

BAB 2 TINJAUAN PUSTAKA DAN DASAR TEORI

2.1. Tinjauan Pustaka

Tinjauan pustaka dilakukan sebagai cara untuk mengetahui metode penyelesaian untuk masalah yang berkaitan dengan ketidakpastian waktu penyelesaian *order* melalui penelitian-penelitian terdahulu. Beberapa solusi serta penggunaan metode-metode yang berkaitan dengan permasalahan tersebut dapat ditemukan melalui tinjauan pustaka ini. Tinjauan pustaka ini dibuat berdasarkan referensi dari jurnal/*paper* yang dicari menggunakan *Google Scholar*. Beberapa kata kunci yang digunakan untuk pencarian pustaka yaitu “waktu penyelesaian *order*”, “penjadwalan produksi”, “sistem *order* percetakan”, “keterlambatan penyelesaian *order*”, dan “perancangan sistem informasi”.

2.1.1. Penelitian Terdahulu Terkait Penjadwalan Produksi

a. Penggunaan Metode *Earliest Due Date* (EDD)

Menurut Karim dan Kurniati dalam (Pradana., dkk. 2020), dikatakan bahwa penjadwalan merupakan proses pengurutan pekerjaan secara menyeluruh pada setiap proses yang dikerjakan pada tiap-tiap mesin, yang fungsinya juga digunakan untuk membantu pengalokasian sumber daya dan mesin yang tersedia agar dapat dilakukan pengurutan pekerjaan dengan batasan-batasan tertentu. Tujuan dari dibuatnya aktivitas penjadwalan yaitu untuk mereduksi waktu proses dan meningkatkan produktivitas dengan melakukan peningkatan terhadap alokasi sumber daya yang ada, kemudian mengurangi keterlambatan dalam proses kerja yang terdapat *deadline* penyelesaian, serta sebagai alat pengambil keputusan pada perencanaan kapasitas yang dibutuhkan oleh suatu industri atau perusahaan.

Penelitian sebelumnya yang membahas tentang perancangan sistem penjadwalan dengan berbagai macam metode sudah dilakukan oleh beberapa peneliti, Pradana dkk. (2020) dan Hamida dan Sugondo (2020) melakukan penelitian mengenai perancangan sistem penjadwalan produksi menggunakan metode *Earliest Due Date* (EDD) dengan tujuan untuk meminimalkan keterlambatan penyelesaian *order* produksi. Penggunaan metode tersebut mampu mengurangi waktu jumlah keterlambatan dengan optimal dengan mengatur urutan pengerjaan *order* berdasarkan tanggal jatuh tempo terdekat yang pertama kali diselesaikan.

b. Penggunaan Metode *Campbell Dudek Smith*

Yusuf dan Aryanny (2022) yang melakukan penelitian di CV.AM. Nanda Putro dan Saputro dan Mundari (2017) yang melakukan penelitian di PT Bioli Lestari juga membahas mengenai perancangan sistem penjadwalan produksi dengan metode yang berbeda dari penelitian sebelumnya yaitu menggunakan metode *Campbell Dudek Smith* dengan tujuan agar lebih efektif dalam mengurangi waktu total produksi ataupun waktu keterlambatan. Pada penelitian Yusuf dan Aryanny (2022), masalah yang muncul yaitu perusahaan sering mengalami keterlambatan pada proses produksi akibat penjadwalan produksi yang kurang efektif sehingga menyebabkan beberapa stasiun kerja mendapatkan waktu menganggur yang cukup lama, dan hal tersebut disebabkan juga oleh sistem penjadwalan yang dirancang menggunakan metode *First Come First Serve* (FCFS) sehingga terjadi penumpukan *order* akibat banyaknya permintaan konsumen yang masuk.

c. Penggunaan Metode *Shortest Processing Time* (SPT)

Penelitian serupa juga dilakukan oleh Subroto (2019), Sia dan Widyadana (2013) yang membahas tentang perencanaan penjadwalan produksi yang optimal dan efektif agar bisa menentukan pekerjaan yang harus diprioritaskan terlebih dahulu. Kedua penelitian tersebut menggunakan metode *Shortest Processing Time* (SPT) dan dari penggunaan metode tersebut mampu meminimasi *flow time* pada sistem serta dapat menentukan pekerjaan mana yang harus dikerjakan terlebih dahulu agar penjadwalan produksi yang dihasilkan efektif dan optimal.

2.1.2. Penelitian Terdahulu Terkait Perancangan Sistem Informasi

Dalam menyelesaikan masalah yang berkaitan dengan ketidakpastian waktu penyelesaian *order*. Penentuan waktu penyelesaian *order* merupakan *input* dari berlangsung proses produksi dan hal tersebut berkaitan dengan penerimaan *order*, data *order*, negosiasi ke pelanggan dan lain sebagainya yang sudah terlaksana dan tersedia dengan baik atau belum. Penelitian mengenai perancangan sistem yang dapat digunakan yaitu berkaitan dengan sistem *order* yang kerap ditunjukkan dengan pembuatan sistem informasi yang terintegrasi melalui pengadaan *database*, dan proses transaksi yang mudah untuk terdata dan terproses oleh pekerja ataupun pelanggan.

a. Penggunaan metode SDLC

Penelitian mengenai perancangan sistem informasi pada suatu instansi atau UMKM sudah pernah dilakukan oleh beberapa peneliti. Dalam merancang sebuah

sistem informasi beberapa peneliti menggunakan metode dan sistem yang biasa digunakan untuk merancang sistem informasi seperti salah satunya yaitu metode SDLC (*System Development Life Cycle*) dengan model *Waterfall* untuk memperinci setiap tahapan yang diperlukan untuk merancang sistem informasi yang dibutuhkan. Metode SDLC dengan model *Waterfall* ini diterapkan pada Harsinta dan Nursari (2020) di Agna Advertising, Purwanto dan Mubarak (2020) di Yoshida, Yuliardi (2021) di PT. Advancenet Indonesia, dan Fadallah dan Rosyida (2018) di PT. Cahaya 99 Bekasi.

b. Penggunaan metode deskriptif (wawancara dan observasi)

Penelitian serupa juga dilakukan oleh Purnomo dkk. (2020) di Percetakan Mufiah Digital Printing, Akhtar dan Anubhakti (2018) di CV. Jet Printing, Ababil dkk. (2020) di PT. YES Communication, dan Nurgana (2017) di Rovell Digital Printing dengan menggunakan metode yang berbeda dengan penelitian sebelumnya, dalam penelitian-penelitian ini menggunakan metode deskriptif, pengumpulan data, dan identifikasi masalah yang dilakukan melalui wawancara, observasi, analisis data dan masalah, studi literatur dan dokumentasi. Pada beberapa penelitian tersebut juga menggunakan UML (*Unified Modelling Language*) yaitu dengan memodelkan sistem informasi yang akan dibuat dalam visual berupa diagram untuk mempermudah dalam merancang dan mengembangkan sistem informasi tersebut, biasanya dengan menggunakan *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan lain-lain. Selain itu, bahasa pemrograman yang sering digunakan yaitu PHP (*Hypertext Preprocessor*) yang biasa digunakan untuk merancang sebuah aplikasi web atau *mobile*, serta menggunakan MySQL sebagai *database* untuk sistem informasi yang dirancang.

Berdasarkan setiap penelitian mengenai perancangan sistem informasi tersebut, tujuan dari perancangan yang dilakukan yaitu untuk mempermudah sistem manajemen (penerimaan *order*, proses transaksi, dll) dalam perusahaan, serta dapat mempermudah dalam penyusunan jadwal produksi dalam suatu usaha dengan kemudahan akses dalam melacak transaksi dan kegiatan produksi yang sedang berjalan.

Keunikan dari penelitian yang dilakukan saat ini yang berkaitan dengan masalah ketidakpastian waktu penyelesaian *order* pelanggan tidak ditemukan pada penelitian lain. Masalah ketidakpastian dalam menentukan waktu penyelesaian *order* telah dijelaskan oleh karyawan *admin* dan terdapat beberapa *customer* yang

mengeluhkan masalah tersebut sebagaimana telah dijelaskan oleh mereka sebagai *stakeholders*. Perbedaan lokasi sejauh 50 meter antara lokasi penerimaan *order* dan lokasi produksi cetak *order* juga menjadi salah satu hal yang dapat mempengaruhi komunikasi antar karyawan mengenai data *order* pelanggan sehingga ketepatan informasi yang diberikan juga menjadi kurang informatif. Selain itu, penelitian ini baru pertama kali dilakukan pada Percetakan Al-Fath Offset dan pihak manajemen di sana juga mempermasalah terkait tidak dapat dilacaknya transaksi *order* karena tidak tersedianya data penerimaan *order* yang jelas sehingga hal tersebut menjadi keunikan masalah dari penelitian ini. Pada Tabel 2.1 dapat dilihat rekapitulasi tinjauan pustaka.



Tabel 2.1. Rekapitulasi Tinjauan Pustaka

No	Penulis	Objek Penelitian	Permasalahan	Solusi	Metode
1	Pradana dkk (2020)	Usaha Konveksi Nolabel Sablon Langsa	Keterlambatan dalam memenuhi target waktu penyelesaian pesanan pelanggan	Merancang sistem penjadwalan produksi untuk meminimalkan keterlambatan penyelesaian order	EDD (<i>Earliest Due Date</i>)
2	Yusuf dan Aryanny (2022)	CV. AM. Nanda Putra	Keterlambatan pada proses produksi karena penjadwalan proses produksi yang kurang efektif	Menghasilkan penjadwalan yang efektif dalam mengurangi waktu total produksi	<i>Campbell Dudek Smith</i>
3	Subroto (2019)	Industri Farmasi	Sering terjadi penundaan produksi, perubahan jadwal, dan keterlambatan dalam pengiriman produk	Melakukan pengurutan prioritas proses produksi sehingga proses pengelolaan data penjadwalan produksi menjadi lebih mudah.	EDD (<i>Earliest Due Date</i>) dan SPT (<i>Shortest Processing Time</i>)
4	Hamida dan Sugondo (2020)	PT XYZ	Terdapat keterlambatan dalam penjadwalan produksi	Mengurangi keterlambatan pekerjaan dengan membuat urutan prioritas pengerjaan order.	EDD (<i>Earliest Due Date</i>)
5	Sia dan Widyadana (2013)	PT Bondi Syad Mulia	Sering mengalami keterlambatan produksi karena belum adanya sistem penjadwalan produksi yang baik	Menghitung waktu proses produksi sehingga dapat meminimasi <i>flow time</i> pada sistem dengan menentukan pekerjaan mana yang harus diprioritaskan terlebih dahulu.	SPT (<i>Shortest Processing Time</i>)
7	Purnomo dkk (2020)	Percetakan Mufidah Digital Printing	Banyaknya permintaan pelanggan terhadap produk dan banyaknya pelanggan yang datang secara langsung untuk melakukan pemesanan, sehingga order menumpuk dan tidak terlayani dengan cepat.	Merancang sistem Informasi berbasis android pada objek lebih efektif dan efisien dan mempermudah dalam penerimaan order.	Perancangan Sistem Informasi menggunakan bahasa UML (<i>Unified Modelling Language</i>) dan menggunakan Android Studio, MySQL, dan PHP sebagai implementasinya.
8	Harsinta dan Nursari (2020)	Agna Advertising	Keterlambatan pengiriman produk kepada pelanggan, kesalahan total biaya pembayaran, masalah tumpang tindih pesanan.	Merancang sistem informasi yang dapat mempermudah perusahaan untuk berinteraksi dengan konsumen dan administrasi perusahaan dapat dilakukan dengan lebih mudah.	Pengembangan Sistem Informasi yang dilakukan berdasarkan pengolahan data berdasarkan SDLC (<i>System Development Life Cycle</i>) dan metode <i>Watterfall</i>
9	Purwanto dan Mubarak (2020)	Yoshida	Sistem order dan sistem perusahaan Yoshida secara keseluruhan belum terkomputerisasi dengan baik	Pengembangan teknologi informasi untuk penyimpanan data (<i>database</i>)	Metode Waterfall dengan melakukan pengembangan software <i>user</i> menggunakan sistem UML dan bahasa pemograman Laravel.

Tabel 2.1. Rekapitulasi Tinjauan Pustaka

No	Penulis	Objek Penelitian	Permasalahan	Solusi	Metode
10	Akhtar dan Anubhakti (2018)	CV. Jet Printing	Pemesanan masih menggunakan metode manual sehingga memperlambat transaksi penjualan, tidak terdapat laporan pengiriman dan pembayaran produk.	Pembuatan sistem informasi untuk penjualan sehingga proses menjadi lebih lancar dan adanya informasi terkait proses barang untuk pemilik	Metode pengumpulan data memiliki tahapan wawancara, observasi, dokumentasi, kuisisioner, dan studi literatur. Bahasa pemrograman yang digunakan adalah <i>Microsoft Visual Basic.Net 2008</i> dan <i>MySQL</i>
11	Ababil., dkk (2020)	PT. YES Communication	Tidak adanya sistem pencatatan data sehingga mempersulit dalam pembuatan laporan dan pencarian data	Menerapkan penggunaan aplikasi untuk pencatatan pemesanan percetakan menggunakan <i>database MySQL</i> sebagai penyimpanan data	Metode pengumpulan data untuk merancang suatu aplikasi berbasis data <i>MySQL</i> dan <i>websit</i>
12	Yulardi (2021)	PT. Advancednet Indonesia	Penerimaan pesanan serta penjadwalan barang masih dilakukan dengan cara manual menggunakan <i>microsoft excel</i> .	Membuat sistem informasi penerimaan pesanan untuk mempermudah proses penerimaan pesanan	Metode SDLC <i>waterfall</i> menggunakan Bahasa pemrograman yang digunakan adalah <i>PHP</i> dengan <i>database MySQL</i> dan <i>UML (Unified Modelling Language)</i>
13	Nurgana (2017)	Rovell Digital Printing	Ketidakpastian jadwal pada proses pengambilan barang dalam memberikan informasi pencetakan <i>order</i> produksi	Merancang sistem informasi untuk memudahkan pemesanan <i>order</i> dan tersedia fitur untuk pengecekan data pesanan	Pengembangan secara <i>object oriented</i> (berorientasi objek), menggunakan <i>UML</i> .
14	Fadallah dan Rosyida (2018)	PT. Cahaya 99 Bekasi	Proses pemesanan percetakan masih manual dan pelayanan pemesanan percetakannya tidak efektif dan efisien.	Membuat sistem terkomputerisasi untuk pemesanan percetakan sehingga lebih efektif dan efisien dalam mengolah data pesanan dan barang	SDLC <i>Waterfall</i> dengan teknik pengumpulan data melalui observasi, wawancara, dan studi pustaka

2.2. Dasar Teori

2.2.1. Perancangan

Menurut Reynaldi (2019), dikatakan bahwa perancangan merupakan suatu proses yang mempunyai tujuan untuk menganalisa, memperbaiki, dan menyusun suatu sistem fisik ataupun nonfisik agar dapat bekerja secara optimal dengan memanfaatkan setiap informasi yang tersedia. Perancangan juga merupakan suatu kegiatan yang tujuan utamanya untuk mendesain sistem baru yang mampu untuk menyelesaikan masalah-masalah yang dihadapi oleh suatu perusahaan atau industri melalui pemilihan alternatif sistem yang paling baik.

Dalam melakukan perancangan sistem, terdapat beberapa alat bantu yang digunakan yaitu sebagai berikut:

- a. *Flowchart* (Diagram Alir)
- b. *Data Flow Diagram* (DFD)
- c. Diagram Konteks
- d. *Entity Relationship Diagram*

2.2.2. Sistem Informasi

Menurut Nugraha dkk. (2018), sistem informasi dapat didefinisikan sebagai sebuah sistem yang terdapat di dalam suatu organisasi dan memiliki fungsi untuk mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi yang bersifat manajerial ataupun kegiatan strategi sehingga terdapat laporan-laporan yang dapat diakses oleh organisasi ataupun pihak luar.

Menurut Hutahean (2015), sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (*building block*) yaitu:

- a. Blok masukan (*input block*)

Blok masukan mempunyai tugas untuk melakukan *input* data ke dalam sistem informasi. *Input* yang dilakukan menggunakan metode ataupun media yang berfungsi untuk merekam data yang akan masuk ke dalam sistem.

- b. Blok model (*model block*)

Blok model terdiri dari prosedur, logika, dan model matematik. Kombinasi tersebut berfungsi untuk memproses data yang ter*input* ataupun tersimpan dalam dalam *database* sesuai dengan metode yang sudah terpilih untuk menghasilkan *output* sesuai dengan yang diinginkan.

c. Blok keluaran (*output block*)

Blok keluaran merupakan hasil *output* dari proses yang dilakukan oleh sistem informasi, berisikan informasi-informasi yang berguna untuk setiap level dalam organisasi serta pengguna sistem.

d. Blok teknologi (*technology block*)

Teknologi mempunyai fungsi untuk menerima *input*, memproses model, menyimpan dan mengakses data, menghasilkan dan mengirimkan *output* serta membantu pengaturan serta pengendalian dalam sistem secara menyeluruh. Terdapat tiga unsur utama yang membentuk teknologi yaitu teknisi (*human ware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

e. Blok basis data (*data base block*)

Blok basis data merupakan kumpulan data-data yang tersimpan di dalam dan saling terikat dalam sistem, serta untuk memproses dan data tersebut memerlukan bantuan dari *software*.

f. Blok kendali (*control block*)

Blok kendali menjadi tempat untuk melakukan pendalian terhadap faktor-faktor internal ataupun eksternal (bencana alam, suhu tinggi, debu, *error* pada sistem, sabotase, dan lain-lain) yang bisa merusak sistem. Pengendalian diperlukan untuk bisa mencegah kerusakan pada sistem, sehingga saat faktor-faktor yang bisa merusak sistem tersebut muncul maka dapat langsung diatasi dengan baik.

Dalam merancang sebuah sistem informasi diperlukan perancangan arsitektur terlebih dahulu agar dapat mendefinisikan fungsi dan tujuan dari sistem informasi. Menurut Prasetyo (2016), arsitektur sistem informasi diperlukan untuk membangun sebuah teknologi informasi dan menjadi pengarah suatu perusahaan. Hal ini bertujuan agar yang dibangun dapat membantu dalam mencapai tujuan dari suatu perusahaan. Arsitektur ini muncul karena adanya rancangan sistem informasi yang dibutuhkan. Pembuatan sistem informasi yang tidak didasarkan pada arsitektur teknologi informasi dapat menimbulkan adanya kesalahan sistem yang tidak sesuai dengan yang diinginkan. Dengan adanya arsitektur teknologi informasi ini dapat mengarahkan perusahaan dalam memiliki sistem informasi terintegrasi yang berguna untuk mencapai tujuan perusahaan.

Terdapat dua tahap arsitektur perancangan sistem informasi yaitu tahap *business architecture* yang melakukan proses analisis untuk menentukan kegiatan dan aktivitas yang terjadi dalam sebuah organisasi atau usaha. Tahap kedua yaitu

information system architecture yang terbagi menjadi dua proses yaitu proses perancangan arsitektur aplikasi dan arsitektur data (Atmajaya dkk, 2016). Perancangan arsitektur aplikasi ditujukan untuk menentukan desain aplikasi sistem informasi dan arsitektur data merupakan penggambaran atau visualisasi sistem yang dimodelkan melalui pembuatan diagram atau UML.

2.2.3. Metode SDLC (System Development Life Cycle)

System Development Life Cycle (SDLC) merupakan sebuah bentuk penggambaran atau visualisasi tahapan yang diperlukan untuk pembuatan atau pengembangan sebuah sistem (Silitonga dan Purba, 2021). SDCL mempunyai fungsi utama yaitu untuk mengakomodasi kebutuhan dari *client* atau pengguna berkaitan dengan sistem yang akan dikembangkan. Pengembangan sebuah sistem dapat berupa pembuatan aplikasi baru ataupun dengan memasukkan perubahan pada sistem tersebut.

Dalam menerapkan metode SDLC, terdapat beberapa tahapan yang perlu dilakukan untuk mengembangkan sebuah sistem yaitu antara lain:

a. Analisis Sistem

Tahapan ini merupakan proses mengidentifikasi dan menganalisis masalah dan kemungkinan solusi yang muncul untuk sistem dan proses pada organisasi.

b. Perancangan Sistem

Tahapan ini meliputi proses perancangan *input*, *output*, struktur, program, *hardware* dan *software* yang dibutuhkan untuk mendukung sistem.

c. Pembangunan dan *Testing* Sistem

Tahapan ini merupakan proses membangun *software* yang diperlukan untuk mendukung sebuah sistem, dan nantinya akan dilakukan *testing* untuk mengetahui kinerja dari *software* yang dibangun tersebut.

d. Implementasi Sistem

Tahapan ini merupakan proses menerapkan sistem yang baru atau dapat disebut juga peralihan dari sistem yang lama ke sistem yang baru, dan di tahap ini juga akan dilakukan *training* dan penjelasan mengenai modul sistem yang baru.

e. Operasi dan Perawatan Sistem

Tahapan ini dilakukan guna mendukung operasi dari sistem operasi yang sudah dibangun agar kinerjanya tetap terjaga, serta untuk mencari tahu celah-celah yang dapat dilakukan pengembangan ataupun perbaikan. Dalam penggunaan metode SDLC, terdapat beberapa model pengembangan yang dapat digunakan yaitu:

a. Model *Waterfall*

Waterfall merupakan salah satu dari berbagai model yang masuk ke metode SDLC. Metode dengan model ini mempunyai ciri khas yang menggambarkan pengerjaan setiap tahapan harus dikerjakan terlebih dahulu sebelum lanjut ke tahapan berikutnya agar di tiap tahapan tersebut dapat fokus dan maksimal dalam pengerjaannya. Terdapat beberapa keuntungan yang didapat dalam menggunakan metode SDLC model *waterfall* yaitu antara lain:

1. Proses pengerjaan dapat diurutkan dengan baik sehingga penggambaran dan pengerjaannya dapat lebih teratur dengan mengerjakan satu tahap ke tahap berikutnya.
2. Pengembang mendapatkan perencanaan dan persiapan yang lebih matang karena sudah terdapat urutan pengerjaan yang pasti.
3. Mampu untuk menentukan jadwal dari setiap proses, sehingga dapat dipantau juga untuk target penyelesaian pengembangan program.

b. Model *Prototype*

Prototype merupakan contoh model kedua yang masuk ke metode SDLC. Model ini merupakan pengembangan dari metode SDLC dengan model *waterfall*. Penggunaan metode dengan model ini mengikutsertakan *client* agar dapat memberi masukan sehingga perangkat lunak yang dihasilkan sesuai dengan yang *client* inginkan. Terdapat beberapa keuntungan dan kelemahan yang didapat dalam menggunakan metode SDLC model *prototype*, dengan keuntungannya yaitu *client*/pelanggan dapat berpartisipasi aktif dalam pengembangan sistem, waktu pengembangan sistem menjadi lebih cepat, dan pengembang sistem dapat menentukan kebutuhan dari *client*/pelanggan dengan lebih baik. Adapun kelemahan yang dimunculkan yaitu proses analisis dan perancangan sistem terlalu singkat waktunya, sistem yang dikembangkan menjadi kurang fleksibel, sistem yang dihasilkan tidak dipikirkan untuk jangka panjang.





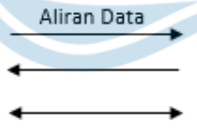



c. Model *Spiral*

Spiral merupakan contoh model ketiga yang masuk ke metode SDLC. Metode dengan model ini setiap tahapannya dapat dilaksanakan secara berulang-ulang sampai mendapatkan hasil sesuai yang diinginkan. Terdapat perulangan yang melalui 4 sektor yaitu analisa resiko, pengembangan validasi, perencanaan evaluasi, dan pembuatan tujuan. Model ini merupakan perbaikan dari kedua model yang terdapat dalam SDLC yaitu *waterfall* dan *prototype*.

2.2.4. Data Flow Diagram (DFD)

Menurut Muslihudin dan Oktafianto (2016), *Data Flow Diagram* (DFD) merupakan gambaran atau visualisasi dari sistem yang sedang dikembangkan atau sudah ada secara logika tanpa memperhitungkan atau memperhatikan lingkungan fisik yang memperlihatkan aliran data. Pada DFD terdapat level tertinggi yang disebut diagram konteks, dengan fungsinya yaitu untuk menggambarkan seluruh *input* dan *output* yang terjadi dalam sistem.

DFD ini sering digunakan sebagai *tools* untuk menerapkan *structured analysis* yaitu sebuah teknik untuk menganalisis sistem yang sedang dirancang ataupun sedang berjalan untuk melihat kebutuhan fungsional pada sistem tersebut (Sri Mulyani, 2017). DFD lebih menekankan penggambarannya pada aliran dan proses yang terjadi pada sistem secara menyeluruh, dan untuk menggambarannya yaitu menggunakan beberapa simbol yaitu seperti yang dapat dilihat pada Gambar 2.1.

Gane/Sarson	Yourdon/De Marco	Keterangan
		Entitas eksternal dapat berupa orang/unit terkait yang berinteraksi dengan sistem tetapi di luar sistem.
		Orang/unit yang mempergunakan atau melakukan transformasi data. Komponen fisik tidak diidentifikasi.
		Aliran data dengan arah khusus dari sumber ke tujuan.
		Penyimpanan data atau tempat data dilihat oleh proses.

Gambar 2.1. Simbol DFD

Berdasarkan Gambar 2.1 dapat diketahui terdapat empat simbol yang digunakan dalam merancang DFD yaitu:

a. *External Entity* (Entitas Eksternal)

External Entity digunakan untuk menggambarkan entitas luar sebagai sebuah elemen pada sistem, seperti orang (*user*) atau program lain.

b. *Process* (Proses)

Process digunakan untuk menggambarkan kegiatan yang dilakukan oleh *user* atau komputer dari hasil *input* data yang masuk kedalam proses, dan nantinya juga akan dihasilkan *output* dari proses.

c. *Data Flow* (Aliran Data)

Data Flow digunakan untuk menggambarkan aliran data dari tiap elemen pada sistem, seperti proses ke entitas ataupun sebaliknya yang digambarkan melalui anak panah.

d. *Data Store* (Data Penyimpanan)

Data Store digunakan untuk menggambarkan data penyimpanan atau biasa disebut *database* dalam sebuah sistem.

Dalam merancang arsitektur sistem menggunakan DFD, terdapat tiga level yang harus digambarkan agar sistem dapat diketahui secara menyeluruh yaitu:

1. Diagram Konteks (*Context Diagram*)

Diagram konteks menggambarkan seluruh proses yang terdapat pada sebuah sistem dan diagram ini biasa disebut sebagai DFD Level 0 yang menunjukkan tingkatan tertinggi dalam struktur DFD. Pada pembuatan diagram ini lebih berfokus pada entitas eksternal dan aliran datanya, dan tidak memuat mengenai penyimpanan data pada sistem.

2. Diagram level 1

Diagram level 1 atau DFD Level 1 ini menggambarkan secara rinci dari pembuatan diagram konteks sebelumnya, dengan memperlihatkan setiap proses dan sub-proses secara lebih detail.

3. Diagram level 2

DFD level 2 merupakan lanjutan dari visualisasi DFD level yang sebelumnya, dengan penggambaran pada tingkatan ini dibuat lebih detail pada setiap prosesnya. Biasanya dalam pembuatan arsitektur sistem menggunakan DFD ini hanya digambarkan sebatas level ini saja.

2.2.5. Entity Relationship Diagram (ERD)

Menurut Adiwijaya dkk. (2021), *Entity Relationship Diagram* (ERD) merupakan model konseptual yang menjabarkan hubungan atau relasi antara setiap entitas dan atribut. Selain itu, struktur data dan hubungan antar data dapat dimodelkan melalui ERD tanpa melihat proses yang dilakukan dari setiap aliran tersebut.

Menurut Afifah dkk. (2022), ERD merupakan tahap pertama dalam mendesain basis data yang sering digunakan dalam merancang suatu data. ERD ini merupakan salah satu bentuk diagram yang menunjukkan kebutuhan data dalam suatu basis data. Terdapat hal-hal yang harus diperhatikan dalam membuat ERD, yaitu masing-masing entitas yang ada harus berhubungan atau berelasi satu dengan yang lain. ERD ini juga sering menjadi penyebab kegagalan dalam membuat sebuah program atau *database*. Kegagalan yang terjadi dapat disebabkan oleh kesalahan prosedur dalam menyusun atau merancangan ERD itu sendiri.

Menurut Mannino dalam Afifah dkk. (2022), terdapat tiga elemen dasar yang ada pada ERD, yaitu entitas, atribut, dan relasi.

a. Entitas

Entitas merupakan objek yang berbentuk persegi panjang dan merupakan objek yang menjadi bagian utama atau perhatian dalam database. Entitas dapat berupa manusia, tempat, atau kondisi apapun yang berkaitan dengan data yang dibutuhkan.

b. Atribut

Atribut merupakan suatu informasi yang berbentuk elips dan biasanya terletak dalam tabel entitas ataupun dapat terpisah dari tabel entitas. Atribut ini harus memiliki *primary key* yang digunakan sebagai ciri khas atau identitas pada suatu entitas.

c. Relasi

Relasi merupakan hubungan antara dua atau lebih dari suatu entitas. Relasi yang ada pada ERD ini terdiri beberapa jenis, yaitu *one to one*, *one to many*, dan *many to many*. *One to one* adalah satu anggota entitas dapat berelasi antara satu dengan anggota entitas lainnya. *One to many* adalah relasi yang terjadi antara satu anggota identitas dengan beberapa anggota identitas lainnya. Lalu pada *many to many* adalah relasi yang terjadi pada beberapa anggota identitas dengan beberapa anggota identitas lainnya.

Menurut Togatorop (2021), untuk memodelkan ERD dapat dilakukan secara manual namun perancangan manual pada ERD ini membutuhkan waktu yang lebih lama. Menurut Simanjuntak dkk (2017), sebelum mengimplementasikan suatu basis data harus melakukan perancangan ERD. ERD merupakan diagram yang menggambarkan suatu entitas yang berkaitan pada suatu sistem yang

dirancang. Setelah ERD dirancang, ERD itu sendiri akan diterjemahkan menjadi *Conceptual Data Model* dan *Physical Data Model*. Pada data tersebut ERD akan diubah menjadi suatu objek yang dapat disebut atribut, tipe, *relationship*, maupun komponen lainnya. ERD yang baik adalah menghasilkan basis data dengan rekaman data yang akurat serta sesuai dengan kebutuhan sistem yang dirancang.

2.2.6. PHP

PHP merupakan singkatan dari PHP: *Hypertext Preprocessor*, yang merupakan sebuah singkatan rekursif atau dapat dikatakan kepanjangan dari kata tersebut terdiri dari singkatan itu sendiri. PHP merupakan pemrograman *script server-side* yang dapat disisipkan pada dokumen *HTML* untuk menyusun situs web yang mudah. Selain itu, penggunaan PHP ini juga tidak memerlukan biaya karena termasuk *software open-source* yang disebarluaskan secara gratis beserta lisensinya dengan mengunduh dari situs resminya.

Kemudahan dari penggunaan PHP ini ditunjukkan dengan banyaknya macam penggunaan *web server* yang dapat mendukung PHP seperti *apache*, *nginx*, DAN *Lightpd*, kemudian memiliki desain pengembangan yang mudah karena banyaknya *resource* dan forum diskusi dengan *developer*. PHP biasa digunakan untuk kebutuhan CGI (*Common Gateway Interface*) yang berarti memiliki kemampuan untuk membaca variabel dari *client* dan diproses ke *server* yang nantinya akan diolah dan menghasilkan *output* untuk *client*. Salah satu contohnya yaitu menampilkan isi dari *database* ke halaman situs web yang nantinya bisa diisi atau dihapus oleh *client*, beberapa *database* yang bisa digunakan untuk PHP yaitu *MySQL*, *Oracle*, dan *SQL Server*.

Dalam bahasa *HTML*, sintak PHP mempunyai posisi sebagai *tag* dengan penggunaan kodenya harus diberi awalan yaitu:

1. `<?php dan ?>`
2. `<? dan?>`
3. `<script language="php">` dan `</script>`
4. `<% dan %>`

Penggunaan cara pertama merupakan format yang dianjurkan untuk digunakan dalam melakukan penyusunan kode. Setiap *statement* (perintah) yang diberikan harus diakhiri dengan tanda titik-koma (;) agar perintahnya dapat dijalankan dan tidak mengalami *error*.

2.2.7. Basis Data / Database

Basis data merupakan kumpulan koleksi data-data yang saling mempunyai koneksi secara logika dengan isi dari data-data tersebut disusun dan didesain menyesuaikan dengan kebutuhan informasi dari sebuah organisasi atau perusahaan. Fungsi utama dari basis data yaitu untuk mengelompokkan dan menyimpan data secara efisien, selain itu terdapat beberapa fungsi yang dimiliki oleh basis data yaitu:

1. Ketersediaan (*availability*) : Melalui penggunaan dan penyimpanan datanya yang sistematis, dapat dipastikan bahwa basis data/database dapat tersedia di berbagai tempat dan waktu sesuai yang diinginkan.
2. Kecepatan (*speed*) : Penggunaan basis data dapat dilakukan dengan cepat dan tidak perlu membutuhkan waktu lama.
3. Kelengkapan (*completeness*): Basis data mampu untuk melakukan penyimpanan data sesuai kebutuhan dari tiap kategori data yang sudah disusun sebelumnya.
4. Akurasi (*accuracy*) : Pengguna dapat mengambil data yang terdapat dalam database sesuai dengan informasi yang dibutuhkan tanpa perlu takut akan terjadinya redundansi data atau duplikasi data secara berulang.
5. Keamanan (*security*) : Basis data dapat diatur dan dikelola agar tidak semua user mempunyai akses untuk mengubah atau menghapus data yang ada sehingga status keamanan lebih baik.

2.2.8. MySQL

MySQL merupakan sebuah *software* yang digunakan untuk sistem manajemen basis data/database SQL. *Software* ini dapat digunakan oleh semua orang secara bebas karena disebarakan secara gratis melalui lisensi GPL (*General Public License*). MySQL memiliki beberapa kelebihan yaitu kemampuannya untuk digunakan secara *multi-thread* sehingga beban kerja tidak bekerja pada satu inti saja tetapi bisa terbagi dalam beberapa *thread* dan bisa dijalankan secara parallel, selain itu dapat digunakan secara *multi-user* dan mampu menunjang sistem jaringan.

Terdapat beberapa fungsi yang digunakan dalam MySQL dan fungsi-fungsi tersebut berkaitan erat dengan *query SQL*. Dalam menggunakan perintah SQL pada *script PHP* dibutuhkan bantuan dari fungsi MySQL yang berguna sebagai perantara agar SQL dapat terkoneksi dan dijalankan pada server sehingga

hasilnya dapat dilihat oleh *client* atau *user*. Pada Tabel 2.2. dapat dilihat fungsi-fungsi *MySQL* yang biasa digunakan untuk penyusunan kode.

Tabel 2.2. Fungsi MySQL

Fungsi	Kegunaan
<code>mysql_connect</code>	Membuka koneksi dengan basis data/ <i>database MySQL</i>
<code>mysql_error()</code>	Menampilkan pesan yang menunjukkan kesalahan dalam bentuk teks
<code>mysql_close()</code>	Menutup koneksi dengan basis data/ <i>database MySQL</i>
<code>mysql_create_db()</code>	Membuat basis data/ <i>database</i> baru pada <i>MySQL</i>
<code>mysql_query()</code>	Melakukan perintah <i>query</i> pada <i>database MySQL</i> menggunakan koneksi yang sudah dibuat sebelumnya
<code>mysql_result()</code>	Mendapatkan hasil data dari <i>database MySQL</i>
<code>mysql_fetch_array()</code>	Menangkap data dari hasil perintah <i>query</i> dan mengubahnya ke dalam bentuk <i>array</i> asosiatif dan <i>array</i> numerik
<code>mysql_fetch_assoc()</code>	Menghasilkan <i>array</i> asosiatif yang <i>indexnya</i> menyesuaikan dengan nama kolom yang sudah dipilih
<code>mysql_fetch_object()</code>	Menangkap data dari hasil perintah <i>query</i> dan menampilkannya dalam bentuk <i>object oriented</i>
<code>mysql_fetch_row()</code>	Menangkap data dari hasil perintah <i>query</i> yang berbentuk <i>array</i> numerik saja

2.2.9. Visual Studio Code

Visual Studio Code (VS Code) merupakan sebuah aplikasi penyunting teks yang mudah digunakan dan ringan untuk dipasang di komputer. Aplikasi teks editor ini dikembangkan oleh *Microsoft* untuk berbagai sistem operasi, termasuk *Linux*, *Mac*, dan *Windows*. Aplikasi ini memiliki dukungan langsung untuk bahasa pemrograman seperti *JavaScript*, *Typescript*, dan *Node.js*, serta bahasa pemrograman lainnya melalui penggunaan *plugin* yang dapat diunduh melalui pasar aplikasi *Visual Studio Code* (seperti *C++*, *C#*, *Python*, *Go*, *Java*, dan lainnya).

Visual Studio Code menawarkan beragam fitur, seperti *intellisense*, *debugging*, dan kemampuan ekstensi yang meningkatkan kemampuan dari aplikasi teks editor. Fitur-fitur ini terus berkembang dengan setiap versi baru dari *Visual Studio Code*. Pembaruan untuk *Visual Studio Code* secara rutin dirilis setiap bulan, hal tersebut yang membedakannya dari penyunting teks lainnya.

Selain itu, *VS Code* bersifat *open source*, yang berarti para pengembang dapat melihat dan berkontribusi dalam pengembangannya. Kode sumber *VS Code* dapat diakses melalui tautan *GitHub*. Konsep ini menjadikan *VS Code* sebagai pilihan utama di kalangan pengembang aplikasi, karena mereka dapat turut serta dalam proses pengembangan aplikasi ini.

