang digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan kriteria tertentu.

b. Metode *Analytic Hierarchy Process (AHP)*

Analytic Hierarchy Process (AHP) adalah sebuah proses yang membantu para pengambil keputusan untuk memperoleh solusi terbaik dengan mendekomposisikan permasalahan kompleks menjadi bentuk yang lebih sederhana untuk kemudian dilakukan sintesis terhadap berbagai faktor yang terlibat dalam permasalahan pengambilan keputusan tersebut. *AHP* mempertimbangkan aspek kualitatif dan kuantitatif dari suatu keputusan dan juga mengurangi kompleksitas suatu keputusan dengan membuat perbandingan satu-satu dari berbagai kriteria yang dipilih untuk kemudian diolah dan diperoleh hasilnya.

c. Metode *PROMETHEE*

PROMETHEE adalah salah satu metode penentuan urutan atau prioritas dalam analisis multikriteria atau *MCDM (Multi Criterion Decision Making)*. Dugaan dari dominasi kriteria yang digunakan dalam *PROMETHEE* adalah penggunaan nilai dalam hubungan outranking. Masalah pokoknya adalah kesederhanaan, kejelasan dan kestabilan. Semua parameter yang dinyatakan mempunyai pengaruh nyata menurut pandangan ekonomi.

d. *Bayesian Decision Theory*

Bayesian Decision Theory adalah pendekatan secara statistik untuk menghitung *tradeoffs* diantara keputusan yang berbeda-beda, dengan

menggunakan probabilitas dan *costs* yang menyertai suatu pengambilan keputusan tersebut. *Bayesian probability* adalah teori terbaik dalam menghadapi masalah estimasi dan penarikan kesimpulan. *Bayesian method* dapat digunakan untuk penarikan kesimpulan pada kasus-kasus dengan *multiple source of measurement* yang tidak dapat ditangani oleh metode lain seperti model hierarki yang kompleks.

e. Metode Regresi Linier

Regresi Linear Sederhana adalah Metode Statistik yang berfungsi untuk menguji sejauh mana hubungan sebab akibat antara variabel faktor penyebab (X) terhadap variabel akibatnya. Faktor penyebab pada umumnya dilambangkan dengan X atau disebut juga dengan *predictor* sedangkan variabel akibat dilambangkan dengan Y atau disebut juga dengan *response*.

2.2.2 Multiple Attribute Decision Making (MADM)

Multiple Attribute Decision Making (MADM) adalah suatu metode yang digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan kriteria tertentu. Inti dari MADM adalah menentukan nilai bobot untuk setiap atribut, kemudian dilanjutkan dengan proses pemberian peringkat yang akan menyeleksi alternatif yang sudah diberikan (Setiaji, 2012). Banyak kasus dengan MADM menggunakan metode SAW untuk mencari sebuah alternatif. Setiaji (2012) memaparkan bahwa masalah yang sering terjadi adalah sulitnya memilih metode mana yang paling relevan untuk menyelesaikan suatu masalah dengan

menggunakan model MADM. Metode SAW merupakan metode MADM yang paling sederhana dan paling banyak digunakan.

Metode ini juga metode yang paling mudah untuk diaplikasikan, karena mempunyai algoritma yang tidak terlalu rumit. Metode SAW sering juga dikenal sebagai metode penjumlahan yang memberikan bobot. Konsep dasar metode SAW adalah mencari penjumlahan, kemudian memberikan bobot dari *rating* kinerja pada setiap alternatif pada semua atribut. Metode SAW membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua *rating* alternatif yang ada. Tahapan dalam metode SAW adalah sebagai berikut:

1. Menentukan alternatif $A = \{A_1, A_2, \dots, A_i\}$
2. Menentukan kriteria yang akan dijadikan acuan dalam pengambilan keputusan $C = \{C_1, C_2, \dots, C_j\}$
3. Menentukan rating kecocokan setiap alternatif pada setiap kriteria.
4. Menentukan bobot preferensi atau tingkatan kepentingan (W) setiap kriteria. $W = \{W_1, W_2, W_3, \dots, W_j\}$
5. Membuat matriks keputusan berdasarkan kriteria (C_j), kemudian melakukan normalisasi matriks berdasarkan persamaan yang disesuaikan dengan jenis atribut (atribut *benefit* atau atribut *cost*) sehingga diperoleh matriks ternormalisasi r .

$$r_{ij} = \frac{X_{ij}}{\text{Max}(X_{ij})} \quad \text{jika } j \text{ adalah atribut } \textit{benefit}.$$

Rumus 2.1

$$r_{ij} = \frac{\text{Min}(X_{ij})}{X_{ij}} \quad \text{jika } j \text{ adalah atribut } \textit{cost}.$$

Di mana r_{ij} adalah *rating* kinerja ternormalisasi dari alternatif A_i pada atribut C_j ; $i = 1, 2, 3, \dots, m$ dan $j = 1, 2, 3, \dots, m$. Nilai preferensi alternatif (V_i) diberikan sebagai

Rumus 2.2

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

Nilai V yang lebih besar, mengindikasikan bahwa alternatif A_i lebih terpilih.

2.2.3 Sistem Informasi Management

Sistem informasi adalah suatu instrumen penting untuk menciptakan nilai bagi perusahaan. Sistem Informasi memungkinkan perusahaan untuk meningkatkan pendapatannya atau mengurangi biayanya dengan menyediakan informasi yang membantu manajer membuat keputusan yang lebih baik dan juga meningkatkan pelaksanaan proses bisnis (Laudon, 2017).

Menurut Laudon (2017) manajemen sendiri mencakup proses perencanaan, pengorganisasian, pengawasan, pengarahan, dan lain-lain, dalam suatu organisasi. Sedangkan, informasi dalam suatu organisasi merupakan data yang diolah sedemikian rupa sehingga memiliki nilai dan arti bagi organisasi tersebut.

Dengan demikian, dapat disimpulkan bahwa Sistem Informasi Manajemen (SIM) merupakan sistem yang mengolah serta mengorganisasikan data dan informasi yang berguna untuk mendukung pelaksanaan tugas dalam suatu organisasi.

Berikut merupakan beberapa contoh konkret penerapan sistem informasi manajemen dalam suatu organisasi:

a. *Enterprise Resource Planning (ERP)*

Sistem ERP ini biasanya digunakan oleh sejumlah perusahaan besar dalam mengelola manajemen dan melakukan pengawasan yang saling terintegrasi terhadap unit bidang kerja Keuangan, *Accounting*, Sumber Daya Manusia, Pemasaran, Operasional, dan Pengelolaan Persediaan.

b. *Supply Chain Management (SCM)*

Sistem SCM ini sangat bermanfaat bagi pihak manajemen di mana data data yang disajikan terintegrasi mengenai manajemen suplai bahan baku, mulai dari pemasok, produsen, pengecer hingga konsumen akhir.

c. *Transaction Processing System (TPS)*

TPS ini berguna untuk memproses data dalam jumlah yang besar dengan transaksi bisnis yang rutin. Program ini biasanya diaplikasikan untuk manajemen gaji dan inventaris. Contohnya adalah aplikasi yang digunakan untuk Bantuan Keuangan Desa Pemprov Jawa Timur.

d. *Office Automation System (OAS)*

Sistem aplikasi ini berfungsi untuk melancarkan komunikasi antar departemen pada suatu perusahaan dengan cara mengintegrasikan *server-server* komputer dari setiap pengguna yang terlibat dalam perusahaan. Contohnya adalah *email*.

e. *Knowledge Work System (KWS)*

Sistem informasi KWS ini mengintegrasikan suatu pengetahuan baru ke dalam organisasi. Dengan ini, diharapkan para tenaga ahli dapat menerapkannya dalam pekerjaan mereka.

f. *Informatic Management System (IMS)*

IMS berfungsi untuk mendukung spektrum tugas-tugas dalam organisasi, yang juga dapat digunakan untuk membantu menganalisis pembuatan keputusan. Sistem ini juga dapat menyatukan beberapa fungsi informasi dengan program komputerisasi, seperti *e-procurement*.

g. *Decision Support System (DSS)*

Sistem ini membantu para manajer dalam mengambil keputusan dengan cara mengamati lingkungan dalam perusahaan. Contohnya, Link Elektronik di sekolah Tunas Bangsa, yang mengamati jumlah pendapatan atau pendaftaran siswa baru setiap tahun.

h. *Expert System (ES) dan Artificial Intelligent (A.I.)*

Sistem ini pada dasarnya menggunakan kecerdasan buatan untuk menganalisis pemecahan masalah dengan menggunakan pengetahuan tenaga ahli yang telah diprogram ke dalamnya. Contohnya, sistem jadwal mekanik.

2.2.4 *User Interface*

Ben Schneiderman adalah seorang ilmuwan komputer dari Amerika dan Professor Teknik Informatika di University of Maryland Human-Computer Interaction Lab, College Park, ia mengemukakan 8 aturan emas dalam menyusun suatu *user interface* yaitu:

1. *Strive for consistency.*

Konsistensi sangat dibutuhkan dalam membuat suatu *user interface*. Gunanya adalah supaya *user* nyaman dan tidak kebingungan dalam mengeksplorasi tiap halaman. Contohnya yaitu penggunaan *font* yang sama pada setiap halaman *website* agar pengguna tidak kebingungan, serta menempatkan *Header* dan *Footer* pada letak yang sama di tiap halaman

2. *Enable frequent users to use shortcuts.*

Dalam merancang *user interface* aplikasi, harus dibuatkan *shortcut* dalam menyelesaikan suatu fungsi / tugas. Gunanya adalah supaya *user* dapat lebih menghemat waktu dan cepat menjalankan suatu fungsi.

3. *Offer informative feedback*

Dalam merancang *user interface*, harus dibuatkan *Informative feedback* setiap kali *user* melakukan aksi. Tidak harus selalu dengan informasi dari aplikasi ke *user*, tetapi dapat berupa perubahan antarmuka setiap *user* melakukan aksi, dengan demikian *user* paham bahwa aksinya sudah direspons oleh aplikasi. Contohnya apabila *user* mengklik tombol “tambah data” maka akan pindah halaman ke pengisian *form* tambah data.

4. *Design dialogs to yield closure*

Poin ini termasuk dari bagian *informative feedback*, maksudnya adalah dibuatkan suatu *dialog box* bila proses yang dijalankan oleh *user* sudah selesai, *user* paham bahwa dia tidak perlu menunggu apakah masih akan ada tahapan lain setelah menyelesaikan suatu proses. Contohnya jika *user* sudah mengisi *form* yang diperlukan untuk tambah data, maka muncul pemberitahuan bahwa *user* berhasil menambahkan data.

5. *Prevent errors*

Poin ini untuk menjaga agar *user* tidak melakukan kesalahan dalam menjalankan suatu aksi. Adanya pemberitahuan mengenai apa yang harus dilakukan apabila aksi tidak dapat dijalankan. Contohnya saat *user* ingin *login*, apabila *user* tidak mengisi *username* dan *password* maka akan muncul pesan *error*.

6. Permit easy reversal of actions

Poin ini untuk memudahkan *user* jika ingin membatalkan tindakan. Biasanya yang dianggap sebagai pemenuhan poin ini adalah tombol *back*. Namun sebenarnya, tombol *back* hanya untuk kembali ke halaman sebelumnya, namun tidak membatalkan aksi. Contoh dari poin ini adalah pada aplikasi *online shop*, *user* dapat mencoba melakukan pembelian, tetapi pada saat selesai memilih barang, *user* dapat melakukan pembatalan barang yang ingin dibeli. Pentingnya dibuatkan seperti ini adalah agar *user* merasa aman dalam mengeksplorasi fungsi dalam aplikasi.

7. Support internal locus of control

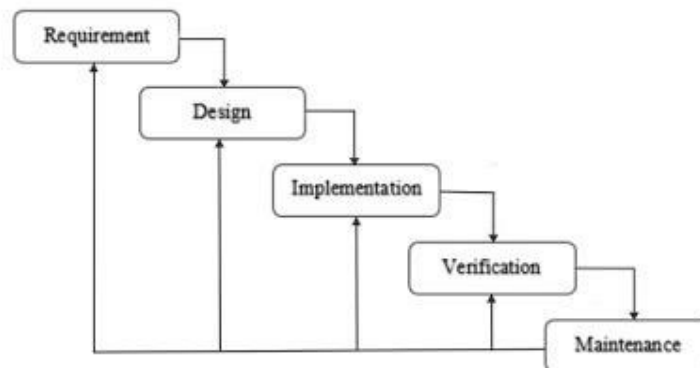
Poin ini untuk memudahkan *user* untuk bebas bernavigasi dan mengatur/mengubah informasi akun yang dimiliki sesuai yang diinginkan.

8. Reduce short-term memory load

Poin ini memfokuskan agar desain *user interface* dibuatkan sesederhana mungkin dan mudah dimengerti oleh *user*. Contohnya penataan lokasi tombol-tombol untuk menjalankan aksi diletakkan di tempat yang normal.

2.2.5 Metode Waterfall

Metode air terjun atau yang sering disebut metode *waterfall* sering dinamakan siklus hidup klasik (*classic life cycle*), dimana hal ini menggambarkan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak, dimulai dengan spesifikasi kebutuhan pengguna lalu berlanjut melalui tahapan-tahapan perencanaan (*planning*), permodelan (*modeling*), konstruksi (*construction*), serta penyerahan sistem ke para pelanggan/pengguna (*deployment*), yang diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan (Pressman, 2015). Tahapan metode *waterfall* dapat dilihat pada gambar di bawah ini.



Gambar 2.1. Metode *Waterfall*

2.2.6 *Unified Modelling Language*

Menurut (Windu & Gata, 2013), *Unified Modeling Language* (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk

mendukung pengembangan sistem. Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

a. *Use Case Diagram*

Use case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

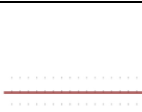
b. Diagram Aktivitas (*Activity Diagram*)



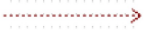


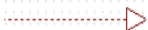
Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis.

c. Diagram Kelas (*Class Diagram*)

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem. *Class Diagram* juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. *Class Diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. Berikut merupakan symbol-simbol pada class diagram.

Tabel 2.2. Simbol *Class Diagram*

GAMBAR	NAMA	KETERANGAN
	<i>Association</i>	Hubungan statis antar kelas. Asosiasi menggambarkan kelas yang memiliki atribut berupa kelas lain.

	<i>Directed Association</i>	Asosiasi dengan makna kelas yang satu digunakan oleh kelas yang lain.
	<i>Generalization</i>	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus) atau untuk menyatakan <i>inheritance</i>
	<i>Dependency</i>	Relasi antar kelas dengan makna kebergantungan antar kelas
	<i>Agregation</i>	Hubungan yang menyatakan bahwa suatu kelas menjadi atribut bagi kelas lain
	<i>Composition</i>	Bentuk khusus dari agregasi dimana kelas yang menjadi bagian diciptakan setelah kelas menjadi <i>whole</i> .
	<i>Realization</i>	Hubungan antara kelas dimana sebuah kelas memiliki keharusan untuk mengikuti aturan yang ditetapkan oleh kelas lainnya.

2.2.7 Tujuan Pemanfaatan UML

Berikut merupakan tujuan utama dalam desain UML (Haviluddin, 2011) :

1. Menyediakan bagi pengguna (analisis dan desain sistem) suatu bahasa pemodelan visual yang ekspresif sehingga mereka dapat mengembangkan dan melakukan pertukaran model data yang bermakna.
2. Menyediakan mekanisme yang spesial untuk memperluas konsep inti.

3. Karena merupakan bahasa pemodelan visual dalam proses pembangunannya maka UML bersifat independen terhadap bahasa pemrograman tertentu.
4. Memberikan dasar formal untuk pemahaman bahasa pemodelan.
5. Mendorong pertumbuhan pasar terhadap penggunaan alat desain sistem yang berorientasi objek (OO / *Object Oriented*).
6. Mendukung konsep pembangunan tingkat yang lebih tinggi seperti kolaborasi, kerangka, pola dan komponen terhadap suatu sistem.
7. Memiliki integrasi praktik terbaik.

2.2.8 Black Box Testing

Pengujian menggunakan sekumpulan aktifitas validasi disebut dengan pendekatan black box testing. Menurut (Rosa & Salahuddin, 2013), black box testing adalah menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk melakukan pengujian black box testing harus dibuat dengan kasus benar dan kasus salah.