

BAB II TINJAUAN PUSTAKA

1.1 Tinjauan Studi

Dalam penelitian ini akan digunakan lima tinjauan pustaka yang nantinya dapat mendukung penelitian, berikut ini merupakan tinjauan pustaka yang diambil yaitu :

Tabel 1.1 Tinjauan Studi

No	Peneliti	Judul	Metode	Hasil
1	Prasetyo dkk (2019)	Implementasi Metode Item-Based Collaborative Filtering dalam Pemberian Rekomendasi Calon Pembeli Aksesoris Smartphone	Penelitian ini menggunakan metode Item-Based Collaborative Filtering, metode ini mencari similarity/kesamaan item dengan item lainnya.	Hasil dari penelitian menunjukkan kecilnya nilai rata-rata MAE 0,572039 namun untuk proses eksekusi, waktu yang dibutuhkan cukup lama yaitu 6,4 detik
2	Marthasari dkk (2015)	Sistem Rekomendasi Penyewaan Perlengkapan Pesta Menggunakan Collaborative Filtering Dan Penggalian Aturan Asosiasi	collaborative filtering	Pada penelitian ini, dibuat sebuah website persewaan alat-alat pesta dengan menerapkan sistem rekomendasi. Sistem rekomendasi dibangun menggunakan aturan-aturan yang dihasilkan oleh Algoritma Apriori. Untuk dapat sedangkan nilai confidence digunakan untuk menentukan N-teratas barang untuk direkomendasikan.

Tabel 2.1 Tinjauan Studi (Lanjutan)

No	Peneliti	Judul	Metode	Hasil
3	Fathoni dkk (2016)	Penerapan Metode Item Based Collaborative Filtering pada Sistem Electronic Commerce Berbasis Website	Maka dari itu, penelitian ini mencoba untuk menerapkan metode item-based collaborative filtering untuk menghasilkan rekomendasi produk yang dalam hal ini adalah buku.	Pendekatan item based collaborative filtering menjadikan item-item yang telah di-rating oleh pengguna menjadi dasar perhitungan. Algoritma ini melakukan perhitungan bagaimana kemiripan item-item yang telah di-rating dengan item-item lain dan selanjutnya dipilih sekelompok item yang mempunyai kemiripan dengan item yang sudah di-rating
4	Arief dkk (2012)	Rancang Bangun Sistem Rekomendasi Pariwisata Mobile dengan Menggunakan Metode Collaborative Filtering dan Location Based Filtering	metode collaborative filtering dan location-based filtering	asil penelitian ini berupa sebuah aplikasi sistem rekomendasi pariwisata berbasis mobile webmenggunakan platformjQuery Mobile, HTML 5, JavaScript, Ajax, PHP dan MySQL
5	Setiawan dkk (2019)	Implementasi Item Based Collaborative Filtering Dalam Pemberian Rekomendasi Agenda Wisata Berbasis Android	Penelitian ini menggunakan metode item based collaborative filtering dimana sistem merekomendasikan pariwisata berdasarkan item-item yang disukai oleh user	Penelitian ini menghasilkan sistem informasi rekomendasi Pariwisata Bengkulu berbasis Android yang di uji menggunakan metode pengujian blackbox dengan hasil 100% berhasil dan memperoleh nilai user experience sangat baik yaitu sebesar 90,752% pada pengujian usability testing.

Tabel 2.1 Tinjauan Studi (Lanjutan)

No	Peneliti	Judul	Metode	Hasil
6	Ahmad, dkk (2019)	Penerapan Algoritma Rekomendasi Pada Aplikasi Rumah Madu Untuk Perhitungan Akuntansi Sederhana Dan Marketing Digital	<i>Collaborative filtering (slope one)</i>	Penelitian ini diharapkan dapat membantu dalam mencari dan memberikan informasi yang berkaitan dengan produk-produk herbal dan memberikan alternatif media pemasaran UMKM herbal dalam penjualan khususnya digital marketing

Berdasarkan uraian kelima jurnal tersebut dapat dilihat beberapa perbedaan pada penelitian yang akan dibangun yaitu:

1. Penelitian dilakukan pada penyewaan mobil
2. Penerapan *framework codeigniter* untuk membangun sistem
3. Cakupan wilayah pada Bandar Lampung

1.2 Penyewaan

Menurut Kurniawan dkk (2018) Sewa-menyewa merupakan salah satu perjanjian yang terdiri pihak penyewa dan pihak yang menyewakan. Sewa– menyewa adalah hubungan hukum antara dua belah pihak didalam hubungan perjanjian masing–masing mempunyai hak dan kewajiban baik peraturan hukum tertulis maupun dalam bentuk kesepakatan.

Menurut Rosiska dan Nopiana (2017) mengungkapkan bahwa sewa menyewa adalah suatu persetujuan, dengan pihak yang satu mengikatkan dirinya untuk memberikan kenikmatan suatu

barang kepada pihak lain selama waktu tertentu, dengan pembayaran suatu harga yang disanggupi oleh pihak terakhir.

1.3 Item-Based Collaborative Filtering

Metode *collaborative filtering* sendiri dibagi lagi menjadi dua, yaitu user based dan item based. Jika *user based* menghitung kesamaan di antara pengguna sebagai parameter untuk menghasilkan rekomendasi.

Maka *item based* akan menghitung kesamaan diantara *item*, dilihat dari rating yang diberikan pengguna untuk item tersebut. Untuk membuat sistem rekomendasi menggunakan metode item *Based Collaborative Filtering*, ada dua langkah yang harus dilakukan, yaitu seperti berikut.

1.3.1 Menghitung *Similarity*

Rumus yang biasa dipakai adalah cosine similarity measure, yang menghitung kemiripan antara dua item dari sudut kosinus yang terbuat diantara kedua item tersebut. Sehingga apabila nilai cosinus-nya adalah 1, maka item tersebut benar-benar identik. Sedangkan jika nilainya 0, maka kedua item tersebut sama sekali tidak mempunyai kemiripan.

$$sim(i,j) = \frac{\sum_{u \in U} (r_{u,i} - r_u)(r_{u,j} - r_u)}{\sqrt{\sum_{u \in U} (r_{u,i} - r_u)^2} \sqrt{\sum_{u \in U} (r_{u,j} - r_u)^2}} \quad (1)$$

$sim(i,j)$ = nilai kemiripan antara item kendaraan i dan item kendaraan j

$\sum_{u \in U}$ = himpunan user u yang memberikan rate pada item kendaraan i
dan item kendaraan j

$r_{u,i}$ = rating user u pada produk i

$r_{u,j}$ = rating user u pada produk j

ru = rata-rata rating user u

Jika kemiripan sudah dihitung, maka kita akan mengurutkan item-item berdasarkan similarity mereka, item-item yang mempunyai similarity besar akan berada pada urutan atas, dan sebaliknya.

1.3.2 Menghitung Prediksi *Rating*

Perhitungan rating dilakukan dengan cara membandingkan rating yang pernah diberikan pengguna pada suatu item dengan kemiripan antara item tersebut dengan item lainnya.

Metode yang dipakai adalah menggunakan metode *weighted sum*

$$P(u, j) = \frac{\sum_{i \in I} (R_{u, i} * S_{i, j})}{\sum_{i \in I} |S_{i, j}|} \quad (2)$$

$P(u, j)$ = prediksi untuk user u pada item kendaraan j

$i \in I$ = himpunan item kendaraan yang mirip dengan item kendaraan j

$R_{u, i}$ = rate user u pada item kendaraan i

$S_{i, j}$ = nilai kemiripan antara item kendaraan i dan item kendaraan j

Jika prediksi rating sudah dihitung, maka rekomendasi item pun dapat dihasilkan.

1.4 *CodeIgniter*

CodeIgniter menurut Raharjo (2015) merupakan *framework* untuk bahasa pemrograman PHP, yang dibuat Rick Ellis pada tahun 2006. *CodeIgniter* memiliki banyak fitur yang membantu para pengembang PHP untuk dapat membuat aplikasi secara mudah dan cepat serta memiliki sifat yang fleksibel dapat dikembangkan dalam perangkat *web*, dekstop maupun *mobile*.

CodeIgniter menjadi sebuah framework PHP dengan model *MVC (Model, View, Controller)* untuk membangun website dinamis dengan menggunakan PHP yang dapat mempercepat pengembang untuk membuat sebuah aplikasi web. Selain ringan dan cepat, *CodeIgniter* juga memiliki dokumentasi yang super lengkap disertai dengan contoh implementasi kodenya. Dokumentasi yang lengkap inilah yang menjadi salah satu alasan kuat mengapa banyak orang memilih *CodeIgniter* sebagai *framework* pilihannya. Karena kelebihan-kelebihan yang dimiliki oleh *CodeIgniter*, pembuat PHP *Rasmus Lerdorf* memuji *CodeIgniter* di *frOSCon* (Agustus 2008) dengan mengatakan bahwa dia menyukai *CodeIgniter* karena “*it is faster, lighter and the least like a framework.*”

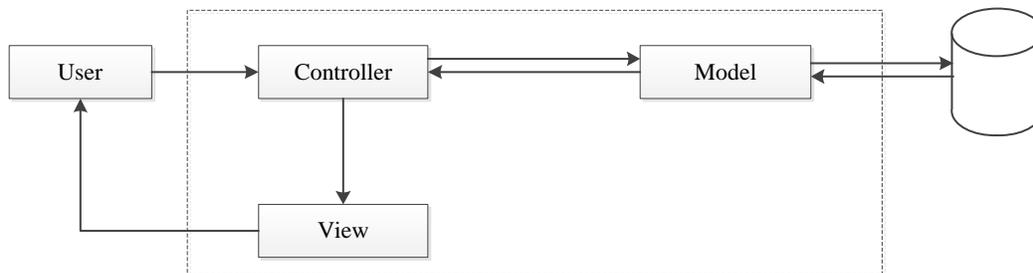
1.4.1 Fungsi *CodeIgniter*

1. Mempercepat dan mempermudah kita dalam pembuatan website.
2. Menghasilkan struktur pemrograman yang sangat rapi, baik dari segi kode maupun struktur file phpnya.
3. Memberikan standar coding sehingga memudahkan kita atau orang lain untuk mempelajari kembali system aplikasi yang dibangun.

1.4.2 MVC

MVC adalah konsep dasar yang harus diketahui sebelum mengenal *CodeIgniter*. MVC (*Model View Controller*) merupakan suatu konsep yang cukup populer dalam pembangunan aplikasi web, berawal pada bahasa pemrograman Small Talk, yang memisahkan bisnis logic (alur piker), data logic (penyimpanan data) dan presentation logic (antarmuka aplikasi) atau secara sederhana adalah memisahkan antara desain, data dan proses. Ada 3 komponen yang

membangun suatu konsep atau pola *Model-View-Controller* (MVC) menurut Raharjo (2015) sehingga kode-kode dapat di sederhanakan.



Gambar 1.1 Arsitektur MVC

Berikut merupakan penjelasan dari arsitektur MVC :

1. Model, biasanya berhubungan dengan data dan interaksi ke database atau webservice. Model juga merepresentasikan struktur data dari aplikasi yang bisa berupa basis data maupun data lain, misalnya dalam bentuk file teks, file XML maupun webservice. Biasanya didalam model akan berisi class dan fungsi untuk mengambil, melakukan update dan menghapus data website. Sebuah aplikasi web biasanya menggunakan basis data dalam menyimpan data, maka pada bagian Model biasanya akan berhubungan dengan perintahperintah query SQL.
2. View, merupakan bagian yang menangani presentation logic. Pada suatu aplikasi web bagian ini biasanya berupa file template HTML, yang diatur oleh controller. View berfungsi untuk menerima dan merepresentasikan data hasil dari model dan controller kepada user. View tidak memiliki akses langsung terhadap bagian model.
3. Controller, merupakan bagian yang mengatur hubungan antara bagian model dan bagian view. Pada controller terdapat class-clas dan fungsi-fungsi yang memproses permintaan dari View ke dalam struktur data di dalam model. Controller juga tidak boleh berisi kode untuk mengakses basis data Karena tugas megakses data telah diserahkan kepada model. Tugas controller adalah menyediakan berbagai variable yang akan ditampilkan di view, memanggil

model untuk melakukan akses ke basis data, menyediakan penanganan kesalahan/error, mengerjakan proses logika dari aplikasi serta melakukan validasi atau cek terhadap input.

1.4.3 Web Based

Web Based menurut Urbietta *et al.* (2019) adalah aplikasi yang dibuat berbasis web yang membutuhkan *web server* dan *browser* untuk menjalankannya. Dengan membuat sistem berbasis *web based* ada beberapa hal yang penting dan harus kita pikirkan sebelum membangun sistem tersebut, diantaranya Infrastruktur jaringan yang dibutuhkan juga cukup besar karena aplikasi yang dibuat dapat diakses dari jaringan luar (internet).

1.4.4 PHP

Menurut Subagja (2018) PHP adalah *Perl Hypertext Preprocessor* bahasa *server-side-scripting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis.

Menurut Aryani, *et al.* (2015) berpendapat bahwa PHP adalah bahasa pemrograman *web server-side* yang bersifat *open source* atau mudah dikembangkan. PHP adalah *script* yang digunakan untuk membuat halaman *website* yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh *client*.

1.4.5 MySQL

Menurut MySQL (2018) *MySQL* adalah singkatan dari *Structure Query Language* yang digunakan untuk mendefinisikan structure data, memodifikasi data pada basis data, menspesifikasi batasan keamanan (*security*), hingga pemeliharaan basis data.

Menurut Amin (2018) mendefinisikan *mysql* MySQL adalah RDBMS yang cepat dan mudah digunakan, serta sudah banyak digunakan untuk berbagai kebutuhan.

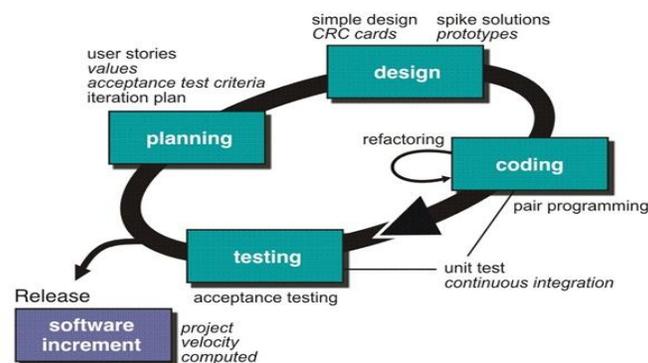
MySQL merupakan bahasa standar yang paling banyak digunakan untuk mengakses *database* relasional dan merupakan aplikasi yang dapat dipergunakan secara bebas.

1.5 Metode Pengembangan Sistem

Metode pengembang sistem merupakan tahapan pembangunan sistem yang dapat diterapkan sebagai alur proses penelitian, metode pengembang sistem digunakan sebagai langkah untuk mendapatkan identifikasi masalah, merancang, implementasi hingga pengujian.

1.5.1 *Extreme Programming*

Menurut (Suryantara, 2017) *extreme programming* merupakan salah satu metode untuk melakukan pengembangan sistem yang memiliki konsep orientasi objek, fleksibel atau dapat menyesuaikan dengan keinginan pengguna serta sederhana yang dapat digambarkan menggunakan diagram yang sesuai dengan konsep orientasi objek.



Gambar 1.2 *Extreme Programming*
Sumber: (Suryantara, 2017)

1.5.2 Tahapan Penelitian

Tahapan dalam penelitian sebagai langkah-langkah penelitian yang harus dikerjakan, berikut adalah tahapan penelitian *extreme programming*.

1. Tahap Perencanaan

Peneliti atau pengembang memutuskan bagaimana hasil *story* dari pengguna dibangun dengan komitmen telah disepakati, adapun *story-story* yang dilakukan dengan cara :

- a. Pengguna menceritakan apa permasalahan pada sistem yang digunakan dan sistem seperti apa yang akan dibangun.
- b. Berdasarkan hasil cerita pengguna maka peneliti menentukan poin pada bagian *value* untuk memutuskan apa saja yang akan dibangun.
- c. Dari hasil kesepakatan tersebut maka peneliti menentukan *acceptence criteria test* yaitu menentukan kriteria-kriteria apa saja yang nantinya sebagai acuan terhadap sistem yang akan di uji.
- d. Sehingga hasil peneliti menyimpulkan berapa kali akan dilakukan *realies* dan perbaikan pada tahap *iteration plan* merencanakan berapa kali akan dilakukan uji terhadap sistem yang dibangun.

2. Tahap Perancangan

Extreme programming pada proses pembuatan desain di lakukan untuk memberikan informasi gambaran sistem yang akan dibangun, berikut adalah beberapa desain yang akan dilakukan oleh peneliti:

- a. CRC card untuk mengenali dan mengatur *object oriented class* yang sesuai dengan pengembangan. Jika pada saat perancangan terdapat ketidak sesuaian maupun perbaikan maka akan dilakukan
- b. *Spike solution* yang dilakukan kepada pengguna untuk mendapatkan kesesuaian antara ke inginan pengguna dengan pengembanan yang dilakukan.

c. *Prototype* adalah bagian perancangan berupa *user interface* dalam bentuk *wireframing* untuk mempermudah pengguna melihat desain sistem.

3. Tahap Pengkodean

Pada proses pengkodean peneliti menyesuaikan terhadap *story* pengguna sehingga sistem yang dibangun sesuai, proses pengkodean yang dilakukan yaitu:

- a. *Pair Programming* merupakan tahap sistem dibangun dengan bahasa pemrograman dan media penyimpanan yang telah disepakati.
- b. *Refactory* merupakan tahapan yang dilakukan ketika terjadi ketidak sesuaian kode program sehingga dilakukan perbaikan guna mendapatkan hasil yang sesuai.

4. Tahap Pengujian

Tahap pengujian dilakukan oleh pengguna sebagai user dengan melakukan uji sesuai dengan *acceptance test* yang telah ditentukan dan disetujui. Unit test yang telah dibuat fokus pada keseluruhan fitur dan fungsional sistem. Sehingga sistem dapat disimpulkan telah sesuai dan dapat di *realies*.

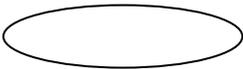
1.6 Alat Pengembang Sistem

Menurut Rosa dan Salahuddin (2019) Alat pengembang sistem merupakan konsep perancangan atau pemodelan yang lebih mengutamakan kepada usulan sistem. Tujuan dari penggunaan alat pengembang sistem tersebut agar konsep orientasi objek dapat disesuaikan dengan pemahaan mengenai pemodelan, adapun bahasa visual untuk memodelkan sistem usulan yaitu dengan menggunakan diagram UML yang merupakan bahasa visual penggambaran sistem usulan dalam bentuk diagram maupun teks.

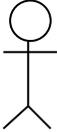
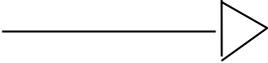
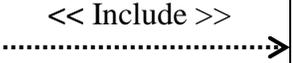
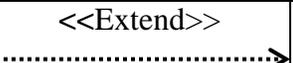
1.6.1 Use Case Diagram

Menurut Rosa dan Salahuddin (2019) *use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Berikut simbol-simbol yang akan digunakan dalam menggambarkan *Use Case Diagram* dapat dilihat pada Tabel 2.3 :

Tabel 1.2 Simbol *Use Case Diagram*

No	Simbol	Deskripsi
1.		<i>Use case</i> : Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor

Tabel 2.3 Simbol *Use Case Diagram* (Lanjutan)

No	Simbol	Deskripsi
2.		Aktor: seseorang/sesuatu yang berinteraksi dengan yang akan dibuat. diluar sistem informasi. Biasanya dinyatakan menggunakan kata benda
3.		Asosiasi (<i>association</i>): merupakan komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4.		Generalisasi (<i>generalization</i>): hubungan (umum – khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum
5.		Include berarti <i>use case</i> yang ditambahkan akan dipanggil saat <i>use case</i> tambahan dijalankan.
6.		Ekstensi (<i>extend</i>) merupakan <i>use case</i> tambahan ke sebuah <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.

Sumber: (Rosa dan Salahuddin, 2019)

1.7 *Black Box Testing*

Menurut Yakub (2012) Pengujian sistem adalah proses untuk mengecek apakah suatu perangkat lunak yang dihasilkan sudah dapat dijalankan sesuai standar atau belum. Pengujian sistem dapat menggunakan metode *black box testing* yaitu merupakan pendekatan komplementer dari teknik *white box testing*, karena pengujian *black box testing* mampu mengungkap kesalahan yang lebih luas. *Black box testing* berfokus pada pengujian persyaratan fungsional perangkat lunak, karena untuk mendapatkan serangkaian kondisi *input* yang sesuai dengan persyaratan fungsional suatu program.

Jadi, pengujian sistem dapat dilakukan dengan pengecekan *input*, pengecekan *output* dan pengecekan proses sebagai berikut:

- a. Pengecekan *input*, meliputi kelengkapan item-item input, kemudahan pengoperasian, kemudahan manipulasi data, dan pengendalian kesalahan.
- b. Pengecekan proses, dilakukan dengan pengecekan *output* program.
- c. Pengecekan *output*, meliputi pengecekan terhadap format dan bentuk-bentuk laporan.

Kelebihan *Black Box Testing* adalah:

- a. Tidak perlu melihat *source code* secara detail.
- b. Mendeteksi kesalahan pengetikan (*typo*).
- c. Mendeteksi kesalahan *Design* atau *User Interface* dari sebuah *software* atau *website*.
- d. Menampilkan asumsi yang tidak sesuai dengan kenyataan, untuk di analisa dan diperbaiki.
- e. Seorang *Tester* tidak harus *Programmer*.

Kekurangan *Black Box Testing* adalah:

- a. Ketergantungan dengan dokumen dan design *software* tersebut.

- b. Tidak sampai level *code*, sehingga tester tidak mengetahui *level security* dari *software* tersebut.

1.7.1 Skala Likert

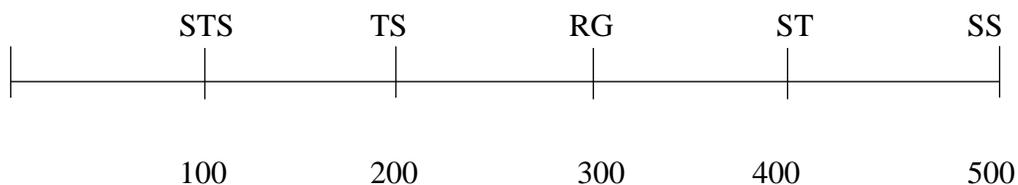
Menurut Sugiyono (2018), *skala likert testing* adalah Skala Likert digunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang tentang fenomena sosial. Dalam penelitian fenomena sosial ini telah ditetapkan secara spesifik oleh peneliti, yang selanjutnya disebut sebagai variabel penelitian.

Dengan *skala likert* maka variabel yang akan diukur dijabarkan menjadi indikator variabel. Kemudian indikator variabel tersebut dijadikan sebagai titik tolak untuk menyusun item-item instrumen yang dapat berupa pernyataan atau pertanyaan. Pengujian menggunakan 5 kategori jawaban dengan bobot yang berbeda untuk setiap jawabannya seperti pada Tabel 2.5.

Tabel 1.3 Bobot Jawaban *Usability*

No.	Pertanyaan	Jawaban				
		SS	ST	RG	TS	STS
1.	Tampilan pada sistem mudah digunakan	5	4	3	2	1

Secara hasil kesimpulan dari hasil kuisisioner yang diperoleh dapat menggunakan pengukuran dari hasil maksimal nilai tertinggi, berikut merupakan gambar aspek *usability* pada Gambar 2.5.



Gambar 1.3 Hasil Pengukuran Aspek *Usability*

Kemudian hasil perhitungan yang didapatkan dari angket, selanjutnya dibandingkan dengan rentang kriteria interpretasi skor untuk menyatakan hasil yang didapatkan dengan rentang pada Tabel 2.6.

Tabel 1.4 Rentang *Criteria Interpretasi*

No	Rentang Kriteria	Kriteria
1	0% - 20%	Sangat Tidak Baik
2	21% - 40%	Tidak Baik
3	41% - 60%	Kurang Baik
4	61% - 80%	Baik
5	81% - 100%	Sangat Baik

(Sumber: Sugiyono, 2018)