

I. PENDAHULUAN

1.1 Latar Belakang

PT Darma Digital Solution adalah perusahaan dengan izin operasional sebagai Perseroan Swasta Nasional (PT/Perseroan Terbatas) yang mengkhususkan diri untuk bergerak dibidang Teknologi Informasi dan Komputer. PT Darma Digital Solution atau yang sering disingkat DDS ini memili kantor di gedung Darmapala, Jalan Pagar Alam No.61, Kecamatan Tanjung Karang Barat., Kota Bandar Lampung. Kegiatan usaha dan Layanan yang dijalankan oleh DDS saat ini dibagi menjadi dua (2) yaitu menjadi Penyediaan Jasa (Business Application Development, Web Development, Mobile Development, Corporate Branding, Networking dan Konsultan IT) dan Pengembangan produk (Sistem Manajemen Aset, Sistem Informasi Akademik, dll). Banyaknya kegiatan dan layanan yang dikembangkan oleh PT DDS ini mengharuskan para karyawan untuk melakukan rapat/pertemuan untuk membahas perencanaan serta perkembangan terkait proyek yang akan dilaksanakan.

Rapat merupakan bentuk komunikasi yang dihadiri beberapa orang dalam suatu organisasi untuk membahas atau memutuskan persoalan-persoalan. Membicarakan dan memecahkan permasalahan tertentu demi terciptanya tujuan bersama dan keberhasilan suatu organisasi. Menurut Yatimah (2009) “Rapat merupakan suatu alat komunikasi langsung antara pimpinan dan stafnya. Rapat memerlukan sekretaris yang terampil, yang merekam segala pembicaraan didalamnya. Tugas utama pimpinan di dalam rapat adalah mempersiapkan dan merekam serta melaporkan kegiatan rapat pimpinan”.

PT Darma Digital Solution saat ini masih melakukan penjadwalan secara manual untuk mengundang peserta rapat, mengatur lokasi, dan mengatur waktu rapat. Sistem penjadwalan tersebut dirasa kurang efektif dan tidak efisien karena apabila peserta yang diundang dalam rapat tersebut tidak mengetahui jika peserta diundang dalam rapat. Disisi lain untuk melakukan evaluasi dalam proses pengembangan sebuah proyek juga masih dilakukan secara manual, hal ini dinilai tidak efisien karena untuk melihat rekam jejak (*History*) dari rapat yang sudah

dilakukan menggunakan berkas cetak yang dapat hilang sehingga menghambat evaluasi dalam pengembangan sebuah proyek.

Berdasarkan permasalahan tersebut maka penulis memiliki ide untuk mengembangkan sebuah aplikasi yang dapat menjadwalkan pertemuan, mengundang peserta yang akan disertakan dalam rapat, dan membuat fitur *History*. Aplikasi ini dibangun menggunakan fitur *Push Notification* dan *Reminder* agar peserta rapat akan mendapatkan *notifikasi* pengingat, sehingga peserta dapat mengetahui bahwa dia diundang dalam agenda rapat. Selain itu, aplikasi ini juga ditambahkan fitur *History* untuk memudahkan dalam melakukan evaluasi dalam pengembangan sebuah proyek.

Aplikasi ini akan dibangun menggunakan metode *Rapid Application Development* (RAD), metode RAD dipilih karena pengembangan yang akan dilakukan akan menjadi lebih cepat sehingga waktu yang digunakan lebih efisien. Aplikasi yang akan dibangun berjudul "Aplikasi Penjadwalan pertemuan Berbasis Web Dengan Fitur *Push Notification* Dan *Reminder* Pada PT Darma Digital Solution Lampung".

Diharapkan dengan aplikasi yang dibangun ini dapat mempermudah dalam mengatur penjadwalan pertemuan yang akan dilaksanakan di PT Darma Digital Solution sehingga perkembangan dan perkembangan dari proyek dapat terpantau.

1.2 Tujuan

Tujuan dalam penyusunan tugas akhir ini adalah untuk menghasilkan Aplikasi Penjadwalan pertemuan Berbasis *Web* Dengan Fitur *Push Notification* Dan *Reminder* Pada PT Darma Digital Solution Lampung yang dapat mempermudah proses penjadwalan pertemuan dan pelaksanaan pertemuan.

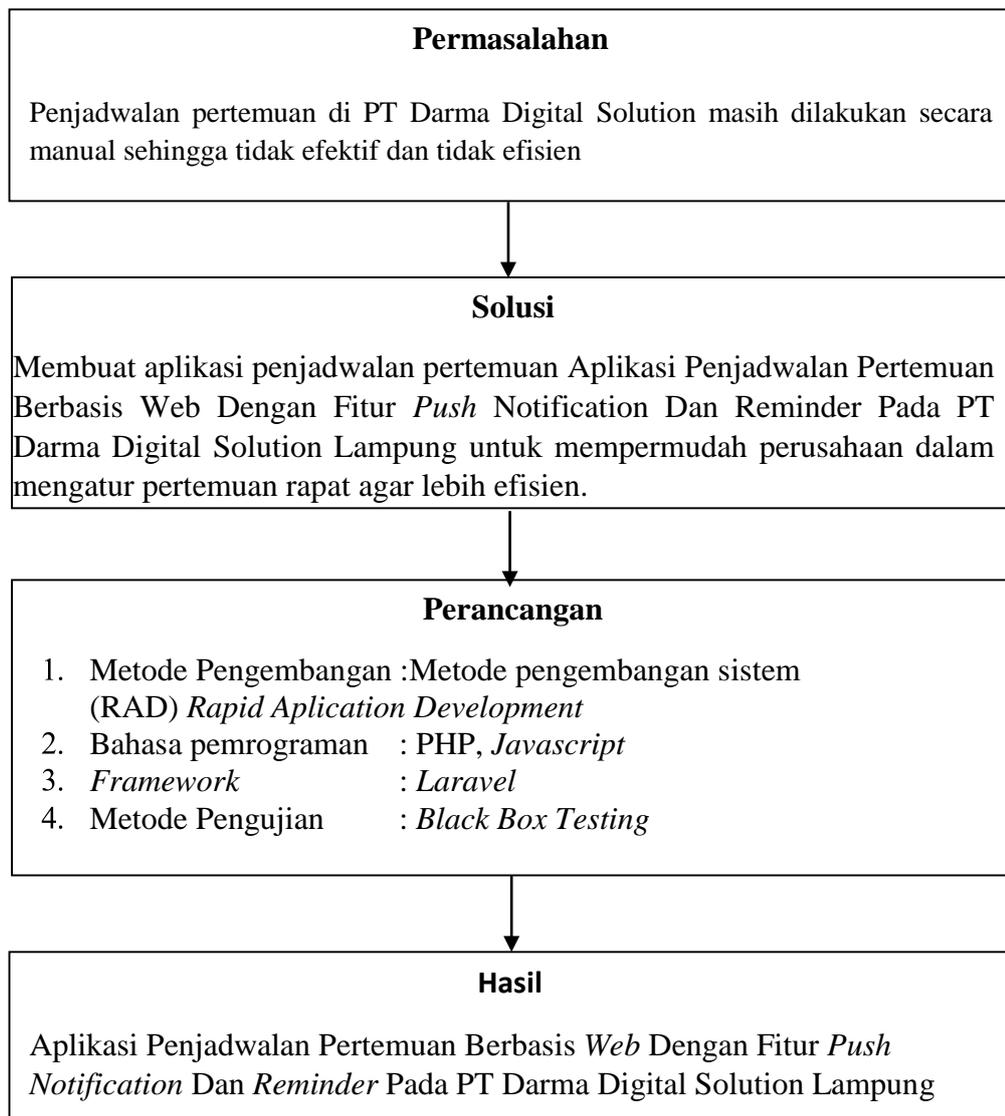
1.3 Kerangka Pemikiran

Sistem Penjadwalan pertemuan yang sedang berjalan secara optimal karena PT Darma Digital Solution (DDS) saat ini masih menjalankan secara manual hal ini dinilai kurang efisien dan jugaberdampak pada karyawan yang tidak mengetahui atau lupa apabila ada agenda pertemuan yang akan dilaksanakan.

Berdasarkan permasalahan di atas PT Darma Digital Solution membutuhkan sebuah aplikasi penjadwalan pertemuan untuk menanggapi permasalahan tersebut. "Aplikasi Penjadwalan pertemuan Berbasis Web Dengan Fitur *Push Notification*

Dan *Reminder* Pada PT Darma Digital Solution Lampung”. Dapat digunakan untuk mengatur waktu, tempat, serta jumlah karyawan yang akan diundang dalam agenda rapat tersebut.

Metode pengembangan yang akan digunakan dalam membangun ” Aplikasi Penjadwalan pertemuan Berbasis *Web* Dengan Fitur *Push Notification* Dan *Reminder* Pada PT Darma Digital Solution Lampung” yaitu metode RAD (*Rapid Application Development*). Bagan kerangka pemikiran akan disajikan dalam Gambar 1.



Gambar 1. Kerangka Pemikiran

1.4 Kontribusi

Pembuatan tugas akhir ini diharapkan dapat memberikan kontribusi pada beberapa pihak yang teribat, diantaranya adalah:

- 1) PT Darma Digital Solution
 - a. Mempermudah untuk melakukan penjadwalan pertemuan rapat.
 - b. Mempermudah dalam mengevaluasi proses pengembangan sebuah proyek.
- 2) Politeknik Negeri Lampung
 - a. Menyubangkan karya tulisan kepada Politeknik Negeri Lampung .

II. TINJAUAN PUSTAKA

2.1 Aplikasi

(Hasugian, 2014) Aplikasi berasal dari kata *application* yaitu bentuk benda dari kata kerja *to apply* yang dalam bahasa Indonesia berarti pengolah. Secara istilah, aplikasi komputer adalah suatu subkelas perangkat lunak komputer yang menggunakan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pemakai.

2.2 Karyawan

Karyawan merupakan kekayaan utama dalam suatu perusahaan, karena tanpa adanya keikutsertaan mereka, aktifitas perusahaan tidak akan terlaksana. Beberapa pengertian karyawan menurut para ahli. Menurut Hasibuan (dalam Karimah, 2012) karyawan adalah orang penjual jasa (pikiran atau tenaga) dan mendapat kompensasi yang besarnya telah ditetapkan terlebih dahulu. Menurut Subri (dalam Karimah, 2012) karyawan adalah penduduk dalam usia kerja (berusia 15-64 tahun) atau jumlah seluruh penduduk dalam suatu negara yang memproduksi barang dan jasa jika ada permintaan terhadap tenaga mereka.

2.3 Penjadwalan

(Pengertian Penjadwalan Menurut Para Ahli, n.d.) Jadwal menurut kamus besar bahasa Indonesia adalah pembagian waktu berdasarkan rencana pengaturan urutan kerja, daftar atau tabel kegiatan atau rencana kegiatan dengan pembagian waktu pelaksanaan yang terperinci. Sedangkan pengertian penjadwalan adalah proses, cara, perbuatan menjadwalkan atau memasukkan ke dalam jadwal. Menurut Chambers (1995) menyatakan bahwa jadwal didefinisikan sebagai sesuatu yang menjelaskan di mana dan kapan orang-orang dan sumber daya berada pada suatu waktu

2.3.1 *Push Notification*

Push notification termasuk ke dalam kategori dari model komunikasi *internet*, berdasarkan cara mempublikasikan/berlangganannya klien tidak harus menyetujui atau merequest dari *server* pusat untuk mendapatkan suatu informasi. Tidak seperti skenario tradisional (*pull*) dimana klien harus merequest setiap kali ingin mendapatkan informasi dari sistem. Secara umum, push notification dikirimkan melalui *Push Notification Service* (PNS) yang spesifik untuk setiap platform: seperti *Apple Push Notification Service* (APNs) untuk *Apple*, *Google Cloud Message* (GCM) untuk *Android*, dan *Microsoft Push Notification Service* (MPNS) untuk *Windows Phone*. Namun algoritma untuk tiap PNS sama. (Laysha 2015)

2.3.2 *Reminder*

Menurut Wahyu (2013) *Reminder* adalah sebuah pesan yang membantu seseorang mengingat sesuatu, *reminder* dapat lebih bermanfaat ketika informasi kontekstual digunakan untuk menyajikan informasi pada waktu yang tepat dan tempat yang tepat.

2.4 *Internet*

Menurut Sibero *Internet* adalah jaringan komputer yang menghubungkan antar jaringan secara global”. Berdasarkan penjelasan dari para ahli, dapat disimpulkan bahwa *Internet* adalah jaringan komputer yang terkoneksi dengan jaringan lain yang mempunyai cakupan luas untuk mendapatkan informasi dari jaringan tersebut.(Arizona, 2017).

Menurut Simarmata dalam Arizona (2017) menjelaskan bahwa “*Internet* adalah kelompok atau kumpulan dari jutaan komputer untuk mendapatkan informasi dari komputer yang ada didalam kelompok tersebut dengan asumsi bahwa pemilik komputer memberikan izin akses”. Berdasarkan dari kutipan diatas dapat disimpulkan bahwa *Internet* adalah *Internet* adalah jaringan komputer yang terkoneksi dengan jaringan lain yang mempunyai cakupan luas untuk mendapatkan informasi dari jaringan.

2.5 Website

Dalam dunia teknologi yang pesat ini diperlukan suatu jaringan yang bisa mempermudah serta mempercepat penyampaian informasi secara luas, dan dapat dengan mudah dan cepat oleh siapapun yang mendapatkan akses internet. Menurut Bekti (2015) menyimpulkan bahwa: *Website* merupakan kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

2.6 PHP

Hypertext Preprocessor (Firman dkk, 2016) adalah salah satu bahasa script yang dapat ditanamkan atau disisipkan ke dalam HTML dan PHP banyak digunakan untuk membuat program *web* dinamis. PHP juga bisa disebut dengan *scripting server-side*, yaitu pemrosesan datanya dilakukan pada sisi server.

2.7 Javascript

Javascript menurut (Sunyoto, 2007) adalah bahasa *scripting* yang populer di internet dan dapat bekerja di sebagian besar *browser* populer seperti *Internet Explorer (IE)*, *Mozilla Firefox*, *Netscape* dan *Opera*. Kode *Javascript* dapat disisipkan dalam halaman *web* menggunakan tag *SCRIPT*. Beberapa hal tentang *Javascript*:

1. *Javascript* didesain untuk menambah interaktif suatu *web*
2. *Javascript* merupakan sebuah bahasa *scripting*.
3. Bahasa *scripting* merupakan bahasa pemrograman yang ringan.
4. *Javascript* berisi baris kode yang dijalankan di komputer (*web browser*).
5. *Javascript* biasanya disisipkan (*embedded*) dalam halaman HTML.
6. *Javascript* adalah bahasa *interpreter* (yang berarti skrip dieksekusi tanpa proses kompilasi).
7. Setiap orang dapat menggunakan *Javascript* tanpa membayar lisensi.

2.8 CSS (*Cascading Style Sheet*)

Menurut Bunafit Nugroho (2014), *Cascading Style Sheet* adalah bahasa *style sheet* yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa *markup*. CSS bekerja sebagai pelengkap pada elemen HTML yang kesemuanya itu dapat dikendalikan dengan menggunakan dengan menggunakan sebuah bahasa *script* CSS. Penggunaan CSS dilakukan untuk memperluas kemampuan HTML dalam memformat dokumen *web* atau untuk memperindah tampilan *web*. Penulisan kode CSS disisipkan pada tag HTML. Kode CSS ditulis dengan tag `<style>` dan `</style>` dengan mendefinisikan suatu *style* baru yang kemudian dapat digunakan berulang kali

2.9 *Framework*

Menurut Hakim (2010) menjelaskan bahwa, *Framework* adalah koleksi atau kumpulan potongan-potongan program yang disusun atau diorganisasikan sedemikian rupa, sehingga dapat digunakan untuk membantu membuat aplikasi utuh tanpa harus membuat semua kodenya dari awal.

Sedangkan menurut Raharjo (2015), *Framework* adalah suatu kumpulan kode berupa pustaka (*library*) dan alat (*tool*) yang dipadukan sedemikian rupa menjadi satu kerangka kerja (*framework*) guna memudahkan dan mempercepat proses pengembangan aplikasi *web*. Jadi, *Framework* adalah kumpulan-kumpulan potongan program yang dipadukan menjadi satu kerangka kerja yang digunakan untuk membantu dalam pembuatan sebuah aplikasi.

2.10 *Laravel*

Menurut Susanto dkk (2015), *laravel* adalah salah satu *web application framework* yang bersifat *open source*. *Framework* ini berjalan diatas PHP 5 dan berbasis MVC (*Model View Controller*). *Laravel* pertama kali dirilis pada 22 Februari 2012. *Model View Controller* adalah sebuah metode untuk membuat sebuah aplikasi dengan memisahkan data (*Model*) dari tampilan (*View*) dan cara bagaimana memprosesnya (*Controller*). Terdapat tiga komponen pembangun suatu MVC yaitu:

a) Model

Model adalah bagian yang berperan menghubungkan controller dengan *database*. Tugas dari model adalah melakukan manipulasi data ke *database* seperti CRUD (*create, read, update, delete*).

b) View

View mengatur bagaimana data akan ditampilkan kepada user. Data yang didapat dari model akan diproses oleh *controller* kemudian oleh *View* ditampilkan kepada user.

c) Controller

Controller merupakan bagian yang menjadi penghubung antara model dan *view*. *Controller* berfungsi memroses fungsi atau perintah dari *user* kemudian menentukan bagaimana aplikasi dijalankan.

2.11 Bootstrap

Menurut Atalas Husein, (2013) *Bootstrap* merupakan *framework* untuk membangun desain *web* secara responsif. Artinya, tampilan *web* yang dibuat oleh *bootstrap* akan menyesuaikan ukuran *layer* dan *browser* yang kita gunakan baik di *desktop, tablet* ataupun *mobile device*. Dengan *bootstrap* kita juga bisa membangun *web* dinamis ataupun statis.

2.12 XAMPP

Menurut Riyanto (2015), XAMPP adalah sebuah *software* yang berfungsi untuk menjalankan *website* berbasis PHP dan menggunakan pengolah data MySQL di komputer lokal. XAMPP berperan sebagai *server web* pada komputer Anda. XAMPP juga dapat disebut sebuah *CPanel server* virtual, yang dapat membantu Anda melakukan *preview* sehingga dapat memodifikasi *website* tanpa harus online atau terakses dengan internet.

2.13 Database

Menurut Abdulloh (2018), *Database* atau basis data adalah kumpulan informasi yang disimpan dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi.

Menurut Enterprise (2017), *Database* adalah suatu aplikasi yang menyimpan sekumpulan data. Setiap database mempunyai perintah tertentu untuk

membuat, mengakses, mengatur, mencari, dan menyalin data yang ada di dalamnya.

2.13.1 Tabel

Tabel adalah Sekumpulan data terstruktur terdiri dari baris dan kolom yang disimpan pada suatu media penyimpanan dimana data tersebut dapat dimanipulasi (tambah, ubah, hapus) dan dapat dilihat dengan menggunakan teknik tertentu untuk menghasilkan informasi yang lebih bermakna.

Suatu tabel ialah koleksi data berhubungan disimpan pada format tabel dalam suatu pangkalan data yang terdiri akan kolom dan baris.

2.13.2 Relasi Tabel

Secara umum relasi adalah hubungan antara tabel yang mempresentasikan hubungan antar objek di dunia nyata. Relasi merupakan hubungan yang terjadi pada suatu tabel dengan lainnya yang mempresentasikan hubungan antar objek di dunia nyata dan berfungsi untuk mengatur mengatur operasi suatu database.

Pada sebuah database, relasi dihubungkan dengan cara memberikan satu kolom dengan value yang sama dengan tabel yang berhubungan, ini disebut *foreign key*. *Foreign key* sendiri merupakan sebuah kolom yang hanya bisa menyimpan data yang sama dengan *primary key* yang berhubungan dengan tabel tersebut. Artinya *foreign key* hanya bisa diisi dengan data yang sudah ada pada *primary key*.

Relasi mencakup tiga macam. Yaitu :

a. Relasi *one to one*

adalah relasi dimana setiap satu baris data pada tabel satu hanya berhubungan dengan satu baris data di tabel dua. Artinya masing - masing hanya memiliki satu hubungan saja.

b. Relasi *one to many*

Relasi *one to many* adalah relasi yang mana setiap baris dari tabel pertama dapat dihubungkan dengan satu baris ataupun lebih dari tabel kedua. Artinya satu baris dari tabel pertama dapat mencangkup banyak data pada tabel kedua.

c. Relasi *many to many*

Relasi *many to many* adalah keadaan dimana satu baris dari tabel satu dapat berhubungan dengan tabel kedua. Dan satu baris dari tabel kedua dapat berhubungan dengan banyak baris dari tabel pertama. Artinya kedua tabel masing - masing dapat mengakses banyak data dari tiap tabel yang lain. Dalam hal ini, kita membutuhkan tabel ketiga sebagai perantara tabel satu dan tabel dua sebagai tempat untuk menyimpan *foreign key* dari masing - masing tabel.

2.13.3 Record

Record (disebut juga struktur, kumpulan data) merupakan kumpulan dari elemen-elemen data yang terkait dalam sebuah basis data. Secara ringkas, basis data dapat dikatakan sebagai sebuah tabel yang memiliki baris alias *record* dan kolom atau *field*. Setiap baris menyatakan elemen-elemen data yang saling berkaitan. Sebagai contoh dalam suatu tabel memiliki kolom nama, alamat, tanggal lahir, dan pekerjaan. Maka satu *record* adalah data satu orang yang terdiri atas nama, alamat, tanggal lahir dan pekerjaan.

2.14 Sistem Basis Data

Waliyanto, dalam Muiz (2007) mengatakan bahwa yang dimaksud dengan sistem basis data merupakan suatu gabungan dan juga perpaduan antara basis data (*database*) dengan suatu sistem manajemen basis data (SMBD) atau yang juga lebih sering dikenal dengan istilah DBMS (*Database Management System*).

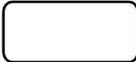
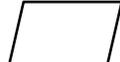
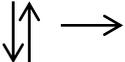
2.15 MySQL

Menurut Riyanto (2015), MySQL adalah RDBMS yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*), di mana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat *close source* atau komersial. Karena sifatnya yang *Open Source*, sehingga komunitas umum turut mengembangkan mesin basis data MySQL dan hal ini menyebabkan kemampuan dan performanya berkembang dengan pesat.

2.16 Flowchart

Flowchart merupakan prosedur yang terstruktur yang dapat dipahami oleh orang lain dalam pemecahan suatu masalah yang dikaitkan dengan simbol-simbol tertentu (Sitorus, 2015). Simbol-simbol *flowchart* dapat dilihat pada Tabel 1.

Tabel 1. Simbol-Simbol *Flowchart*

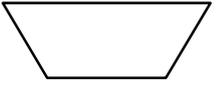
SIMBOL	NAMA	Keterangan
	Proses	Menyatakan suatu proses sistem.
	<i>Decision</i>	Menyatakan suatu kondisi yang menghasilkan dua kemungkinan.
	Terminal	Menyatakan awal dan akhir suatu program.
	<i>Input/ Output</i>	Menyatakan input dan output.
	<i>Punched Card</i>	Menyatakan <i>input</i> berasal dari kartu atau <i>output</i> berasal ditulis ke kartu.
	<i>Puch Tape</i>	Menyatakan <i>input</i> atau <i>output</i> yang menggunakan pita kertas berlubang.
	<i>Connector</i>	Digunakan untuk menghubungkan dari proses ke proses lainnya dalam halaman yang sama.
	<i>Offline Connector</i>	Digunakan untuk menghubungkan dari proses ke proses lainnya dalam halaman yang berbeda.
	<i>Flow</i>	Menyatakan aliran arus suatu proses.

Sumber : (Santoso dan Nurmalina, 2017)

2.17 Mapping Chart

Mapping *Chart* merupakan penggambaran secara grafik dari langkah-langkah dan urutan-urutan prosedur dari suatu program. *Mapping Chart* efektif digunakan untuk menggambarkan proses maupun prosedur dalam sebuah organisasi. *Mapping Chart* digunakan untuk menggambarkan urutan prosedur/proses kerja dalam pembuatan sistem (Widarma & Rahayu, 2017). Simbol-simbol dalam *mapping chart* disajikan pada Tabel 2.

Tabel 2. Simbol-Simbol *Mapping Chart*

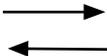
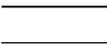
Nama	Simbol	Keterangan
<i>Keyboard</i>		Menyatakan <i>inputan</i> yang menggunakan keyboard yang terkomputerisasi.
<i>Manual Activity</i>		Menyatakan proses <i>input</i> atau output tanpa tergantung jenis peralatannya
<i>Database</i>		Menunjukkan penyimpanan data yang terkomputerisasi.
<i>Connector</i>		Menyatakan sambungan dari satu proses ke proses lain dalam halaman yang sama.
<i>Document</i>		Data yang berbentuk informasi, bisa berbentuk dalam softcopy.
<i>Process</i>		Menunjukkan proses yang dilakukan sebuah fungsi.

Sumber: Widjojo (2014)

2.18 DFD

DFD (*Data Flow Diagram*) adalah suatu metode dalam membuat sebuah perancangan sistem yang berorientasi pada alur data yang bergerak kesebuah sistem lainnya. Dalam membuat Sistem Informasi ini, DFD sering dipakai. DFD dibuat oleh para analis untuk membuat sebuah sistem dengan baik. Dimana DFD ini nantinya dikasihkan kepada para *programmer* untuk memulai proses *coding*. *Programmer* melakukan coding sesuai dengan DFD yang dibuat oleh para analis sebelumnya. Berikut ini adalah simbol-simbol DFD (*Data Flow Diagram*) yang disajikan pada Tabel 3.

Tabel 3. Simbol-Simbol DFD (*Data Flow Diagram*)

Nama	Simbol	Keterangan
Eksternal <i>Entity</i>		Dapat berupa orang, organisasi, atau unit terkait yang dapat berinteraksi dengan sistem tetapi diluar sistem.
<i>Process</i>		Proses yang dilakukan oleh orang atau unit yang mempergunakan transformasi data atau mentransformasikan data.
<i>Data Flow</i>		Data <i>Flow</i> merupakan aliran data pada sistem yang menunjukkan arah khusus dari sumber ke tujuan.
<i>Data Store</i>		Data <i>Store</i> berfungsi untuk menyimpan pada data <i>database</i> yang biasanya berupa <i>Eksternal Entity</i> tabel dan tempat data yang diserahkan oleh proses.

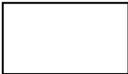
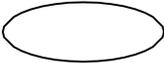
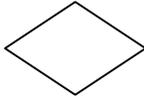
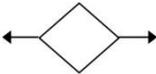
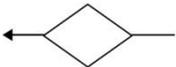
Sumber: Sitorus, (2015).

2.19 ERD

Menurut Brady dan Loonam (2010), *Entity Relationship* diagram (ERD) merupakan teknik yang digunakan untuk memodelkan kebutuhan data dari suatu organisasi, biasanya oleh *System Analysts* dalam tahap analisis persyaratan proyek pengembangan *system*.

Di bawah ini merupakan simbol-simbol dalam membuat suatu ERD.

Tabel 4. Simbol-simbol ERD

Nama	Simbol	Keterangan
Entitas		Individu yang mewakili suatu objek dan dapat dibedakan dengan objek yang lain.
Atribut		Properti yang dimiliki oleh suatu entitas dimana dapat mendeskripsikan karakteristik dari entitas tersebut.
Relasi		Menunjukkan hubungan diantara sejumlah entitas yang berbeda
Relasi 1 : 1		Relasi yang menunjukkan bahwa setiap entitas pada himpunan entitas pertama berhubungan dengan paling banyak satu entitas pada himpunan entitas kedua.
Relasi 1 : N		Relasi yang menunjukkan bahwa hubungan antara entitas pertama dengan entitas kedua adalah satu banding banyak atau sebaliknya. Setiap entitas dapat berelasi dengan banyak entitas pada himpunan entitas yang lain.

Sumber: Edi & Betshani, (2009).

2.20 Visual Studio Code

Menurut Agustini dkk (2019) Visual Studio Code adalah kode editor sumber yang dikembangkan oleh Microsoft untuk Windows, Linux dan macOS. Ini termasuk dukungan untuk debugging, kontrol git yang tertanam dan GitHub, penyorotan sintaksis, penyelesaian kode cerdas, *snippet*, dan *refactoring* kode. Ini sangat dapat disesuaikan, memungkinkan pengguna untuk mengubah tema, pintasan *keyboard*, preferensi, dan menginstal ekstensi yang menambah fungsionalitas tambahan.

2.21 Rapid Application Development (RAD)

Menurut Tilley & Rosenblatt (2017) *Rapid Application Development* (RAD) merupakan metode dalam mengembangkan perangkat lunak dalam waktu yang relatif singkat. Waktu pengembangan yang pada umumnya 180 hari namun dengan menggunakan metode ini dapat diselesaikan dalam waktu 30-90 hari. Menghemat waktu pengembangan, meningkatkan keberhasilan serta menghemat biaya produksi. Adapun 4 tahapan pengembangan perangkat lunak menggunakan RAD sebagai berikut:

1. *Requirements Planning*

Tahapan ini adalah kegiatan perencanaan sistem dan sistem analisis, mendiskusikan dan menyetujui kebutuhan bisnis, ruang lingkup proyek, dan kendala hingga dapat diambil kesimpulan masalah utama sistem yang akan dibuat.

2. *User Design*

Setelah melalui tahapan perencanaan pengguna dan analisis sistem saling berinteraksi untuk menggambarkan model dan *prototipe* yang mewakili proses, input serta output sistem.

3. *Construction*

Tahapan ini merupakan tahap pembuatan program dan aplikasi seperti perbaikan dan perubahan dari aplikasi yang sedang dikerjakan.

4. *Cutover*

Tahap akhir disebut *cutover*, yaitu berupa implementasi seperti konversi data pengujian program atau aplikasi yang selesai dibuat.

2.22 Blackbox Testing

Menurut Rosa dan Salahuddin (2015) “*Blackbox testing* yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program”. Pengujian ini berusaha menemukan beberapa kesalahan dalam kategori, fungsi-fungsi yang tidak benar, kesalahan *interface*, kesalahan dalam struktur data atau *database*, dan kesalahan kinerja.