

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Pengertian Rancang Bangun**

Rancang Bangun adalah kegiatan menerjemahkan hasil analisa ke dalam bentuk paket perangkat lunak kemudian menciptakan sistem tersebut atau memperbaiki sistem yang sudah ada.

#### **2.2 Pengertian Informasi**

Informasi merupakan sekumpulan data atau fakta yang diorganisasi atau diolah dengan cara tertentu sehingga mempunyai arti bagi penerima.

#### **2.3 Pengertian Sistem Informasi**

Sistem informasi merupakan kumpulan dari sub-sub sistem yang saling berhubungan satu sama lain, dan bekerja sama secara harmonis untuk mencapai suatu tujuan, yaitu mengolah data menjadi informasi yang berguna.

#### **2.4 Pengertian Efektifitas**

Efektifitas berasal dari kata efektif yang berarti berhasil guna. Menurut Ensiklopedi Nasional Indonesia, efektifitas berarti menunjukkan keberhasilan dari segi tercapai atau tidaknya sasaran yang telah ditetapkan, hasil yang makin mendekati sasaran berarti tinggi efektifitasnya. Jadi dapat dikatakan bahwa efektifitas berarti sesuatu yang menunjukkan taraf tercapainya suatu tujuan. Suatu usaha dapat dikatakan efektif kalau usaha itu mencapai tujuan secara ideal. Dengan demikian efektifitas menunjukkan keberhasilan dari segi tercapai atau tidaknya sasaran yang telah ditetapkan. Hasil yang mendekati sasaran berarti tinggi efektifitasnya, sebaliknya hasil yang jauh dari sasaran berarti kurang efektifitasnya.

#### **2.5 Pengertian Efisiensi**

Efisiensi adalah kemampuan untuk meminimalkan penggunaan sumber daya dalam mencapai tujuan organisasi. Seorang yang bertindak secara efisien mampu meminimalkan biaya sumber daya yang diperlukan. Efisiensi kerja merupakan pelaksanaan kegiatan dengan cara yang termudah dalam mengerjakannya, termurah

dalam biayanya, tersingkat dalam waktunya, teringan dalam bebannya dan terpendek dalam jaraknya untuk mencapai tujuannya. Efisiensi merupakan suatu ukuran keberhasilan suatu kegiatan dilihat dari seberapa besar sumber daya yang digunakan untuk mencapai hasil yang di inginkan. Semakin sedikit penggunaan sumber daya yang digunakan untuk mencapai hasil yang diharapkan maka prosesnya dapat dikatakan semakin efisien.

## **2.6 Pengertian Kinerja Karyawan**

Kinerja karyawan adalah kemampuan yang ditunjukkan oleh karyawan dalam melaksanakan tugas atau pekerjaannya. Kinerja dikatakan baik dan memuaskan apabila apabila tujuan yang dicapai sesuai dengan standar yang telah ditetapkan.

## **2.7 *The Open Group Architecture Framework*(TOGAF)**

TOGAF atau *The Open Group Architecture Framework* merupakan sebuah kerangka arsitektur penelitian atau framework yang berguna sebagai alat pendukung penelitian. TOGAF memiliki tools dalam mengimplementasikan penggunaannya. Kerangka TOGAF dimulai pada awal tahun 1990 sebagai pengembangan arsitektur teknis yang merupakan pengembangan dari *forum Open Group*, Pada tahun 1995 TOGAF mulai diperkenalkan dengan keutamaan *Architecture Framework* teknis pengelolaan

Fleksibilitas merupakan sifat dari TOGAF karena kerangka ini memiliki sifat *Open Source*, sistematis namun kaya akan teknis arsitektur. Dalam TOGAF *Architecture Enterprise* dibagi menjadi 4 kategori yaitu:

1. ***Business architecture***

Mendeskripsikan TOGAF dalam menjelaskan proses bisnis dalam memenuhi dan mencapai tujuannya.

2. ***Application architecture***

Bagaimana aplikasi di desain dan berinteraksi satu dengan yang lain.

3. ***Data architecture***

Menjelaskan tentang proses pengelolaan data dan di akses.

4. ***Technical architecture***

Menggambarkan infrastruktur yang digunakan dalam mendukung aplikasi seperti *hardware* dan *software* serta bagaimana interaksinya.

#### 2.7.1 TOGAF ADM

TOGAF *Architecture Development Method* merupakan kunci dalam proses arsitektur TOGAF. ADM dapat memberikan kemungkinan bagi organisasi atau perusahaan dalam mendefinisikan kebutuhan bisnis mereka dengan tujuan membangun arsitektur yang spesifik sesuai dengan kebutuhan. Menurut Yunis, R., & Surendro, K. TOGAF ADM merupakan metode yang kompleks dengan menggunakan model yang bisa digunakan pada proses pengembangan arsitektur. Lingkup cakupan ADM adalah penetapan *framework*, pembangunan konten arsitektur, peralihan, dan mengatur proses realisasi arsitektur, yang diatur melalui siklus iteratif yang terdiri dari 10 fase sebagai berikut:

1. Fase *preliminary*, pada fase ini menjelaskan persiapan dalam melakukan perancangan arsitektur dengan mendefinisikan kerangkadan prinsip dengan tujuan menyelaraskan TOGAF dan tujuan dari diciptakannya perancangan arsitektur.
2. Fase A : *Architecture Vision*, pada fase ini menjelaskan awal dari perancangan arsitektur yang mencakup ruang lingkup, identifikasi stakeholder, penyusunan visi arsitektur, dan izin persetujuan untuk perancangan arsitektur.
3. Fase B : *Business Architecture*, pada fase ini menjelaskan arsitektur bisnis dalam perancangan arsitektur. Tools dan metode umum yang digunakan untuk pengembangan model yang diperlukan adalah *Unified Mode lLanguage(UML)*, *Business Process Modeling Notation(BPMN)* dan *Integration Definition(IDEF)*.

4. Fase C : *Information System Architecture*, pada fase ini menjelaskan pengembangan arsitektur system informasi yang dirancang meliputi 2 (dua) domain yaitu arsitektur aplikasi dan arsitektur data.
5. Fase D : *Technology Architecture*, pada fase ini memfokuskan pada pengembangan arsitektur teknologi yang diperlukan oleh organisasi dan perusahaan.
6. Fase E: *Opportunity and Solution*, pada fase ini menjelaskan manfaat yang di dapatkan dari perancangan arsitektur enterprise dengan Kembali mengevaluasi model yang telah dibangun dengan arsitektur yang telah ada.
7. Fase F : *Migration and Planning*. pada fase ini menjelaskan analisis resiko dan biaya berdasarkan daftar prioritas urutan proyek. Daftar prioritas ditujukan sebagai dasar implementasi dari proyek.
8. Fase G : *Implementation Governance*, pada fase ini menjelaskan proses pengawasan dari perancangan arsitektur. rekomendasi setiap proyek yang di implementasi, merancang kontrak arsitektur dalam proses implementasi.
9. Fase H : *Architecture Change Management*, pada fase ini menjelaskan pengelolaan prosedur perubahan pada arsitektur yang baru setelah selesai di implementasikan.
10. *Requirement* pada fase ini menjelaskan pengelolaan kebutuhan arsitektur.

## **2.8. Bahasa Pemrograman dan Perangkat Lunak Pendukung**

Bahasa pemrograman dan perangkat lunak pendukung yang digunakan dalam penulisan ini adalah antara lain Web, PHP, My SQL, XAMPP, HTML, CSS, dan JavaScript, Dreamweaver cs6.

### **2.8.1 WEB**

Web (*Website*) awalnya merupakan suatu layanan sajian informasi yang menggunakan konsep hiperlink yang memudahkan surfer (sebutan bagi pemakai komputer yang melakukan penyelusuran informasi di Internet) untuk mendapatkan informasi dengan cukup mengklik suatu link berupa teks atau gambar maka informasi dari teks atau gambar akan ditampilkan secara lebih terperinci (detail).

### **2.8.2 PHP (*Hypertext Preprocessor*)**

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman web server-side yang bersifat open source. PHP merupakan script yang terintegrasi dengan HTML dan berada pada server (server side HTML embeded scripting). PHP adalah script

yang digunakan untuk membuat halaman website yang dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat saat halaman itu diminta oleh client. Mekanisme ini menyebabkan informasi yang diterima client selalu yang terbaru uptodate. Semua script dieksekusi pada server dimana script tersebut dijalankan.

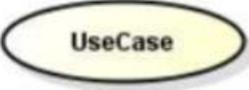
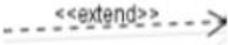
## **2.9 Unified Modeling Language (UML)**

UML adalah *Unified Modeling Language* (UML) merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek.

### **2.9.1 Use Case Diagram**

Merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

**Tabel 2.1 Simbol *Use Case Diagram***

No	Simbol	Keterangan
1.	<p><i>UseCase</i></p> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor.
2.	<p><i>Actor</i></p> 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, walaupun simbol aktor adalah orang namun aktor belum tentu merupakan orang. Biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
3.	<p><i>Association</i></p> 	Komunikasi antara aktor dan use case yang berpartisipasi pada usecase atau use case memiliki interaksi dengan aktor.
4.	<p>Ekstensi/ <i>Extend</i></p> 	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walaupun tanpa use case tambahan itu.
5.	<p><i>Generalization</i></p> 	Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6.	<p><i>Include</i></p> 	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan use case ini.

### 2.9.2 *Class Diagram*

*Class Diagram* menggambarkan struktur sistem dari segi pendefinisian kelas kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Sebuah kelas diagram terdiri dari sejumlah kelas yang dihubungkan dengan garis yang menunjukkan hubungan antar kelas.

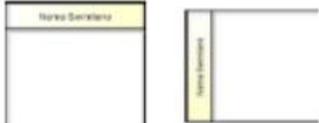
**Tabel 2.2 Simbol *Class Diagram***

Simbol	Keterangan
<p>Kelas/Class</p> 	Kelas pada struktur system
<p>Antar muka/<i>interface</i></p> 	Sama dengan konsep <i>interface</i> dalam pemograman berorientasi objek
<p>Asosiasi/<i>association</i></p> 	Relasi antar kelas dengan makna umum ,asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
<p>Asosiasi berarah/<i>directed</i></p> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
<p>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
<p>Kebergantungan atau <i>Dependency</i></p> 	Relasi antar kelas dengan makna kebergantungan antar kelas.
<p>Agregasi/<i>aggregation</i></p> 	Relasi antar kelas dengan makna semua-bagian(whole-part)

### 2.9.3 Activity Diagram

*Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan actor jadi aktivitas yang dapat dilakukan oleh sistem.

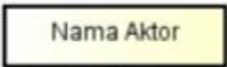
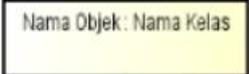
**Tabel 2.3 Simbol *Activity Diagram***

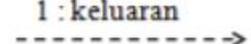
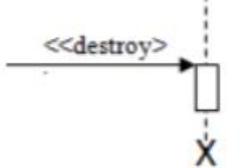
Simbol	Keterangan
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan/ <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

### 2.9.4 Sequence Diagram

*Diagram sequence* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek atau message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan *diagram sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah usecase beserta metode-metode yang dimiliki kelas. Berikut ini adalah simbol-simbol yang ada pada diagram aktivitas:

Tabel 2. 4 *Simbol Sequence Diagram*

Simbol	Keterangan
<p>Aktor</p>  <p>atau</p>  <p>Tanpa waktu aktif</p>	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal <i>frase</i> nama aktor.</p>
<p>Garis hidup</p> 	<p>Menyatakan kehidupan suatu objek</p>
<p>Objek</p> 	<p>Menyatakan objek yang berinteraksi pesan</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya.</p>

Simbol	Keterangan
Pesan tipe <i>create</i> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
Pesan tipe <i>call</i> 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.
Pesan tipe <i>send</i> 	Menyatakan bahwa suatu objek mengirim data /masukan/informasi ke objek lainnya , arah panah mengarah pada objek yang dikirim.
Pesan tipe <i>return</i> 	Menyatakan bahwa suatu objek yang telah menjalankan sesuatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu.
Pesan Tipe <i>Destroy</i> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

## 2.10 Penelitian Terkait

No	Judul Jurnal	Penulis	Keterangan
1	Blueprint Enterprise Architecture Sistem Akademik Sekolah Berbasis Zachman Framework	Dewi Herliah, Sutedi	Dengan di susunnya Arsitektur enterprise dengan menggunakan kerangka kerjazachman yang terdiri dari arsitektur di tingkat akademik, maka di harapkan proses Proses pengembangan sistem akan sesuai dengankebutuhan dan dapat mengatasi permasalahan yang di hadapi oleh SMA Negeri 9 Bandar Lampung

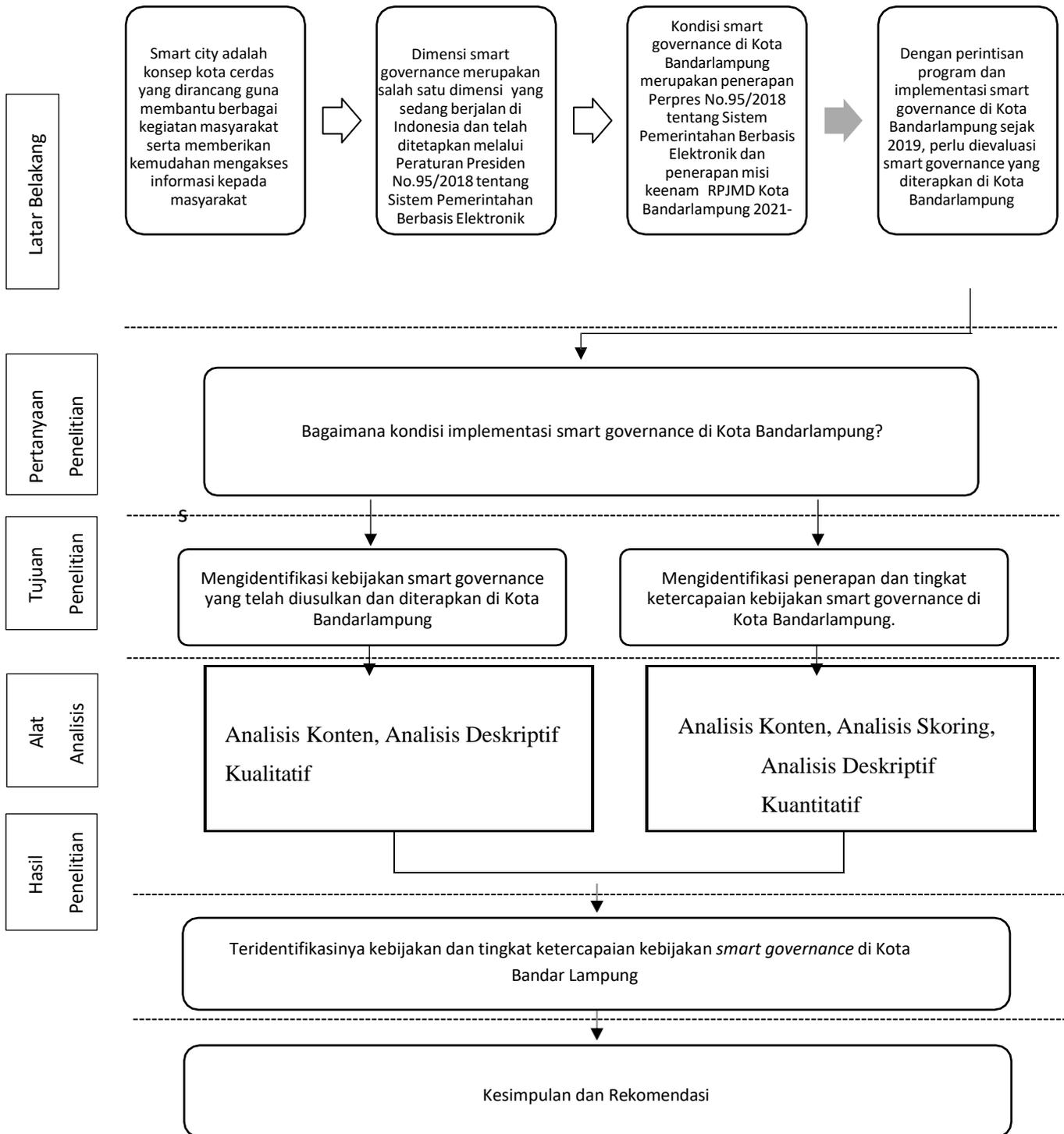
2	Perancangan Arsitektur Enterprise Menggunakan TOGAF (Studi Kasus: Desa Sitorang )	Hikmatulloh, Elisa Silaen, Salmalisa Putri Sudarsono, Mohammad Fergi Argiansyah	Tujuan dari penelitian ini adalah untuk memberikan arahan agar dapat menerapkan sistem informasi untuk penyimpanan data. kinerja di masa yang akan datang
3	Perancangan Arsitektur Enterprise Sistem Informasi menggunakan Togaf Adm Di Sma Negeri 1 Singaraja	Dewa Nyoman Adi Sista, I Made Candiasa, I Gede Aris Gunadi	Penelitian ini bertujuan untuk mendefinisikan serta menguji efektivitas TOGAF ADM sebagai model arsitektur enterprise sistem informasi guna mendukung aktivitas bisnis di SMA Negeri 1 Singaraja.
4	Perancangan Arsitektur Enterprise SMAN 3 Tulang Bawang Tengah Menggunakan Framework TOGAF ADM	Minarni, Wasilah	Diperlukannya suatu perencanaan strategis dalam melakukan perancangan arsitektur enterprise yang dapat menyelaraskan antara perkembangan sistem informasi dan teknologi informasi yang ada dengan proses bisnis organisasi di SMA N 3 TBT. The Open Group Architecture Framework for Architecture Development Method (TOGAF ADM) untuk Arsitektur Entepriasanya
5	Perancangan Enterprise Architecture Menggunakan Framework Togaf Architecture	Krisma Monita, Adhitia Erfina, Cecep Warman	Penelitian ini membuat sebuah perancangan enterprise architecture menggunakan framework TOGAF-ADM (Architecture Development method) untuk memberikan solusi dalam pengoptimalan

	Development Method (Togaf-Adm) Pada SMK BinaMandiri 2		pemanfaatan sistem dan teknologi informasi agar dapat diselaraskan dengan visi, misi, strategi, dan sumberdaya yang dimiliki oleh SMK Bina Mandiri 2 tersebut.
--	---	--	--

No	Nama Peneliti	Judul	Metodologi	Fokus	Lokasi
6	Sri Eniyati, dkk. (2017)	Perhitungan Tingkat Kesiapan Implementasi Smart City Dalam Perspektif Smart Governance Dengan Metode FIS Mamdani	<i>Mix Research Methods</i> , Metode Fuzzy Inference Systems (FIS) Mamdani	Tingkat kesiapan optimal kota untuk menerapkan Smart City	Kota Pekalongan
7	Vanya Rizka Fahira (2021)	Evaluasi Tingkat Kematangan <i>Smart governance</i> di Kota Bandung	Analisis Konten, Analisis Deskriptif Kualitatif, Analisis Deskriptif Kuantitatif, Analisis Skoring	Perumusan indikator tingkat kematangan dan evaluasi <i>smart governance</i>	Kota Bandung
8	Haris Fadila (2021)	Evaluasi Implementasi <i>Smart governance</i> Dalam Mewujudkan Program <i>Smart city</i> di Kota Padang Panjang	Analisis Deskriptif Kualitatif, Analisis Konten, Analisis Skoring	Evaluasi implementasi <i>smart governance</i> dalam konteks program <i>smart city</i>	Kota Padang Panjang
9	Dwi Wahyu Handayani (2021)	Problem Realisasi Kebijakan <i>Smart city</i> di Indonesia: Kasus Kota Bandarlampung	studi literatur, pengamatan dan data, pengambilan kesimpulan	Permasalahan realisasi kebijakan <i>smart city</i>	Kota Bandarlampung
10	Eva Kusuman dari (2020)	Arahan Kebijakan Pengembangan Smart Governance di Kota Metro Dalam Urusan Pembangunan Dan Tata Ruang	Analisis Deskriptif Kualitatif, Analisis Skoring,	merumuskan arahan kebijakan <i>smart governance</i> dalam urusan pembangunan dan tata ruang.	Kota Metro

Sumber: Hasil Analisis Peneliti, 2021

## 2.11 Kerangka Berpikir



Sumber: Hasil Analisis Peneliti, 2021

Gambar 2.1 Kerangka Berpikir