

## **BAB II**

### **TINJAUAN LITERATUR**

#### **2.1 Menurut Penelitian Terkait**

Banyak peneliti-peneliti sebelumnya yang sudah melakukan penelitian terkait dengan penelitian dan literatur terkait dengan pengembangan dan penerapan sistem informasi perjalanan dinas berbasis web. Tinjauan ini bertujuan untuk memberikan gambaran lebih dalam mengenai aspek teknis yang terkait dengan perancangan, pengembangan, dan implementasi sistem informasi berbasis web dalam konteks administrasi pemerintahan, khususnya untuk pengelolaan perjalanan dinas.

##### **2.1.1 Definisi dan Karakteristik Sistem Informasi Perjalanan Dinas**

Sistem informasi perjalanan dinas berbasis web adalah sebuah sistem yang dirancang untuk mengelola, memantau, dan memproses segala bentuk aktivitas perjalanan dinas yang dilakukan oleh pegawai pemerintah, baik di tingkat pusat maupun daerah. Sistem ini biasanya mencakup beberapa modul, di antaranya pengajuan perjalanan dinas, persetujuan, pengelolaan anggaran, pengaturan jadwal, serta laporan perjalanan dan pertanggungjawaban keuangan.

Menurut Rahman (2017) dalam bukunya "**Sistem Informasi Pemerintahan: Konsep dan Implementasi**", SIPD dapat membantu instansi pemerintah dalam mengelola perjalanan dinas dengan lebih efisien, transparan, dan terorganisir.

Di dalam sistem pemerintahan, perjalanan dinas merupakan aktivitas yang sering dilakukan oleh pegawai pemerintah untuk tujuan pekerjaan yang berkaitan dengan pelaksanaan tugas di luar kantor. Oleh karena itu, sistem yang baik diperlukan untuk mengatur seluruh aspek perjalanan dinas mulai dari perencanaan, pengajuan anggaran, hingga pelaporan pertanggungjawaban keuangan dan kegiatan yang dilakukan. Penelitian ini juga mengungkapkan bahwa penerapan sistem berbasis web dalam perjalanan dinas dapat mengatasi masalah ketidakteraturan dan inkonsistensi data yang sering terjadi pada pengelolaan manual.

Menurut Widodo (2018), "**Sistem Informasi Perjalanan Dinas (SIPD) adalah sistem yang memungkinkan pegawai pemerintah untuk mengajukan, menyetujui, dan melaporkan perjalanan dinas secara elektronik, tanpa perlu berhubungan langsung dengan dokumen fisik atau berinteraksi dengan berbagai pihak secara manual**". SIPD berbasis web memungkinkan untuk diproses secara lebih cepat, transparan, dan dapat diakses oleh berbagai pihak dalam instansi pemerintah terkait.

### **2.1.2 Keuntungan Penggunaan Sistem Informasi Berbasis Web**

Sistem informasi berbasis web memiliki banyak keuntungan, antara lain kemudahan akses, efisiensi, dan fleksibilitas. Seperti yang diungkapkan oleh Danuprasetyo (2020) dalam penelitian "**Efektivitas Sistem Informasi Berbasis Web di Lingkungan Pemerintahan Daerah**", sistem berbasis web memungkinkan pengguna untuk mengakses data dan informasi terkait perjalanan dinas dari berbagai lokasi tanpa tergantung pada perangkat atau platform tertentu.

Akses yang lebih mudah ini juga memungkinkan koordinasi yang lebih cepat antara

berbagai pihak yang terlibat dalam perjalanan dinas, baik itu pejabat yang memberikan izin, petugas keuangan, maupun pegawai yang melakukan perjalanan.

Dengan demikian, efisiensi waktu dapat tercapai, karena sistem berbasis web mengurangi kebutuhan untuk memproses data secara manual dan memungkinkan pengguna untuk melakukan pengajuan, persetujuan, dan pelaporan secara elektronik. Proses yang otomatis ini mengurangi kemungkinan kesalahan input data, mempercepat proses administrasi, dan memastikan keakuratan data.

Selain itu, kelebihan lain yang ditemukan dalam penerapan sistem berbasis web adalah kemudahan dalam melakukan pemantauan dan pengawasan. Menurut studi yang dilakukan oleh Susilo dan Wijaya (2019) dalam "**Optimasi Sistem Informasi Berbasis Web pada Instansi Pemerintahan**", penggunaan dashboard berbasis web memudahkan atasan atau pihak yang berwenang untuk memantau status perjalanan dinas secara real-time, baik itu status pengajuan, anggaran yang telah disetujui, maupun laporan pertanggungjawaban yang telah dikirim.

### **2.1.3 Transparansi dan Akuntabilitas dalam Pemerintahan**

Salah satu isu yang paling penting dalam pengelolaan administrasi pemerintahan adalah **transparansi** dan **akuntabilitas**. Penelitian yang dilakukan oleh Haryanto dan Santosa (2021) dalam jurnalnya "**Meningkatkan Transparansi dan Akuntabilitas Melalui Sistem Informasi Berbasis Web di**

**Pemerintahan Daerah"** menunjukkan bahwa sistem berbasis web dapat menciptakan proses yang lebih transparan. Dengan adanya sistem yang mencatat setiap tahap perjalanan dinas secara digital, semua informasi dapat diakses dengan mudah oleh pihak yang berwenang. Hal ini memastikan bahwa proses pengajuan, persetujuan, dan pelaporan dilakukan secara transparan, tanpa ada manipulasi data.

Dengan menggunakan sistem berbasis web, instansi pemerintah juga dapat mengurangi potensi kecurangan dan penyalahgunaan anggaran perjalanan dinas. Oleh karena itu, dalam implementasi sistem informasi berbasis web di pemerintahan, transparansi dan akuntabilitas menjadi salah satu nilai utama yang dijaga untuk meningkatkan integritas pemerintahan.

#### **2.1.4 Pemanfaatan Sistem Informasi dalam Pengelolaan Perjalanan Dinas**

Beberapa penelitian juga menyoroti pemanfaatan sistem informasi dalam meningkatkan pengelolaan perjalanan dinas. Penelitian oleh Mulyadi dan Kurniawan (2018) dalam "**Penerapan Sistem Informasi dalam Pengelolaan Perjalanan Dinas untuk Meningkatkan Kinerja Pegawai Pemerintah**" menyimpulkan bahwa penggunaan sistem informasi yang terintegrasi mampu meningkatkan kinerja pegawai dengan mengurangi waktu yang dihabiskan untuk urusan administrasi perjalanan dinas. Dengan sistem yang terkomputerisasi, proses persetujuan perjalanan dinas menjadi lebih cepat, meminimalisir potensi kesalahan, dan memungkinkan petugas untuk fokus pada tugas utama mereka.

Lebih jauh lagi, penelitian tersebut mengungkapkan bahwa sistem berbasis web yang terintegrasi dengan sistem lain, seperti sistem keuangan dan sistem manajemen pegawai, mampu memberikan gambaran yang lebih jelas mengenai anggaran yang tersedia, pengeluaran yang telah dilakukan, dan status perjalanan dinas secara keseluruhan. Hal ini sangat penting dalam pengelolaan anggaran pemerintah yang sering kali melibatkan berbagai pihak dan instansi terkait.

### **2.1.5 Keamanan dan Privasi Data dalam Sistem Berbasis Web**

Salah satu tantangan terbesar dalam penerapan sistem berbasis web adalah masalah **keamanan data**. Mengingat informasi yang dikelola dalam sistem perjalanan dinas bersifat sensitif, seperti data pegawai, anggaran, dan rincian perjalanan, maka sistem tersebut harus dilengkapi dengan mekanisme pengamanan yang ketat.

Penelitian oleh Handayani dan Rahmawati (2021) dalam "**Analisis Keamanan dan Kerentanannya pada Sistem Informasi Pemerintahan Berbasis Web**" menekankan pentingnya pengamanan data dalam sistem berbasis web. Beberapa teknik yang digunakan untuk memastikan data tetap aman antara lain penggunaan enkripsi data, autentikasi pengguna yang kuat, serta pemantauan terhadap potensi ancaman dari luar (hacker) atau dari dalam (kesalahan administrasi).

Salah satu contoh penerapan pengamanan data yang baik dalam sistem berbasis web adalah penggunaan **SSL (Secure Sockets Layer)** dan **TLS (Transport Layer Security)** yang dapat mengamankan data yang dikirim melalui jaringan internet. Selain itu, pengelolaan hak akses pengguna yang ketat, sehingga hanya

pihak yang berwenang yang dapat mengakses atau memodifikasi data tertentu, juga menjadi salah satu metode penting dalam menjaga integritas data.

### **2.1.6 Tantangan dalam Implementasi Sistem Perjalanan Dinas Berbasis Web**

Walaupun sistem informasi berbasis web menawarkan banyak manfaat, implementasinya tidak selalu berjalan mulus. Penelitian oleh Utama dan Kurniawan (2020) dalam "**Tantangan Implementasi Sistem Informasi Berbasis Web di Pemerintahan Daerah**" mengidentifikasi beberapa hambatan yang dihadapi dalam penerapan sistem ini, antara lain:

- **Kurangnya Infrastruktur Teknologi:** Banyak daerah di Indonesia, terutama di daerah-daerah yang lebih terpencil, belum memiliki infrastruktur teknologi yang memadai, seperti koneksi internet yang stabil dan perangkat keras yang mendukung. Hal ini menjadi kendala dalam penerapan sistem informasi berbasis web.
- **Resistensi terhadap Perubahan:** Penggunaan sistem baru sering kali menemui hambatan dalam bentuk resistensi dari pegawai yang sudah terbiasa dengan cara kerja lama. Hal ini memerlukan upaya pelatihan dan sosialisasi yang lebih intensif agar pegawai dapat beradaptasi dengan sistem yang baru.
- **Keterbatasan Sumber Daya Manusia:** Beberapa daerah kekurangan tenaga ahli di bidang teknologi informasi yang mampu mengelola dan memelihara sistem informasi berbasis web. Oleh karena itu, pengadaan tenaga ahli yang

terampil dan berpengalaman sangat dibutuhkan untuk memastikan keberlanjutan dan pengembangan sistem.

### 2.1.7 Keamanan Sistem Informasi Perjalanan Dinas

Keamanan menjadi salah satu aspek yang sangat penting dalam pengembangan sistem informasi berbasis web, terutama yang berkaitan dengan data

sensitif pegawai dan anggaran perjalanan dinas. Penelitian oleh Handayani dan Rahmawati (2021) dalam "**Analisis Keamanan dan Kerentanannya pada Sistem Informasi Pemerintahan Berbasis Web**" mengidentifikasi beberapa aspek keamanan yang harus diperhatikan dalam pembangunan SIPD berbasis web:

- **Autentikasi dan Otorisasi Pengguna:** Setiap pengguna sistem, baik pegawai maupun pejabat pengesah perjalanan dinas, harus memiliki akses yang terbatas sesuai dengan hak akses yang dimilikinya. Penggunaan autentikasi berbasis **OAuth, JWT (JSON Web Tokens)**, atau **2-Factor Authentication (2FA)** dapat meningkatkan tingkat keamanan sistem.
- **Enkripsi Data:** Data yang dikirimkan antara client dan server harus dienkripsi untuk mencegah pencurian data. Protokol enkripsi seperti **SSL/TLS** sangat penting untuk memastikan komunikasi antara aplikasi dan pengguna tetap aman.
- **Pengelolaan Hak Akses:** Sistem harus memiliki mekanisme untuk membatasi akses data berdasarkan peran pengguna (**role-based access control**)

/ RBAC). Hal ini memastikan bahwa hanya pihak yang berwenang yang dapat mengakses data tertentu.

- **Pemantauan Keamanan:** Log aktivitas pengguna dan upaya akses tidak sah harus dipantau secara terus-menerus. Sistem harus memiliki kemampuan untuk mendeteksi dan merespon potensi ancaman keamanan secara real-time.

### 2.1.8 Tantangan dalam Implementasi Sistem Informasi Berbasis Web

Walaupun penerapan sistem informasi berbasis web menawarkan banyak keuntungan, ada berbagai tantangan teknis yang harus dihadapi dalam implementasinya. Beberapa tantangan yang sering ditemukan antara lain:

1. **Keterbatasan Infrastruktur Teknologi:** Beberapa daerah atau instansi pemerintah menghadapi keterbatasan infrastruktur IT, terutama di daerah-daerah terpencil yang memiliki akses internet terbatas. Penelitian oleh Sutrisno (2019) menunjukkan bahwa infrastruktur yang belum memadai dapat menyebabkan kegagalan dalam implementasi sistem berbasis web, seperti lambatnya pengiriman data dan kesulitan dalam pemeliharaan sistem.
2. **Resistensi terhadap Teknologi Baru:** Pengguna (pegawai pemerintah) sering kali terbiasa dengan cara kerja manual dan mungkin merasa kesulitan untuk beradaptasi dengan sistem baru. Sosialisasi dan pelatihan yang intensif perlu dilakukan untuk mengatasi hambatan ini. Selain itu, **change management** yang baik sangat diperlukan untuk memperkenalkan sistem ini.

3. **Kompleksitas Integrasi dengan Sistem Lain:** SIPD sering kali harus berintegrasi dengan sistem lain, seperti sistem manajemen keuangan, sistem pengelolaan SDM, dan sistem laporan keuangan. Integrasi ini bisa menjadi tantangan, terutama jika sistem yang ada sebelumnya tidak memiliki standar API yang baik atau tidak kompatibel dengan sistem baru.
4. **Keamanan Data dan Privasi:** Sistem yang mengelola data sensitif seperti informasi pegawai dan anggaran perjalanan dinas memerlukan perhatian khusus terhadap aspek keamanan dan privasi data. Penelitian oleh Andriani (2020) menunjukkan bahwa kebocoran data dalam sistem pemerintahan dapat merusak reputasi dan kepercayaan publik.

## 2.2 Perancangan

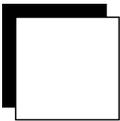
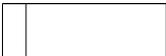
Perancangan adalah proses menciptakan rencana atau desain untuk mencapai tujuan tertentu (Nugroho, 2016). Proses ini terdiri dari beberapa langkah antara lain menentukan kebutuhan, mengumpulkan informasi, menganalisisnya, mengembangkan konsep, mengembangkan solusi, dan mengevaluasinya. Perancangan dapat digunakan secara lebih luas dalam berbagai domain, termasuk bisnis, teknik, arsitektur, dan teknologi informasi. Tujuan utama perancangan adalah untuk menghasilkan solusi praktis dan hemat biaya yang menjawab permintaan atau menyelesaikan masalah yang diketahui.

## 2.3 *Data Flow Diagram*

*Data Flow Diagram* (DFD) merupakan gambaran visual tentang struktur sebuah sistem, termasuk komponen-komponennya, aliran data di antara mereka,

serta asal, tujuan, dan penyimpanan data tersebut (Budiani, 2000). *Data Flow Diagram* (DFD) adalah alat pembuatan model yang umum dipakai, terutama jika fungsi-fungsi sistem lebih krusial dan rumit daripada data yang diolah oleh sistem (Bukovský, 2017). Pada DFD, langkah-langkah aliran informasi dalam sistem dijelaskan, dimulai dari *input* awal hingga menghasilkan *output* (Teang, Faizah and Nurcahyo, 2023). DFD memanfaatkan simbol-simbol grafis untuk menggambarkan entitas seperti pengguna atau sistem lain, proses seperti pengolahan data, penyimpanan data, dan pergerakan data antara komponen-komponen tersebut.

Tabel 2.1 Simbol DFD

No	Simbol	Keterangan
1.		Sebagai transformasi data, komponen fisik yang tidak ditentukan.
2.		Sebagai sumber data target dari sistem yang sedang dijelaskan.
3.		Sebagai petunjuk arah tertentu dari sumber ke tujuan.
4.		Sebagai menyimpan <i>file</i> transaksi ataupun referensi.

#### 2.4 *Entity Relationship Diagram* (ERD)

*Entity Relationship Diagram* (ERD) merupakan visualisasi grafis dari struktur data dalam sebuah sistem informasi, menampilkan entitas dan relasinya.

Keterkaitan antara ERD, CDM, dan PDM dijelaskan sebagai berikut:

1. ERD mendukung perancangan hubungan antara entitas-entitas dan struktur data secara keseluruhan.
2. CDM memberikan gambaran tingkat tinggi tentang konsep dan keterkaitan data tanpa memperhatikan rincian teknis.
4. PDM mengonsep dari CDM dan mengubahnya menjadi struktur fisik yang dapat diimplementasikan dalam basis data.

Maka, ERD menjadi perantara dalam merancang CDM, yang kemudian diterapkan dalam PDM saat merancang basis data.

## **2.5. Tools yang Digunakan**

Berikut adalah tools yang digunakan untuk membangun aplikasi SPPD:

### **2.5.1. BPMN (Business Process Modelling Notation)**

Business Process Modelling Notation (BPMN) merupakan notasi grafis yang dirancang untuk perancangan proses bisnis maupun implementasi proses bisnis yang dikelola oleh Object Management Group (OMG) [5]. BPMN menyediakan notasi grafis untuk menspesifikasikan proses bisnis dalam sebuah Business Process Diagram (BPD). Tujuan utama pemodelan proses bisnis menggunakan BPMN adalah untuk memodelkan proses bisnis yang mudah dipahami oleh semua pengguna bisnis ( business user ), dari analis bisnis ( business analyst ) yang membuat draf inisial dari proses, hingga ke pengembang teknis yang bertanggungjawab untuk mengimplementasikan teknologi yang akan melakukan proses tersebut, dan akhirnya ke pelaku bisnis ( business people ) Sehingga BPMN menjadi standar yang menjembatani kesenjangan (gap) antara perancangan proses bisnis dan implementasi proses bisnis [5].

Pada mulanya BPMN dirancang oleh Business Process Management Initiative (BPMI), kemudian dikembangkan oleh Object Management Group pada 2005. Versi terakhir BPMN yaitu versi 2.0 yang dirilis pada Januari 2011 [5]. Notasi pemodelan yang dimiliki oleh BPMN didukung oleh aplikasi - aplikasi pemodelan proses bisnis, dan perusahaan - perusahaan mulai menggunakannya sebagai teknik pemodelan yang standar [5]. Saat ini standardisasi BPMN dipegang oleh sebuah badan, yaitu Object Management Group (OMG) [5]. Terdapat tiga jenis sub-model dasar dari BPMN, yaitu [5] :

1. Process Jenis model BPMN process menggambarkan urutan atau aliran aktivitas yang terorganisir untuk melakukan suatu pekerjaan. Pada model jenis process terdapat activity event , gateway , serta sequence flow yang membentuk suatu graf berarah.
2. Collaboration Jenis model BPMN collaboration menggambarkan hubungan antar kelas participant yang saling berkolaborasi. Participant digambarkan dengan pool. Interaksi antar pool dapat berupa message flow atau process di dalam pool atau choreography antar pool.
3. Choreography Jenis model BPMN choreography menggambarkan urutan proses interaksi antar participant yang berfokus pada pertukaran informasi/message antar participant. Dalam penggambaran proses pada BPMN, proses terdiri atas kontrol logika seperti urutan, pemilihan (choice), eksekusi yang bersifat parallel serta iterasi.

### 2.5.2. UML

UML (*unified Modeling Language*) adalah salah satu standar Bahasa yang banyak digunakan untuk di dunia industry untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemograman berorientasi objek. UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasi, menggambarkan dokumentasi dari sistem perangkat lunak. UML meliputi *usecase Diagram, Skenario, Class Diagram* [6].

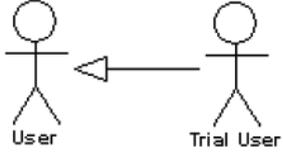
### 2.5.3. Use Case Diagram

Use case atau diagram use case merupakan pemodelan untuk kelakuan (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 didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Syarat penamaan pada use case adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada use case yaitu pendefinisian apa yang disebut aktor dan use case.

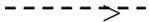
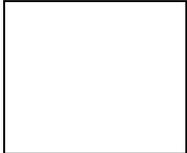
1. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
2. Use case merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau actor.

Berikut adalah symbol – symbol yang digunakan pada Use Case :

Tabel 2.2 Simbol – simbol dalam Use Case

NO.	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Orang, Proses atau aplikasi lain yang berinteraksi dengan aplikasi informasi yang akan dibuat. Aktor mewakili external entity dalam sistem dan bisa manusia, perangkat keras, atau sistem lainnya.
2		<i>Generalization</i>	Hubungan antara dua buah <i>use case</i> dan fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
3		Include	Fungsi atau syarat yang dijalankan <i>use case</i>
		Extend	relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> bisa dapat berdiri sendiri walau tanpa <i>use case</i> tambahan.
4		Association	Komunikasi antara actor dan <i>use case</i>
5		<i>Use case</i>	fungsionalitas yang disediakan aplikasi sebagai

			unit-unit yang saling bertukar pesan antar unit atau actor.
6		System / Entity	Menspesifikasikan paket yang menampilkan aplikasi secara terbatas

7		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
8		<i>Collaboration</i>	Interaksi aturan – aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen – elemennya.
9		Boundary	Disimbolkan dalam bentuk kotak yang mewadahi use case, sebagai representasi

			<p>dari ruang lingkup sistem yang akan dikembangkan. Biasanya digunakan apabila terdapat beberapa alternatif sistem, yang dapat dijadikan pilihan.</p>
--	--	--	--

#### 2.5.4. ERD

Pemodelan awal basis data yang paling banyak digunakan yaitu Entity Relationship Diagram (ERD). ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional.

1. Pengertian Entity Relationship Diagram (ERD) Menurut Rosa dan Shalahuddin (2013:53) “ERD adalah bentuk paling awal dalam melakukan perencanaan basis data relasional. Jika menggunakan OODBMS maka perancangan ERD tidak perlu dilakukan”. Lain pula menurut Ladjamudin (2013:142) “Entity Relationship Diagram (ERD) adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak”. Jadi dapat disimpulkan bahwa ERD (Entity Relationship Diagram) merupakan tahapan perancangan basis data yang menggunakan susunan data berupa gambar.
2. Komponen Entity Relationship Diagram (ERD) Komponen ERD terbagi menjadi beberapa bagian menurut Chen dalam buku Rosa dan Shalahuddin (2013:50) yaitu:

**Tabel 2.3. Simbol pada Entity Relationship Diagram**

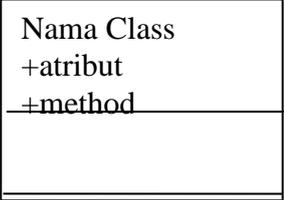
Notasi	Keterangan
	Entitas adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai.
	Relasi menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda.
	Atribut berfungsi mendeskripsikan karakter entitas (atribut yang berfungsi sebagai <i>key</i> diberi garis bawah).
	Garis sebagai penghubung antara relasi dan entitas atau relasi dan entitas dengan atribut.

### 2.5.5. Class Diagram

Class diagram merupakan himpunan dari objek – objek yang sejenis. Sebuah objek memiliki keadaan sesaat (*state*) dan perilaku (*behavior*). *State* sebuah objek adalah kondisi objek tersebut yang dinyatakan dalam *atribut/properties*.

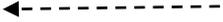
Sedangkan perilaku suatu objek mendefinisikan bagaimana sebuah objek bertindak/beraksi dan memberikan reaksi .

Tabel 2.4. Simbol – simbol *class* diagram

Simbol	Nama Komponen	Keterangan
	<i>Class</i>	<p><i>Class</i> adalah blok – blok pembangun pada pemograman berorientasi obyek. <i>Class</i> digambarkan sebagai sebuah kotak yang terbagi 3 bagian. Bagian atas nama dari class, yang kedua atribut yang ketiga <i>method</i> dari sebuah <i>class</i>.</p>

		Sebuah asosiasi
--	--	-----------------

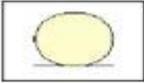
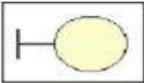
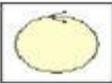


	<i>Composition</i>	class yang lain, maka class tersebut memiliki relasi <i>composition</i> terhadap class tempat dia bergantung.
	<i>Dependency</i>	Sebuah <i>class</i> menggunakan class yang lain. Digunakan untuk menunjukan operasi pada suatu class yang menggunakan <i>class</i> yang lain.
	<i>Aggregation</i>	Relasi.

### 2.5.6. *Sequence Diagram*

*Sequence diagram* adalah suatu diagram yang menggambarkan interaksi antar obyek dan mengindikasikan komunikasi diantara obyek-obyek tersebut. Diagram ini juga menunjukkan serangkaian pesan yang dipertukarkan oleh obyek-obyek yang melakukan suatu tugas atau aksi tertentu. Obyek-obyek tersebut kemudian diurutkan dari kiri ke kanan, aktor yang menginisiasi interaksi biasanya ditaruh di paling kiri dari diagram. Berikut merupakan simbol yang ada pada *sequence* diagram.

**Table 2.5. Simbol pada *Sequence Diagram***

Simbol	Deskripsi
Actor 	Menggambarkan orang yang berinteraksi dengan sistem.
Entity Class 	Menggambarkan hubungan kegiatan yang akan dilakukan.
Boundary Class 	Menggambarkan sebuah penggambaran dari form.
Control Class 	Menggambarkan penghubung antara boundary dengan tabel.
Lifeline 	Menggambarkan tempat mulai dan berakhirnya sebuah pesan.
Line Message 	Menggambarkan pengiriman pesan.

## 2.6. *Tools* Pembangunan Aplikasi

Berikut ini adalah *Tools* dalam pembangunan aplikasi SPPD :

### 2.6.1. Bahasa Pemrograman

Pemrograman digunakan untuk proses menulis, menguji dan memperbaiki (*debug*), dan memelihara kode yang membangun sebuah program komputer. Kode ini ditulis dalam berbagai bahasa pemrograman. Tujuan dari pemrograman adalah untuk membuat suatu program yang dapat melakukan suatu perhitungan atau pekerjaan sesuai dengan keinginan si pemrogram.

## 1. *PHP (Hypertext Preprocessor)*

PHP adalah salah satu Bahasa pemrograman skrip yang dirancang untuk membangun aplikasi *web*. Aplikasi *web* adalah aplikasi yang disimpan dan dieksekusi (oleh *PHP Engine*) di lingkungan *web server*. PHP sangat mudah untuk digunakan karena memiliki beberapa keunggulan, yaitu:

- a) *Open source* dan gratis. Teknologi semakin lama semakin berkembang dan semakin canggih sehingga lebih terjangkau, murah, mudah dan aman.
- b) *Life Cycle* karena PHP selalu up to date mengikuti perkembangan teknologi.
- c) *Cross Platform* yaitu dapat dipakai disemua webserver dan dijalankan di sistem operasi (Linux, Windows, FreeBSD).
- d) Mendukung ke banyak *database* baik gratis maupun komersil. Kemudahan dalam mendapatkan dokumentasi di internet dan banyaknya aplikasi berbasis web yang dibuat dengan menggunakan PHP.

## 2. HTML

*Hypertext Markup Language* (HTML) adalah bahasa *standard* yang digunakan untuk menampilkan halaman *web*. Yang bias dilakukan dengan HTML yaitu mengatur tampilan dari halaman *web* dan isinya, membuat tabel dalam halaman *web*, Mempublikasikan halaman *web* secara *online*, membuat *form* yang bias digunakan untuk menangani registrasi dan transaksi via web, menambahkan objek-objek seperti citra, audio, video, animasi, java applet dalam halaman *web*, menampilkan area gambar (canvas) di *browser*.

### 3. *Codeigniter*

*CodeIgniter* adalah sebuah *framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi berbasis PHP dibandingkan jika menulis semua kode program dari awal. *Framework* merupakan kumpulan potongan-potongan program yang disusun atau diorganisasikan sedemikian rupa, sehingga dapat digunakan untuk membantu membuat aplikasi utuh tanpa harus membuat kodenya dari awal.

#### 2.6.2. **Basis Data**

Basis Data kumpulan data yang saling berhubungan secara logikal serta deskripsi dari data tersebut, yang dirancang untuk memenuhi kebutuhan informasi suatu organisasi. Basis Data adalah sebuah penyimpanan data yang besar yang bisa digunakan oleh banyak pengguna.

Menurut Rosa dan Shalahuddin (2013:43) “Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan”. Selain itu, menurut Madcoms (2011:12) “*Database* atau sering juga disebut basis data adalah sekumpulan informasi yang disimpan dalam komputer secara sistematis dan merupakan sumber informasi yang dapat diperiksa menggunakan suatu program komputer”. Dari penjelasan di atas dapat disimpulkan bahwa *database* atau basis data ialah data yang disimpan dalam komputer yang telah diolah suatu program, dimana dengan program tersebut pengguna tidak akan sulit mencari data yang dibutuhkan.

### ***MySql***

*MySQL* merupakan *software* RDBMS (atau *server database*) yang dapat mengelola *database* dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak *user* (*multi-user*), dan dapat melakukan suatu proses secara sinkron atau berbarengan (*multi-threaded*). *MySQL* sebuah program *database* server yang mampu menerima dan mengirimkan datanya dengan sangat cepat, multi user serta menggunakan perintah SQL.

### **2.6.3. Web Server**

*Web server* digunakan untuk menjelajah situs *web* yang berisikan semua informasi yang ada di dalam situs *web*.

Dari penjelasan di atas dapat disimpulkan bahwa *web server* ialah sebuah aplikasi yang menyediakan layanan untuk mengakses program lebih mudah melalui protokol komunikasi. Contohnya adalah *Apache*.

*Apache Web Server* adalah sebuah nama web server yang bertanggung jawab pada *request-response HTTP* dan *logging* informasi secara *detail* (kegunaan dasarnya). Selain itu, *Apache* juga diartikan sebagai suatu *web server* yang kompak, modular, mengikuti standar protokol HTTP, dan tentu saja sangat digemari[12].

Menurut Kurniawan (2008b:2) *Apache* adalah *web server* yang dapat dijalankan dibanyak sistem operasi (Unix, BSD, Linux, *Microsoft windows* dan *Novell Netware* serta *Platform* lainnya) yang berguna untuk memfungsikan situs *web*. Protokol yang digunakan untuk melayani fasilitas web ini menggunakan HTTP.

#### 2.6.4. *Text Editor*

*Text Editor* merupakan sebuah software aplikasi atau program komputer yang memungkinkan penggunanya membuat, mengubah, atau mengedit *file* teks (plain text). *Text editor* dapat digunakan untuk membuat program-program komputer, mengedit *source code* bahasa pemrograman, serta membuat halaman *web* atau *template web design* dan pembuat aplikasi tertentu. Aplikasi ini secara umum banyak digunakan untuk tujuan pemrograman. Contohnya adalah *Sublime Text*.

*Sublime text* salah satu kode editor yang biasa digunakan oleh para *programmer* untuk membuat suatu program. Menurut Supono dan Putratama (2016:14) “*Sublime text* merupakan perangkat lunak *text editor* yang digunakan untuk membuat atau meng-edit suatu aplikasi. *Sublime text* mempunyai fitur *plugin* tambahan yang memudahkan *programmer*”. Selain itu, menurut Faridi (2015:3) menjelaskan bahwa “*Sublime Text 3* adalah editor berbasis *python*, sebuah *text editor* yang elegan, kaya akan fitur, *cross platform*, mudah dan *simple* yang cukup terkenal di kalangan *developer* (pengembang), penulis dan desainer”. Jadi dapat disimpulkan bahwa *sublime text* ialah teks editor yang digunakan untuk membuat program aplikasi yang secara otomatis untuk mempermudah *programmer* dalam mengetikkan kode editor.

## **2.7. Pengujian**

Berikut adalah pengujian yang digunakan untuk membangun aplikasi SPPD:

### **2.7.1. *Black Box Testing***

Pengujian *Black Box* menunjukkan bahwa fungsi-fungsi perangkat lunak operasional, bahwa output dihasilkan secara benar dari input, dan bahwa *database* diakses dan di update secara benar. Untuk melakukan test seperti itu, pemakai perlu mempunyai pengetahuan seperti yang disyaratkan dalam keperluan pemakai. Dengan demikian, pengujian *black box* tidak secara langsung memeriksa sintaks dan struktur logis internal dari perangkat lunak tersebut, dan oleh karenanya ia tidak menjadi alternative dari pengujian *white box*.

Menurut Rosa dan Shalahuddin (2013:275) “*Black Box Testing* yaitu 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.”

### **2.7.2. *User Acceptance Testing (UAT)***

*User Acceptance Testing (UAT)* merupakan proses verifikasi bahwa solusi yang dibuat dalam sistem sudah sesuai untuk pengguna. Proses ini berbeda dengan pengujian sistem (memastikan *software* tidak crash dan sesuai dengan dokumen permintaan pengguna), melainkan memastikan bahwa solusi dalam sistem tersebut akan bekerja untuk pengguna (yaitu, tes bahwa pengguna menerima

solusi di dalam sistem). UAT umumnya dilakukan oleh klien atau pengguna akhir, biasanya tidak fokus pada identifikasi masalah sederhana seperti kesalahan ejaan, maupun di cacat *showstopper*, seperti crash perangkat lunak. Penguji dan pengembang mengidentifikasi dan memperbaiki masalah ini selama tahap awal pengujian fungsionalitas, pengujian saat integrasi dan pada tahap sistem testing.