

BAB II LANDASAN TEORI

2.1. Tinjauan Pustaka

Pada penelitian ini, penulis melakukan tinjauan pustaka pada penelitian-penelitian terdahulu sebagai referensi penulis untuk menunjang dan mempermudah dalam penelitian ini. Di bawah ini merupakan tinjauan pustaka yang telah diteliti sebelumnya dapat dilihat pada Tabel 2.1. berikut:

Tabel 2.1. Tinjauan Pustaka

No	Detail Jurnal	
1	Judul	<i>E-Tourism</i> Provinsi Bali Berbasis Web dengan <i>Framework Laravel</i>
	Tahun Penerbit	2020
	Penulis	Rifky Lana Rahardian, Ni Luh Gede Pivin Suwirmayanti
	Metode Pengembangan Sistem	Metode <i>Waterfall</i>
	Pengujian Sistem	<i>Blackbox</i>
	Hasil	Hasil dari penelitian tersebut adalah Informasi yang terdapat pada sistem sudah dibuat sesederhana mungkin dan menggunakan Bahasa Indonesia dikarenakan sistem ini berfokus pada wisatawan domestik, serta memberikan informasi yang lengkap tidak hanya berupa gambar atau text melainkan pencarian lokasi atau mapping untuk memudahkan wisatawan.

Tabel 2.1. Tinjauan Pustaka (Lanjutan)

No	Detail Jurnal	
2	Judul	Rancang Bangun Sistem Informasi Pariwisata Tasikmalaya Berbasis Web
	Tahun Penerbit	2021
	Penulis	Lutphi Pasha Nugraha, Eka Wahyu Hidayat, Aldy Putra Aldya
	Metode Pengembangan Sistem	Metode <i>Extreme Programming</i> (XP)
	Pengujian Sistem	<i>Usability</i>
	Hasil	Hasil dari penelitian tersebut adalah mempermudah masyarakat luas dalam mengetahui informasi obyek wisata yang ada di Kota dan Kabupaten Tasikmalaya tanpa harus mengunjungi secara langsung ke lokasi
3	Judul	Pengembangan Sistem Informasi Sewa Mobil Dan Paket Wisata Berbasis Web Menggunakan Teknologi <i>Framework Laravel</i>
	Tahun Penerbit	2020
	Penulis	Ronaldo Grasberg Wijaya, Widhy Hayuhardhika Nugraha Putra, Welly Purnomo

Tabel 2.1. Tinjauan Pustaka (Lanjutan)

No	Detail Jurnal	
	Metode Pengembangan Sistem	Metode <i>Software Development Life Cycle</i> (SDLC) dengan model <i>Waterfal</i>
	Pengujian Sistem	<i>Blackbox Testing</i> dan <i>Usability</i>
	Hasil	Hasil dari penelitian tersebut adalah sistem informasi yang membantu perusahaan untuk mendata daftar mobil yang tersedia, jenis paket wisata, informasi kuliner dan informasi biaya.
4	Judul	Sistem Informasi Pemesananan Tiket Tour And Travel Berbasis <i>Website</i> Menggunakan Metode <i>Waterfall</i> Pada <i>Camelia Travel</i>
	Tahun Penerbit	2022
	Penulis	Muhtajuddin Danny, Daimin
	Metode Pengembangan Sistem	Metode <i>Waterfall</i>
	Pengujian Sistem	-
	Hasil	Hasil dari penelitian tersebut adalah sistem informasi yang dapat memberikan kemudahan kepada konsumen dalam melakukan pemesanan tiket perjalanan dan memudahkan <i>Camelia Travel</i> Dalam melakukan pengelolaan data reservasi atau pemesanan paket wisata.

Tabel 2.1. Tinjauan Pustaka (Lanjutan)

No	Detail Jurnal	
5	Judul	Sistem Informasi Pariwisata Daerah Kabupaten Minahasa Selatan Berbasis Web GIS
	Tahun Penerbit	2023
	Penulis	Graciano Emmanuel Martin Pongantung, Aditya Lapu Kalua, Deiby Tineke Salaki
	Metode Pengembangan Sistem	Metode <i>Rapid Application Development (RAD)</i>
	Pengujian Sistem	-
	Hasil	Hasil dari penelitian tersebut adalah sistem informasi yang dapat memberikan memberikan informasi kepada wisatawan mengenai pariwisata di Minahasa Selatan, serta terdapat fitur peta dari rute perjalanan ke tempat wisata tersebut.

2.2. Sistem Informasi

Sistem informasi adalah suatu metode yang digunakan untuk secara terstruktur mengumpulkan, memasukkan, mengolah, menyimpan, mengelola, mengawasi, dan melaporkan data dan informasi, dengan tujuan membantu perusahaan atau organisasi mencapai tujuan yang telah ditetapkan (Nova, et al., 2022). Sistem informasi dirancang secara khusus dengan tujuan memfasilitasi pengguna dalam menjalankan aktivitas mereka. Komponen-komponennya terdiri dari elemen-elemen berbasis komputer seperti perangkat keras, perangkat lunak,

basis data, telekomunikasi, dan peran manusia (Wijaya, et al., 2020). Jadi dapat disimpulkan, sistem informasi bertujuan untuk mempermudah pengguna dalam melaksanakan aktivitas dari berbagai layanan informasi.

2.3. Wisata Alam

Istilah wisata dalam bahasa Inggris disebut *tour*, yang berasal dari kata *torah* dalam bahasa Ibrani yang berarti belajar, *tornus* dalam bahasa Latin yang mengacu pada alat untuk membuat lingkaran, dan istilah *tour* dalam bahasa Perancis kuno yang berarti melakukan perjalanan mengelilingi suatu sirkuit. Umumnya, orang sering mengaitkan kata wisata dengan rekreasi (Nurhuda & Putra, 2022). pariwisata adalah sektor yang sedang berkembang dengan cepat. Bahkan, sektor wisata memiliki potensi besar dalam kontribusinya terhadap pertumbuhan ekonomi suatu wilayah karena dapat menarik wisatawan dari dalam dan luar negeri untuk mengunjungi daerah tersebut. Ini tentu saja memberikan keuntungan yang signifikan dan menggerakkan pertumbuhan pendapatan daerah serta penduduknya. (Zevita & , 2023).

Sedangkan menurut (Nurhuda & Putra, 2022), Wisata alam adalah aktifitas rekreasi dan pariwisata yang memanfaatkan sumber daya alam untuk menikmati keindahan alam, baik yang masih dalam kondisi alami maupun yang telah diatur secara budidaya, dengan tujuan untuk menarik para wisatawan ke lokasi tersebut. Wisata alam sering digunakan sebagai cara untuk mengimbangi kehidupan yang sibuk dan suasana ramai di kota setelah melakukan aktivitas yang padat. Dalam menjalankan aktivitas wisata alam, penting untuk menjaga kelestarian area yang masih alami, merasakan dampak ekonomi yang positif, dan menjaga integritas budaya masyarakat lokal. Jadi dapat disimpulkan, wisata alam merupakan kegiatan

seseorang untuk melakukan perjalanan yang bertujuan dalam memanfaatkan kekayaan sumber daya alam untuk menikmati keindahan alam.

2.4. Website

Website adalah salah satu layanan utama yang banyak dimanfaatkan melalui internet. Ini juga dapat dianggap sebagai sebuah sistem yang mengandung informasi dalam berbagai format seperti teks, gambar, suara, dan sejenisnya. Semua informasi ini disimpan di *server web* di internet dan ditampilkan dalam bentuk *hypertext* (Danny & , 2022). Halaman *web* ini dapat diakses saat ada koneksi internet, di mana halaman tersebut berisi informasi yang telah disusun berdasarkan jenis konten tertentu. Ada beberapa keuntungan utama dalam menggunakan *web*, yang menjadi dasar mengapa *web* digunakan (Wijaya, et al., 2020). Jadi, kesimpulannya adalah *website* merupakan suatu layanan yang bisa menampilkan layanan informasi dan pemasaran suatu produk yang dapat diakses menggunakan *Linux, Safari, Windows*, dan sejenisnya.

2.5. Laravel

Laravel merupakan sebuah *Framework* yang berfungsi untuk memudahkan proses pengembangan perangkat lunak berbasis *web*. Dalam pengembangan menggunakan *Laravel*, digunakan konsep *Model-View-Controller* (MVC) dan dilengkapi dengan alat baris perintah yang disebut *Artisan*. Konsep MVC ini adalah inovasi yang sangat membantu dalam menyederhanakan proses pengembangan *Website*, dengan *Artisan* sebagai fitur bantuannya. Dengan menggunakan *Laravel*, penulisan kode program menjadi lebih mudah dibandingkan dengan tidak menggunakan *Laravel*, dan juga memudahkan dalam pengembangan lanjutan (Wijaya, et al., 2020). Jadi, kesimpulannya adalah *Laravel* merupakan sebuah

pengembangan *software* yang memudahkan pengembang suatu *Website* dikarenakan menggunakan *tools* Artisan dalam mengembangkan suatu *project* berbasis *Website*.

2.6. *MySQL*

MySQL merupakan sebuah perangkat lunak *server* basis data yang memiliki kemampuan untuk mengolah data secara cepat dalam hal penerimaan dan pengiriman, serta mendukung banyak pengguna secara bersamaan. *MySQL* menggunakan perintah-perintah standar dalam bahasa *SQL* (*Structured Query Language*) dan dapat digunakan baik sebagai *client* maupun sebagai *server* (Danny & , 2022). Sedangkan menurut (Wijaya, et al., 2020), *MySQL* merupakan salah satu sistem pengelolaan basis data, *Database Management System* (DBMS) yang digunakan untuk mengelola dan mengatur data. Oleh karena itu, kesimpulannya adalah bahwa *MySQL* adalah perangkat lunak basis data yang menggunakan perintah-perintah dalam bahasa *SQL* untuk mengelola dan mengatur data, sehingga memberikan kemudahan penggunaan dengan kemampuan berfungsi sebagai *client* maupun *server*.

2.7. *Visual Studio Code*

Visual Studio Code adalah sebuah editor teks yang telah dikembangkan oleh *Microsoft* dan dapat digunakan pada berbagai sistem operasi seperti *Windows*, *Linux*, dan *MacOS*. Editor teks ini menyediakan fitur-fitur yang sangat berguna, termasuk kemampuan *debugging*, integrasi *GIT* yang kuat, otomatisasi kode yang cerdas, penggunaan *snippet*, serta kemampuan untuk membangun kembali kode. *Visual Studio Code* juga sangat fleksibel, memungkinkan pengguna untuk menyesuaikan tema, pintasan *keyboard*, dan preferensi sesuai dengan kebutuhan

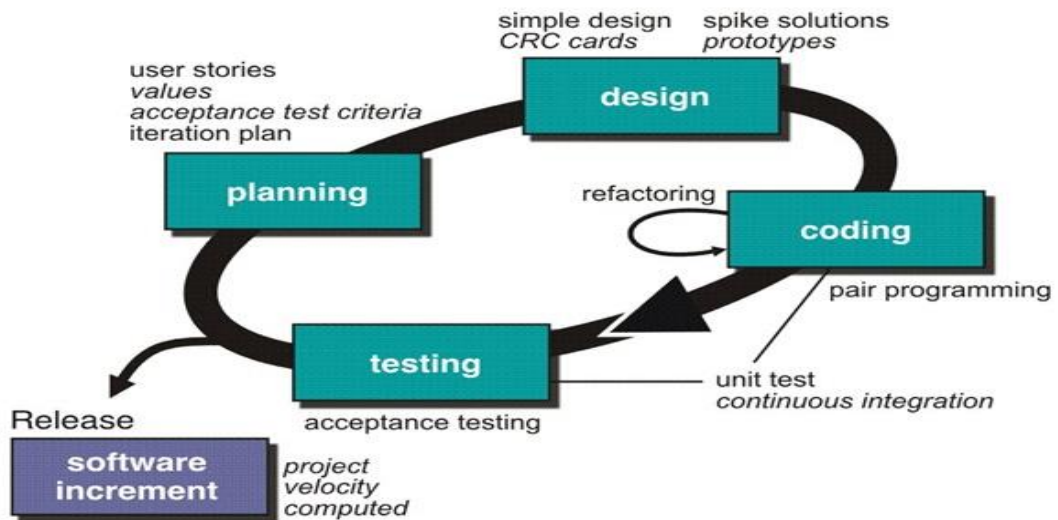
mereka (Lubis & , 2023). Jadi, kesimpulannya *visual studio code* merupakan sebuah kode editor yang dirancang oleh Microsoft untuk berjalan pada sistem operasi *Windows*, *Linux*, dan *MacOS*. Ini memiliki dukungan terintegrasi untuk *debugging*, pengelolaan kontrol versi dengan *GIT*, dan integrasi dengan layanan *GitHub* dalam proses pengembangannya.

2.8. Metode Pengembangan Sistem

2.8.1. *Extreme Programming*

Extreme Programming (XP) adalah metode yang umum digunakan dalam rekayasa perangkat lunak, pengembangan aplikasi, dan sistem informasi. Dalam pembuatan situs *web*, XP digunakan karena memfasilitasi fleksibilitas untuk mengubah atau menambahkan kebutuhan sistem dan alur kerja selama proses pengembangan. Dalam pengembangannya, XP menerapkan pendekatan pengembangan perangkat lunak yang berorientasi pada proyek-proyek berukuran kecil yang memungkinkan perubahan yang cepat dan fleksibel (Hirawan & Sama, 2022).

Sedangkan Menurut (Wongso, 2022), *Extreme Programming* (XP), adalah suatu pendekatan dalam pengembangan perangkat lunak yang bertujuan untuk menganalisis dan menyederhanakan berbagai tahapan pengembangan agar menjadi lebih mudah digunakan dan praktis. XP tidak hanya menitikberatkan pada proses pengkodean, tetapi juga melibatkan seluruh aspek dalam pengembangan perangkat lunak. XP terdiri dari empat tahapan utama, yaitu perencanaan, perancangan, pengkodean, dan pengujian, yang dapat dilihat pada Gambar 2.1.



Gambar 2.1. Tahapan *Extreme Programming* (XP)

1. Perencanaan (*Planning*)

Pada tahap perencanaan, langkah pertama adalah mengidentifikasi masalah yang dan menganalisis kebutuhan yang cocok. Data yang perlu dikumpulkan oleh penulis untuk merancang sebuah *Website* melalui metode seperti observasi di lokasi penelitian, yang memberikan informasi penting tentang kebutuhan yang diperlukan oleh objek penelitian. Wawancara juga dilakukan oleh penulis kepada objek penelitian untuk memahami masalah yang sering muncul dalam pengelolaannya.

2. Perancangan (*Design*)

Dalam proses desain, digunakan UML untuk memodelkan sistem, arsitektur, dan pemodelan basis data.

3. Pengkodean (*Coding*)

Model yang telah disusun diimplementasikan ke dalam bentuk antarmuka pengguna menggunakan bahasa pemrograman.

4. Pengujian (*Testing*)

Pada tahap pengujian sistem yang telah selesai dilakukan dengan tujuan untuk memastikan bahwa sistem berfungsi tanpa kesalahan atau error dalam fungsionalitasnya.

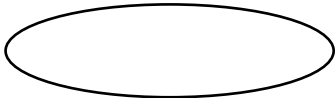
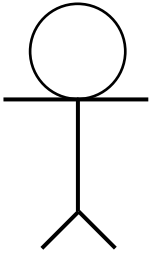


2.9. *Unified Modelling Language (UML)*

UML adalah singkatan dari *Unified Modeling Language*, yang merupakan standar untuk menggambarkan dan mewakili perangkat lunak berorientasi objek serta sistem bisnis dalam bentuk diagram dan metode yang telah disepakati (Mulyani, 2016). Di bawah ini merupakan pengertian dari tiga diagram, sebagai berikut:


2.9.1. *Use Case Diagram*

Use case diagram adalah suatu *diagram* yang digunakan untuk mengilustrasikan dan menggambarkan aktor-aktor, kasus pengguna (*use cases*), serta ketergantungan-ketergantungan dalam suatu proyek. Tujuan dari *diagram* ini adalah untuk menjelaskan konsep hubungan antara sistem dengan lingkungan (Mulyani, 2016). Tabel 2.2 adalah simbol yang umum digunakan dalam *use case diagram*.

Tabel 2.2. Simbol *Use Case Diagram*

Simbol	Deskripsi
<p><i>Use Case:</i></p> 	<p>Fungsionalitas yang ditawarkan oleh sistem dianggap sebagai unit-unit yang berkomunikasi melalui pertukaran pesan antara unit atau aktor yang berbeda.</p>
<p>Aktor:</p> 	<p>Aktor dalam konteks sistem informasi yang sedang dikembangkan adalah entitas eksternal yang terlibat dalam interaksi dengan sistem tersebut. Meskipun simbol aktor sering digambarkan sebagai manusia, sebenarnya bisa mewakili berbagai entitas selain manusia.</p>
<p>Asosiasi:</p> 	<p>Ini adalah proses di mana informasi pertukaran terjadi antara aktor dan use case yang terlibat dalam suatu skenario use case tertentu. Dalam konteks ini, use case dapat berkomunikasi dengan aktor-aktor yang terlibat dalam proses tersebut.</p>
<p><i>Extend:</i> <<extend>></p> 	<p>Merupakan relasi dalam diagram use case yang menunjukkan bahwa suatu use case tambahan dapat berdiri sendiri dan tidak bergantung pada use case tambahan.</p>

Tabel 2.2. Simbol *Use Case Diagram* (Lanjutan)



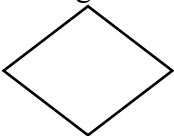
<p><i>Include:</i> <<include>></p> 	<p>Ini adalah hubungan dalam diagram use case yang menunjukkan bahwa suatu use case tambahan membutuhkan atau bergantung pada use case yang terkait dengannya.</p>
---	--

Sumber : (Mulyani, 2016)


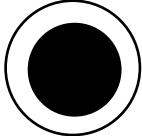
2.9.2. Activity Diagram

Activity Diagram adalah jenis diagram *UML* yang digunakan untuk mengilustrasikan aliran aktivitas dari suatu proses atau aktivitas tertentu (Mulyani, 2016). Dibawah ini merupakan simbol yang umum digunakan dalam *activity diagram*.

Tabel 2.3. Simbol *Activity Diagram*

Simbol	Deskripsi
<p>Mulai/<i>Start</i>:</p> 	<p>Ini adalah simbol awal yang digunakan dalam diagram aktivitas.</p>
<p>Aktivitas :</p> 	<p>Merupakan sebuah aktivitas yang akan dilakukan oleh sebuah system, yang mana dapat diawali dengan sebuah kata.</p>
<p>Asosiasi Percabangan :</p> 	<p>Ini adalah simbol yang mengindikasikan adanya berbagai pilihan yang dapat diambil.</p>

Tabel 2.3. Simbol *Activity Diagram* (Lanjutan)

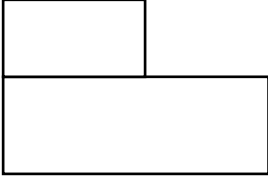
Asosiasi Penggabungan : 	Ini adalah simbol yang digunakan dalam diagram aktivitas untuk menggabungkan beberapa aktivitas menjadi satu.
Akhir / <i>Finish</i> : 	Ini adalah simbol akhir yang digunakan dalam diagram aktivitas.

Sumber : (Mulyani, 2016)

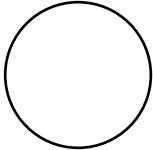

2.9.3. *Class Diagram*

Class diagram adalah suatu *diagram* yang digunakan untuk menggambarkan kelas-kelas, komponen-komponen kelas, serta relasi yang ada di antara kelas-kelas tersebut. *Class diagram* ini menguraikan jenis-jenis objek yang ada dalam sistem dan berbagai jenis hubungan statis yang terdapat di antara objek-objek tersebut (Mulyani, 2016). Dibawah ini merupakan simbol dari *class diagram*.

Tabel 2.4 Simbol *Class Diagram*

Simbol	Deskripsi
Package: 	Ini adalah sekumpulan satu atau lebih kelas yang digabungkan menjadi satu kesatuan.

Tabel 2.4. Simbol *Class Diagram* (Lanjutan)

<p>Operasi:</p> <table border="1" data-bbox="496 342 691 510"> <tr><td>Nama_kelas</td></tr> <tr><td>+ Attribute 1</td></tr> <tr><td>+ Attribute 2</td></tr> <tr><td>+ Operation ()</td></tr> </table>	Nama_kelas	+ Attribute 1	+ Attribute 2	+ Operation ()	<p>Ini adalah sebuah kelas yang ada dalam struktur sistem dari diagram kelas.</p>
Nama_kelas					
+ Attribute 1					
+ Attribute 2					
+ Operation ()					
<p>Antarmuka / <i>Interface</i>:</p> 	<p>Ini adalah simbol dalam diagram kelas yang mewakili konsep yang sama dengan antarmuka.</p>				
<p>Agregasi :</p> 	<p>Ini adalah hubungan antara kelas-kelas yang menunjukkan hubungan semua-bagian (<i>whole-part</i>) di antara mereka.</p>				

Sumber : (Mulyani, 2016)

2.10. Pengujian *Black Box*

Pengujian ini mirip dengan memperlakukan perangkat lunak sebagai suatu "Kotak Hitam" atau (*Black Box*). Artinya, pengujian ini tidak memerlukan pengetahuan mendalam tentang bagaimana sistem bekerja secara internal. Meskipun pengujinya perlu memahami struktur umum sistem, mereka tidak harus terlibat dalam rincian teknis kode program. *Black Box testing* memiliki sejumlah kelebihan, seperti efesiennya dalam menguji segmen kode yang besar, kemudahan dalam pemahaman bagi penguji yang mendasarkan pengujian pada perspektif pengguna, dan pengembang yang memiliki sudut pandang berbeda. Metode ini juga dianggap sebagai salah satu cara paling cepat dalam menguji pengembangan sistem. *Black Box testing* juga dikenal dengan sejumlah istilah lain, termasuk *opaque*

testing, fungsional testing, close box testing, dan behavioural testing (Wijaya, et al., 2020).

2.11. Pengujian *User Acceptance Testing* (UAT)

Pengujian *User Acceptance Testing* (UAT) merupakan pengujian tahap terakhir dalam pengujian perangkat lunak yang bertujuan untuk memverifikasi bahwa sistem yang telah dibangun sesuai dengan kebutuhan pengguna (Maharesi & Maharesi, 2023).

Sedangkan Menurut, (Bastari, et al., 2022) *User Acceptance Testing* (UAT) adalah proses pengujian sistem yang telah dikembangkan, di mana pengujinya adalah pengguna akhir. Hasil dari UAT adalah dokumen yang menjadi bukti bahwa pengguna telah menerima aplikasi yang dikembangkan dan merasa bahwa kebutuhan mereka telah terpenuhi berdasarkan hasil pengujian tersebut. Selain itu, UAT juga digunakan untuk mendapatkan masukan dari pengguna atau responden terkait sistem yang telah dibangun.

Dalam pengembangan sistem informasi, *User Acceptance Testing* (UAT) merupakan langkah kritis untuk mengonfirmasi persyaratan pengguna, mengidentifikasi potensi kesalahan, mengurangi risiko, dan memastikan kepuasan pengguna akhir. Dengan melibatkan pengguna dalam proses pengujian, UAT berperan penting dalam memastikan bahwa sistem memenuhi kebutuhan dan harapan pengguna, meningkatkan kesuksesan implementasi, serta memberikan nilai tambah yang signifikan bagi organisasi atau entitas yang menggunakan sistem tersebut (Anjasmara & Sumitro, 2023).

2.12. Skala *Likert*

Skala *Likert* adalah jenis skala ordinal yang terdiri dari pernyataan atau pertanyaan, diikuti oleh pilihan respons yang telah terstruktur, seringkali berkisar dari "Sangat Setuju" hingga "Sangat Tidak Setuju." Skala ini digunakan secara luas untuk mengukur sikap, pandangan, opini, dan preferensi. Dalam skala *Likert*, responden diberikan kisaran pilihan yang mencakup kesetujuan penuh hingga ketidaksetujuan penuh, dengan opsi netral di tengah jika responden tidak memiliki preferensi yang kuat (Krismayanti & Sutabri, 2023).

Skala *Likert* umumnya digunakan untuk mengukur atau menyusun skala sikap, dan selain itu, dapat digunakan untuk mengukur persepsi seseorang atau kelompok terhadap fenomena sosial yang ingin diteliti. Skala *Likert* sangat populer, terutama di kalangan mahasiswa yang menggunakannya dalam penulisan tugas akhir atau skripsi. Dengan mengetahui jumlah variabel yang akan diteliti oleh mahasiswa atau peneliti, penelitian dapat lebih mudah dilakukan jika variabel yang akan diteliti memiliki aspek, ciri, bentuk, dimensi, dan sebagainya. Skala *Likert* biasanya memiliki lima alternatif jawaban yang kemudian dimodifikasi menjadi empat alternatif jawaban. Selain itu, jawaban akan masuk ke dalam pernyataan atau pertanyaan yang bersifat positif (*favorable*) atau negatif (*unfavorable*) (Anasti, et al., 2022).