

BAB II

TINJAUAN PUSTAKA

2.1 Literature Review

Beberapa penelitian yang terkait dengan penelitian ini terutama dari basis aplikasi dan kemiripan obyek yang diteliti adalah sebagai berikut. Penelitian pertama dilakukan oleh Iswanto dan Abdul Aziz Muslim, tahun (2018) dengan judul “Perancangan dan Implementasi Rekrutmen Karyawan Menggunakan Metode SAW berbasis Web (Studi Kasus PT. Ateja)”. Adapun tujuan penelitian ini adalah untuk membantu perusahaan dalam meningkatkan efisiensi waktu pada penerimaan karyawan baru dan meningkatkan efektivitas pengambilan keputusan untuk memecahkan masalah penerimaan karyawan baru. Hasil dari penelitian ini berupa sistem rekrutmen karyawan berbasis web yang membuat proses perekrutan karyawan di PT. Ateja menjadi lebih efisien dan efektif [2].

Penelitian selanjutnya oleh Nia Nuraeni (2018), yang berjudul “Penerapan Metode *Simple Additive Weighting* (SAW) Dalam Seleksi Calon Karyawan”. Penelitian ini bertujuan untuk menghasilkan sistem pendukung keputusan yang dapat membantu dalam proses seleksi di PT. Dolarindo Intravalas Primatama. Hasil dari penelitian ini menunjukkan bahwa penggunaan metode *Simple Additive Weighting* (SAW) memiliki keakuratan tinggi sebesar 81%, sehingga dapat diaplikasikan dalam proses perekrutan calon karyawan dibandingkan dengan penilaian tes seleksi karyawan secara manual [3].

Penelitian berikutnya oleh Andika Invari Candra Dewi pada tahun (2019) yang berjudul “Implementasi Metode SAW (*Simple Additive Weighting*) Pada Sistem Seleksi Penerimaan Karyawan CV Parama Creative”. Tujuan penelitian adalah untuk membuat sistem seleksi karyawan dengan menggunakan metode SAW untuk membantu perusahaan dalam membuat keputusan. Hasil dari penelitian ini menunjukkan bahwa tingkat kecocokan keputusan yang dihasilkan oleh sistem pendukung keputusan hampir mendekati kecocokan dari manajer HRD dengan nilai sebesar 93% yang artinya termasuk didalam zona akurasi yang Sangat Baik berdasarkan skala nilai *Area Under Curve*(AUC) [4].

Adapun penelitian yang dilakukan oleh Desi Pibiana (2020) dengan judul “Penggunaan Metode *Simple Additive Weighting* (SAW) dalam Pengambilan Keputusan Rekrutmen Karyawan Pada PT. ABC”, dengan tujuan penelitian untuk memanfaatkan metode SAW dalam rangka membantu bagian HRD dalam mengambil keputusan dengan lebih objektif terkait rekrutmen karyawan baru. Hasil dari penelitian adalah perbandingan terhadap sejumlah alternatif calon karyawan dan mendapatkan hasil berupa daftar calon karyawan yang berhak untuk menempuh proses wawancara yang merupakan rangkaian dari proses rekrutmen karyawan setelah proses administrasi [1].

Penelitian lain dilakukan oleh Sifa Pradina (2023) dengan judul “Implementasi Metode *Simple Additive Weighting* (SAW) Pada Sistem Seleksi Penerimaan Calon Karyawan Baru (Studi Kasus: PT. Dalang Digital Studio)”. Penelitian ini bertujuan untuk membangun sistem seleksi penerimaan calon karyawan baru di PT. Dalang Studio dan menerapkan metode *Simple Additive Weighting* (SAW). Dalam penelitian ini, memperoleh hasil bahwa metode *Simple Additive Weighting* (SAW) terbukti dapat digunakan untuk membangun sistem seleksi penerimaan calon karyawan baru pada PT. Dalang Digital Studio karena kriteria dan bobot yang dimiliki sesuai dengan kebutuhan perusahaan sehingga pengambilan keputusan menjadi lebih objektif. Selain itu, penerapan metode *Simple Additive Weighting* (SAW) juga membantu dalam pengambilan keputusan karena adanya proses perbandingan setelah menentukan nilai bobot untuk setiap atribut dan pemilihan alternatif terbaik dari sekumpulan alternatif [5].

Tabel 2.1 Perbandingan Penelitian

No	Judul	Tahun	Peneliti	Hasil Penelitian
1	Perancangan dan Implementasi Rekrutmen Karyawan Menggunakan Metode SAW Berbasis Web (Studi Kasus PT. Ateja)	2018	Iswanto dan Abdul Aziz Muslim	Hasil dari penelitian ini berupa sistem rekrutmen karyawan berbasis web yang membuat proses perekrutan karyawan di PT. Ateja menjadi lebih efisien dan efektif.
2	Penerapan Metode Simple Additive Weighting (SAW) Dalam Seleksi Calon Karyawan	2018	Nia Nuraeni	Hasil dari penelitian menunjukkan penggunaan metode <i>Simple Additive Weighting (SAW)</i> memiliki keakuratan tinggi sebesar 81%, sehingga dapat diaplikasikan dalam proses perekrutan calon karyawan dibandingkan dengan penilaian tes seleksi karyawan secara manual.
3	Implementasi Metode SAW (<i>Simple Additive Weighting</i>) Pada Sistem Seleksi Penerimaan Karyawan CV Parama Creative	2019	Andika Invari Candra Dewi	Hasil dari penelitian ini menunjukkan bahwa tingkat kecocokan keputusan yang dihasilkan oleh sistem pendukung keputusan hampir mendekati kecocokan dari manajer HRD dengan nilai sebesar 93% yang artinya termasuk didalam zona akurasi yang Sangat Baik berdasarkan skala nilai <i>Area Under Curve (AUC)</i> .

Tabel 2.1 Perbandingan Penelitian

No	Judul	Tahun	Peneliti	Hasil Penelitian
4	Penggunaan Metode <i>Simple Additive Weighting</i> (SAW) dalam Pengambilan Keputusan Rekrutmen Karyawan Pada PT. ABC	2020	Desi Pibriana	Hasil dari penelitian adalah perangkingan terhadap sejumlah alternatif calon karyawan dan mendapatkan hasil berupa daftar calon karyawan yang berhak untuk menempuh proses wawancara yang merupakan rangkaian dari proses rekrutmen karyawan setelah proses seleksi administrasi.
5	Implementasi Metode <i>Simple Additive Weighting</i> (SAW) Pada Sistem Seleksi Penerimaan Calon Karyawan Baru (Studi Kasus: PT. Dalang Digital Studio)	2023	Sifa Pradina	Hasil dari penelitian menunjukkan bahwa Metode <i>Simple Additive Weighting</i> (SAW) terbukti dapat digunakan untuk membangun sistem seleksi penerimaan calon karyawan baru pada PT. Dalang Digital Studio karena kriteria dan bobot yang dimiliki sesuai dengan kebutuhan perusahaan sehingga pengambilan keputusan menjadi lebih objektif.

2.2 Landasan Teori

Pada bagian ini akan dijelaskan mengenai dasar-dasar teori yang berhubungan dengan permasalahan yang dibahas pada tugas akhir. Adapun dasar-dasar teori yang digunakan sebagai berikut:

2.2.1 Konsep Dasar Aplikasi

2.2.1.1 Pengertian Aplikasi

Aplikasi merupakan sebuah perangkat lunak yang diciptakan untuk tujuan tertentu dengan maksud untuk mempermudah tugas-tugas manusia [6]. Menurut [7], aplikasi adalah suatu perangkat lunak yang diinstal pada komputer dan berisi serangkaian perintah yang dirancang untuk melaksanakan berbagai jenis pekerjaan atau tugas tertentu, seperti pengolahan data, penggunaan aplikasi, dan penambahan informasi. Sedangkan menurut [8], aplikasi adalah salah satu komponen dari perangkat lunak yang memiliki fungsi untuk mengatasi permasalahan khusus yang dihadapi oleh pengguna dengan memanfaatkan teknologi komputer. Dari pengertian diatas, dapat disimpulkan bahwa aplikasi merupakan sebuah perangkat lunak yang terdiri dari serangkaian perintah khusus dan diciptakan dengan tujuan untuk mempermudah tugas-tugas manusia.

2.2.2 Rekrutmen

Rekrutmen adalah suatu proses yang dimulai dari kebutuhan untuk mengisi posisi pekerjaan yang tersedia hingga pelamar mengirimkan berkas lamaran mereka [9]. Menurut [10], Rekrutmen adalah proses yang melibatkan analisis persyaratan pekerjaan, pencarian, dan pengumpulan calon atau kandidat karyawan yang memenuhi syarat, serta penarikan calon atau kandidat yang sesuai dengan kebutuhan perusahaan. Tujuan utama dari perekrutan adalah untuk mengisi posisi yang tersedia dengan individu yang memiliki kualitas dan kompetensi yang tepat, sehingga perusahaan dapat mencapai keunggulan kompetitif melalui kekuatan tenaga kerja yang berkualitas. Sedangkan menurut [11], Rekrutmen merupakan proses penarikan karyawan yang berkaitan dengan memenuhi kebutuhan karyawan dalam hal jumlah dan kualitas, baik dalam jangka pendek, menengah, maupun panjang. Dari penjelasan sebelumnya dapat disimpulkan bahwa Rekrutmen adalah

suatu proses yang dimulai dari kebutuhan untuk mengisi posisi pekerjaan yang tersedia, melibatkan analisis persyaratan pekerjaan, pencarian, pengumpulan kandidat karyawan yang memenuhi syarat, serta penarikan kandidat yang sesuai dengan kebutuhan perusahaan sehingga memperoleh individu yang berkualitas dan mencapai keunggulan kompetitif.

Ada dua sumber utama rekrutmen karyawan yaitu internal dan eksternal. Rekrutmen internal merupakan pencarian kandidat atau calon karyawan dari karyawan-karyawan yang sudah ada dalam organisasi, sedangkan rekrutmen karyawan eksternal adalah merekrut karyawan dari luar organisasi [9].

2.2.3 Metode *Simple Additive Weighting* (SAW)

2.2.3.1 Definisi Metode *Simple Additive Weighting* (SAW)

Metode *Simple Additive Weighting* (SAW) atau dikenal dengan istilah metode penjumlahan terbobot adalah metode untuk menyelesaikan masalah penyeleksian dalam sistem pengambilan keputusan multiproses [12]. Prinsip dasar dari metode *Simple Additive Weighting* (SAW) adalah melakukan penjumlahan terbobot dari rating kinerja pada setiap alternatif untuk setiap atribut yang ada. Dalam metode SAW, dilakukan proses normalisasi matriks keputusan (X) ke dalam skala yang dapat dibandingkan dengan semua rating alternatif yang ada [13].

2.2.3.2 Langkah-langkah Metode *Simple Additive Weighting* (SAW)

Adapun langkah-langkah metode *Simple Additive Weighting* (SAW) [3] adalah sebagai berikut :

1. Menentukan kriteria yang akan digunakan dalam pengambilan keputusan yaitu C_i
2. Menentukan rating kecocokan setiap alternatif pada setiap kriteria (x)
3. Membuat matriks keputusan berdasarkan kriteria (C_i) , lalu melakukan normalisasi matriks berdasarkan persamaan yang disesuaikan dengan jenis atribut (atribut *benefit* atau atribut *cost*) sehingga diperoleh ternormalisasi \bar{x} . Adapun rumus yang digunakan untuk normalisasi matriks seperti Persamaan 1.

Persamaan 1 :

$$rij = \begin{cases} \frac{xij}{\max(xij)} & \text{Jika } j \text{ adalah atribut keuntungan (benefit)} \\ \frac{\min(xij)}{xij} & \text{Jika } j \text{ adalah atribut biaya (cost)} \end{cases} \quad (1)$$

Keterangan :

- rij = Nilai rating kinerja ternormalisasi
- xij = Nilai atribut yang dimiliki dari setiap kriteria
- $\max(xij)$ = Nilai maksimum dari setiap kriteria
- $\min(xij)$ = Nilai minimum dari setiap kriteria
- benefit* = Jika nilai terbesar adalah nilai terbaik
- cost* = Jika nilai terkecil adalah nilai terbaik

4. Hasil akhir diperoleh dari penjumlahan dari perkalian matriks yang sudah ternormalisasi r dengan vektor bobot dipilih sebagai alternatif terbaik A_i sebagai solusi. Nilai rij adalah rating kinerja yang ternormalisasi dari alternatif A_i pada atribut $C_i = 1, 2, \dots, m$ dan $j = 1, 2, \dots, m$. Nilai preferensi untuk setiap alternatif (V_i) diberikan rumus pada Persamaan 2.

Persamaan 2 :

$$V_i = \sum_{j=1}^n W_j rij \quad (2)$$

Keterangan :

- V_i = Nilai akhir atau ranking untuk setiap alternatif
- W_j = Nilai bobot dari setiap kriteria yang telah ditentukan
- rij = Nilai rating kinerja yang ternormalisasi atau normalisasi matriks

Nilai V_i yang lebih besar menunjukkan bahwa alternatif A_i lebih dipilih

2.2.4 Konsep Dasar Web

2.2.4.1 Pengertian Web

Web adalah suatu aplikasi yang mengandung berbagai dokumen multimedia seperti teks, gambar, suara, animasi ataupun video di dalamnya. Web menggunakan

protokol HTTP (*Hypertext Transfer Protocol*) dan dapat diakses menggunakan perangkat lunak yang bernama browser [14].

2.2.4.2 Komponen Web

Berikut ini adalah komponen-komponen yang pada sebuah aplikasi web [14] yaitu:

1. Bahasa Pemrograman/*Scripting Language*
Terdiri dari dua kategori yaitu *Client side scripting* seperti HTML (*Hypertext Markup Language*), CSS (*Cascading Style Sheet*), serta JavaScript dan *Server side scripting* seperti ASP (*Active Server Pages*) dan PHP (*Hypertext Preprocessor*). Untuk *Client side scripting* dieksekusi oleh web browser langsung, sedangkan *Server side scripting* dieksekusi atau dilakukan oleh sebuah modul yang disebut *web engine*.
2. Web Editor/Kode Editor
Web Editor atau Kode Editor adalah sebuah program aplikasi yang dirancang untuk memfasilitasi pengetikkan perintah-perintah dokumen web, baik *Client side scripting* maupun *Server side scripting*. Web editor yang paling umum digunakan adalah Notepad++, Sublime Text, dan Visual Studio Code.
3. Web Browser
Web Browser merupakan sebuah program yang berperan dalam menampilkan berbagai dokumen web dalam format HTML. Web browser yang umum digunakan adalah Google Chrome, Microsoft Edge, dan Mozilla Firefox.
4. Web *Server*
Web *server* adalah sebuah program aplikasi yang bertugas sebagai tempat penyimpanan berbagai dokumen web.
5. *Database Server*
Database Server adalah sebuah program yang digunakan untuk menyimpan

data yang akan diproses di dalam halaman web.

6. *Image Editor*

Image Editor adalah sebuah program aplikasi yang memiliki fungsi untuk melakukan pengolahan gambar atau *image*.

2.2.5 Metode Pengembangan Aplikasi

2.2.5.1 *Waterfall*

Model *Waterfall* merupakan salah satu model SDLC yang sering digunakan dalam pengembangan sistem informasi atau perangkat lunak. Model ini mengadopsi pendekatan sistematis dan urutan langkah-langkah yang jelas. Proses pengembangan sistem dilakukan melalui serangkaian tahapan yang dimulai dari perencanaan hingga pengelolaan atau pemeliharaan, dan setiap tahapan dilakukan secara bertahap dan berurutan. Selain itu, model ini mengikuti prinsip bahwa tahapan berikutnya tidak dapat dimulai sebelum tahapan sebelumnya diselesaikan dengan baik, dan tidak memungkinkan adanya pengulangan tahap sebelumnya [15].

2.2.6 Perancangan Sistem

2.2.6.1 *Unified Modeling Language (UML)*





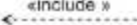
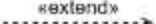
Unified Modeling Language (UML) merupakan istilah yang berkaitan dengan pemrograman berorientasi objek dan teknik penggambaran diagram yang digunakan untuk memodelkan proyek pengembangan sistem dari tahap analisis hingga desain. UML terdiri dari sejumlah notasi grafis yang terkait satu sama lain dan didukung oleh satu meta-model tunggal. UML membantu dalam deskripsi dan perancangan sistem perangkat lunak, terutama sistem yang dibangun menggunakan pendekatan pemrograman berorientasi objek [16].

2.2.6.2 *Use Case Diagram*


Use Case Diagram adalah sebuah diagram yang digunakan untuk menjelaskan fungsi-fungsi sistem pada tingkat konseptual. Diagram ini merupakan elemen kunci dalam berbagai teknik diagram UML. *Use Case Diagram* menggambarkan fungsi dasar sistem, termasuk tindakan-tindakan yang dapat dilakukan oleh pengguna serta respons sistem terhadap tindakan-tindakan tersebut. Diagram ini memberikan gambaran secara visual mengenai interaksi antara pengguna dan sistem yang

sedang dibangun [16]. Elemen-elemen dari *Use Case Diagram* [17] dijelaskan pada Tabel 2.2 berikut:

Tabel 2.2 Simbol – Simbol *Use Case Diagram*

Simbol	Keterangan
	<i>Actor/role</i> : adalah orang atau sistem lain yang berinteraksi dengan sistem yang ada saat ini.
	<i>Use Case</i> : merupakan bagian utama dari fungsionalitas sistem.
	<i>Subject Boundary</i> : berisikan nama dari sistem yang diletakkan di dalam atau di bagian atas <i>boundary</i> .
	<i>Association Relationship</i> : menghubungkan <i>Actor</i> dengan <i>Use Case</i> . Menunjukkan komunikasi satu arah jika menggunakan tanda panah, jika tidak maka menunjukkan komunikasi dua arah. Tanda * untuk keragaman dari asosiasi.
	<i>Include Relationship</i> : memasukkan satu <i>Use Case</i> dalam <i>Use Case</i> lainnya.
	<i>Extend Relationship</i> : memperluas <i>Use Case</i> untuk memasukkan perilaku opsional. Tanda panah akan mengarah dari <i>Use Case</i> tambahan ke <i>base Use Case</i> atau pusat.





Tabel 2.2 Simbol – Simbol *Use Case Diagram*

	<i>Generalization Relationship:</i> mewakili <i>Use Case</i> khusus untuk <i>Use Case</i> yang lebih umum.
---	---





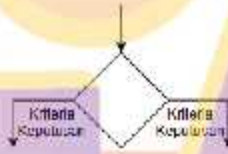
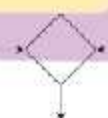
2.2.6.3 *Activity Diagram*

Activity Diagram adalah diagram yang digunakan untuk menggambarkan bagaimana setiap aliran kegiatan dimulai, keputusan yang mungkin terjadi, dan bagaimana aliran tersebut berakhir. Diagram ini memberikan representasi visual tentang urutan langkah-langkah atau aktivitas yang terjadi dalam suatu proses atau sistem [16]. Berikut beberapa elemen dari *Activity Diagram* [17] pada Tabel 2.3 :

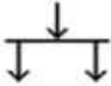
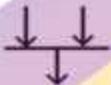
Tabel 2.3 Elemen-elemen *Activity Diagram*

Notasi	Simbol	Keterangan
<i>Action</i>		Perilaku yang sederhana dan tidak dapat diuraikan.
<i>Activity</i>		Digunakan untuk mewakili sekumpulan tindakan.
<i>Object Node</i>		Digunakan untuk mewakili sebuah objek yang terhubung ke sekumpulan arus objek.
<i>Control Flow</i>		Menunjukkan urutan eksekusi.

Tabel 2.3 Elemen-elemen *Activity Diagram*

<i>Object Flow</i>		Menunjukkan aliran suatu objek dari satu aktivitas (atau tindakan) ke aktivitas (atau tindakan) lain.
<i>Initial Node</i>		Menggambarkan awal dari serangkaian tindakan atau kegiatan.
<i>Final-activity Node</i>		Digunakan untuk menghentikan semua arus kontrol dan arus objek dalam suatu aktivitas (atau tindakan).
<i>Final-flow Node</i>		Digunakan untuk menghentikan aliran kontrol atau aliran objek tertentu.
<i>Decision Node</i>		Digunakan untuk mewakili kondisi pengujian untuk memastikan bahwa aliran kontrol atau aliran objek hanya turun satu jalur.
<i>Merge Node</i>		Digunakan untuk menyatukan kembali berbagai jalur keputusan yang dibuat menggunakan simpul keputusan.





Tabel 2.3 Elemen-elemen *Activity Diagram*

<i>Fork Node</i>		Digunakan untuk membagi perilaku menjadi seperangkat aktivitas yang paralel atau bersamaan dari aktivitas (atau tindakan).
<i>Join Node</i>		Digunakan untuk menyatukan kembali serangkaian arus aktivitas (atau tindakan) yang paralel atau bersamaan.
<i>Swimlane</i>		Digunakan untuk memecah diagram aktivitas menjadi baris dan kolom untuk menetapkan kegiatan individu (atau tindakan) kepada individu atau objek yang bertanggung jawab untuk melaksanakan aktivitas (atau tindakan).

2.2.6.4 *Class Diagram*

Class Diagram adalah sebuah diagram yang menggambarkan berbagai jenis objek dalam sistem serta berbagai macam hubungan statis yang terdapat di antara objek-objek yang ada. Tujuan utama dari *Class Diagram* adalah untuk menunjukkan relasi yang dimiliki oleh kelas dengan kelas lainnya [16]. Berikut beberapa elemen dari *Class Diagram* [17] pada Tabel 2.4 :





Tabel 2.4 Elemen-elemen *Class Diagram*

Elemen	Keterangan
	<i>Class</i> (kelas): Sebuah <i>class</i> merepresentasikan orang, tempat, atau berbagai hal yang diperlukan oleh sistem untuk menangkap dan menyimpan informasi.
	<i>Attribute</i> (atribut): <i>Attribute</i> adalah properti yang menggambarkan keadaan dari sebuah objek.
	<i>Operation</i> (operasi): <i>Operation</i> merupakan fungsi atau tindakan yang dapat dilakukan oleh kelas.
	<i>Association</i> (asosiasi): <i>Association</i> adalah elemen yang mewakili hubungan antara banyak kelas atau kelas dan dirinya sendiri.

2.2.6.5 *Sequence Diagram*

Sequence Diagram adalah sebuah diagram yang menunjukkan urutan pesan eksplisit yang diteruskan antara suatu objek dengan objek yang lainnya dalam suatu interaksi yang ditentukan [16]. Berikut beberapa elemen dari *Sequence Diagram* [17] pada Tabel 2.5 :

Tabel 2.5 Elemen-elemen *Sequence Diagram*

Notasi	Keterangan
	<p><i>Actor</i>: adalah orang atau sistem yang mendapatkan manfaat dari dan berada di luar sistem.</p>
	<p><i>Object</i>: adalah notasi yang berpartisipasi secara berurutan dengan mengirim atau menerima pesan.</p>
	<p><i>Lifeline</i>: adalah notasi yang menampilkan masa hidup suatu objek selama dalam urutan proses.</p>
	<p><i>Object Destruction</i>: adalah notasi yang menampilkan sebuah objek akan menghilang.</p>
	<p><i>Execution Occurrence</i>: adalah notasi yang menampilkan saat suatu objek mengirim atau menerima pesan.</p>
	<p><i>Message</i>: adalah notasi yang berfungsi untuk menyampaikan informasi dari satu objek ke objek lainnya.</p>

2.2.7 Framework Laravel

Laravel adalah sebuah *framework* PHP yang dikembangkan oleh Taylor Otwell dengan konsep Model, View, dan Controller (MVC). Tujuan utama Laravel dikembangkan adalah meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan pemeliharaan. Selain itu, Laravel juga memberikan pengalaman pengembang yang lebih baik dengan menyediakan sintaks yang ekspresif, jelas, dan efisien, sehingga menghemat waktu dan usaha dalam proses pengembangan aplikasi [18].



Gambar 2.1 Logo Laravel

2.2.7.1 Kelebihan Laravel

Berikut ini merupakan beberapa kelebihan Laravel [18] :

- a. Mudah dan Dokumentasi Lengkap
Jika sudah memiliki dasar pemahaman PHP yang baik maka akan mudah memahami struktur Laravel. Selain itu dokumentasi Laravel rapi dan mudah dibaca.
- b. *Open Source*
Laravel merupakan *framework* yang dapat digunakan secara gratis dan bebas.
- c. Arsitektur MVC
Penggunaan Arsitektur MVC memungkinkan untuk dapat membuat struktur kode yang lebih rapi, dimana arsitektur tersebut memisahkan antara logika dan

view.

d. *Blade Template*

Terdapat fitur di Laravel yang bernama *Blade*, yang dapat digunakan untuk membagi template yang kita miliki menjadi beberapa bagian untuk dipetakan sehingga lebih mudah untuk mengaturnya.

e. Memiliki Fitur *Migration*

Migration salah satu keunggulan dari *framework* Laravel dimana kita dapat mempertahankan struktur *database* yang kita miliki tanpa harus membuat ulang. *Migration* juga mampu mengembalikan beberapa perubahan terakhir yang dilakukan pada *database*.

f. Keamanan

Laravel menawarkan keamanan yang bagus untuk aplikasi yang dibuat, seperti contohnya pada *Object Relational Mapping* (ORM) Laravel menggunakan PDO yang mampu mencegah *SQL Injection*. itu, Laravel juga memiliki proteksi *CSRF Token*.

g. Komunitas yang besar

Dukungan komunitas yang besar pada Laravel dapat membantu belajar dan mencari solusi jika terdapat permasalahan yang mungkin terjadi.

h. Hemat Waktu

Dengan abstraksi-abstraksi yang dilakukan Laravel, kita bisa lebih fokus memikirkan logika bisnis dari aplikasi yang dibuat sehingga dapat menghemat waktu.

2.2.8 Pengujian Sistem

2.2.8.1 *Blackbox Testing*

Blackbox Testing adalah suatu metode pengujian yang digunakan untuk memverifikasi apakah fungsi-fungsi masukan dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang telah ditentukan. Pengujian dilakukan dengan menggunakan input yang diberikan dan memeriksa output yang dihasilkan, tanpa

memperhatikan atau mengetahui bagaimana proses internal perangkat lunak tersebut bekerja secara detail [6].

