

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Secara umum pengertian penelitian terdahulu adalah sumber lampau dari hasil penelitian yang nantinya diusahakan oleh peneliti untuk membandingkan penelitian yang akan dilaksanakan. Penelitian terdahulu juga bisa berfungsi sebagai sumber inspirasi yang nantinya akan membantu pelaksanaan penelitian. Selain itu, peneliti juga bisa memeriksa apa yang kurang dan kelebihan untuk dikembangkan, Sehingga ilmuwan juga bisa membuat sebuah penelitian yang orisinal/baru karena tahu mana yang sudah ditemukan dan mana yang belum.

Penelitian yang pertama adalah Penelitian yang dilakukan (Sopandi, dkk., 2019) dalam jurnal yang berjudul Rancang Bangun Sistem Informasi Pendaftaran Pasien Berbasis *Web* Pada Klinik Johar Kabupaten Karawang, memiliki permasalahan Pendaftaran pasien secara manual dapat mengakibatkan lambatnya proses pengelolaan data yang manual berdampak kepada kurang baiknya pelayanan terhadap pendaftaran pasien. Data pendaftaran pasien akan banyak yang rusak dan sulit dicari jika dikelola secara manual. Dengan adanya Aplikasi Sistem Informasi Pendaftaran Pasien ini dapat menjadi salah satu alternatif pemecahan masalah tersebut. Metode yang digunakan pada pengembangan perangkat lunak ini menggunakan model *Waterfall*. Pada implementasinya, Sistem Informasi Pendaftaran Pasien yang telah dibuat pasien dapat melakukan pendaftaran pasien, melihat data pasien, cek token dan cek hasil pemeriksaan, melihat data dokter yang siap melayani pengobatan, melihat halaman hubungi kami. Admin dapat melakukan *login* di halaman admin mengelola data pasien, mengelola data dokter, mengelola data spesialis, melihat kotak pesan. Dokter dapat melakukan *login* di halaman dokter, mengelola data pasien, *mengentry* hasil pemeriksaan. Berdasarkan pengujian yang dilakukan, Sistem Informasi Pendaftaran Pasien yang dibuat berhasil dijalankan dengan baik.

Penelitian kedua yang dilakukan (Sihombing & Irawan, 2019) dalam jurnal yang berjudul Perancangan Sistem Informasi Pendaftaran Pasien Menggunakan

VB.Net Pada Klinik Pratama Rbg Rz Bandung, memiliki permasalahan dalam pelayanan pasien rawat jalan, pasien rawat inap dan ruang rawat inap. Hampir dalam segala kegiatan di atas, pengolahan datanya masih dilakukan secara manual. Hal tersebut mengakibatkan keterlambatan dalam pembuatan laporan dan pencarian data tertentu yang membutuhkan waktu yang relatif cukup lama, termasuk juga dalam hal pengolahan data pendaftaran pasien. Dengan adanya Sistem Informasi Pendaftaran Pasien dapat menjadi salah satu alternatif pemecahan masalah tersebut. Sistem informasi pendaftaran pasien berbasis desktop dengan menggunakan metode *Waterfall*. Sedangkan tools untuk perancangan sistem adalah metode/teknik yang disebut dengan *Unified Modelling Language (UML)*. Aplikasi dibuat dengan menggunakan VB.Net dengan *editor* berikut *compiler* nya Visual Studio 2013 dan *desktop My SQL* Metode yang digunakan peneliti untuk pengembangan perangkat lunak adalah dengan metode *Rational Unified Process (RUP)*, yaitu perancangan perangkat lunak yang dilakukan secara kontinu/itersi, dengan penekanan yang lebih ke arah pemecahan kasus (*use case driven*). Pada implementasinya, Sistem Informasi Pendaftaran Pasien yang telah dibuat petugas administrasi dapat menginput data pembayaran, maka pasien akan membayar biaya pemeriksaan berikut biaya pembelian obat. Petugas tersebut akan membuat struk bukti pembayaran. Berdasarkan pengujian yang dilakukan, Sistem Informasi Pendaftaran Pasien yang dibuat berhasil dijalankan dengan baik.

Penelitian ketiga yang dilakukan oleh (Zahratonnisa, dkk., 2022), dalam jurnal yang berjudul Aplikasi pendaftaran pasien secara *online* dan sistem pelayanan administrasi kesehatan pada puskesmas tapin utara, Kesehatan merupakan hal penting yang harus diperhatikan oleh setiap orang, karena kesehatan merupakan keadaan sejahtera dari badan, jiwa, dan sosial yang memungkinkan setiap orang hidup produktif secara sosial dan ekonomi. Kesehatan menjadi salah satu fokus pemerintah yang bertujuan untuk meningkatkan kesejahteraan masyarakat dibidang kesehatan dan salah satunya dengan program dibangunnya Pusat Kesehatan Masyarakat (Puskesmas) di setiap kecamatan. Dalam proses kerjanya, Puskesmas pasti melakukan pencatatan-pencatatan seperti pendaftaran pasien, daftar obat, laporan jumlah pasien per bulan, dan lainnya. Tetapi di beberapa Puskesmas

termasuk di Puskesmas Tapin Utara, pendataan pendaftaran pasien baru dan pasien lama, data pelayanan, data pemeriksaan dan lain-lainnya masih dalam bentuk tertulis sehingga memperlambat petugas puskesmas dalam mencari berkas-berkas pasien yang pernah melakukan pelayanan kesehatan pada Puskesmas Tapin Utara. Solusi dari permasalahan tersebut yaitu dengan membangun sebuah aplikasi pendaftaran pasien secara *online* serta pelayanan administrasi kesehatan pada Puskesmas Tapin Utara yang diharapkan dapat membantu petugas serta pasien dalam melakukan pelayanan kesehatan pada Puskesmas Tapin Utara. Pembuatan aplikasi ini menggunakan metode *waterfall* serta bahasa pemrograman PHP dengan menggunakan *framework CodeIgniter*, menggunakan MySQL sebagai *database* atau tempat penyimpanan data. Berdasarkan latar belakang permasalahan tersebut maka penulis mengangkat judul "Aplikasi Pendaftaran Pasien Secara *Online* Dan Sistem Pelayanan Administrasi Kesehatan Pada Puskesmas Tapin Utara".

Penelitian keempat yang dilakukan oleh (Rochman, dkk., 2018) dalam jurnal yang berjudul Perancangan Sistem Informasi Profil dan Pendaftaran Pasien Rawat Jalan Pada RSIA Pratiwi, memiliki permasalahan dalam pelayanan administrasi pendaftaran khususnya pendaftaran pasien rawat jalan masih bersifat manual yaitu pasien datang ke rumah sakit mengambil no antrian di bagian pendaftaran untuk mendaftar ke poliklinik yang dituju serta informasi-informasi yang berkaitan dengan pelayanan rumah sakit masih bersifat manual yaitu pengumuman atau informasi di tempel di madding rumah sakit. Dengan adanya Sistem Informasi Profil dan Pendaftaran Pasien dapat menjadi salah satu alternatif pemecahan masalah tersebut. Aplikasi dibuat dengan menggunakan bahasa pemrograman PHP dan basis data MySQL Metodologi dalam pembuatan aplikasi ini antara lain metode pengumpulan data yang diantaranya metode wawancara, metode observasi dan metode pustaka. Pada implementasinya, Sistem Informasi Profil dan Pendaftaran Pasien yang telah dibuat pasien dapat membuka *web* rumah sakit ibu dan anak pratiwi, sistem menampilkan sistem menampilkan halaman beranda dan pilihan daftar *online* jika memilih daftar *online* dan pasien input data lalu klik *button* daftar dan sistem menampilkan bukti pendaftaran *online*. Setelah pasien berhasil melakukan daftar secara *online*, sistem akan menampilkan bukti daftar *online*

selanjutnya pasien mencetak bukti daftar online untuk kemudian di bawa ke rumah sakit untuk melakukan pendaftaran di bagian pendaftaran sesuai dengan poliklinik atau dokter yang dituju. Untuk manajer setelah melakukan *login* sistem akan menampilkan menu utama selanjutnya manajer memilih daftar pasien kunjungan per hari selanjutnya sistem menampilkan jumlah pasien yang melakukan daftar online. Berdasarkan pengujian yang dilakukan, Sistem Informasi Profil dan Pendaftaran Pasien yang dibuat berhasil dijalankan dengan baik.

Penelitian kelima yang dilakukan oleh (Eko, dkk., 2021) dalam jurnal yang berjudul Aplikasi Sistem Informasi Puskesmas Bagian Registrasi Pendaftaran Pasien, memiliki permasalahan pencatatan data di puskesmas yang manual akan menghabiskan waktu untuk menyelesaikannya, serta ketika pasien lupa membawa kartu berobat diharuskan menunggu lama guna memperoleh berkasnya alhasil memerlukan waktu yang agak lama ketika proses pelayanan pendaftaran pasien. Jika berkasnya tidak dapat ditemukan, maka otomatis akan dibuatkan berkas yang baru sehingga menghasilkan data ganda (*double*). Kebanyakan pasien lupa membawa kartu berobat ketika mau berobat. Selain itu juga, dalam pembuatan laporan bulanan, petugas harus mencatat satu persatu data yang akan dijadikan laporan bulanan. Dengan adanya Aplikasi Sistem Informasi Puskesmas Bagian Registrasi Pendaftaran Pasien dapat menjadi salah satu alternatif pemecahan masalah tersebut. Aplikasi dibuat dengan menggunakan bahasa pemrograman PHP dan basis data MySQL, juga di bangun dengan *Admin Template Starbootstrap 2* juga dengan menggunakan *Framework Codeigniter*. Metode penggunaan model Waterfall atau Waterfall model untuk pengembangan sistem atau perangkat lunak. Pada implementasinya, Sistem Informasi Puskesmas Bagian Registrasi Pendaftaran Pasien proses alur pendaftaran pasien dimulai oleh pasien yang dapat langsung mendaftar lewat *website* setelahnya pasien diminta untuk verifikasi data pendaftaran setelahnya maka akan dapat mencetak nomor antrian. Pasien juga dapat mencetak kartu berobat serta pasien dapat melakukan *update*, *delete*, serta mencari data pasien. Setelah mendaftar pasien datang ke puskesmas tunjukan bukti pendaftaran ke petugas untuk di verifikasi, kemudian pasien tinggal menunggu antrian di ruang poli yang sesuai pendaftaran pasien. Berdasarkan pengujian yang

dilakukan, Sistem Informasi Puskesmas Bagian Registrasi Pendaftaran Pasien yang dibuat berhasil dijalankan dengan baik.

Tabel 2.1 Perbandingan Penelitian Terdahulu Dengan Penelitian Sekarang

No	Penelitian	Persamaan	Perbedaan
1	Sopandi, dkk., 2019. Rancang Bangun Sistem Informasi Pendaftaran Pasien Berbasis <i>Web</i> Pada Klinik Johar Kabupaten Karawang.	1. Bertujuan untuk membantu permasalahan pendaftaran pasien puskesmas yang masih manual. 2. Menggunakan metode pengembangan sistem model <i>Waterfall</i> .	1. Aplikasi sudah bisa digunakan dipuskesmas yang berkaitan. 2. Terdapat fitur limit berobat disetiap poli yang dapat diubah, pembayaran online dan rekap laporan pasien yang bisa dicek baik perhari, perminggu, perbulan maupun pertahun kemudian dapat menjumlahkan pendapatan pembayaran pendaftaran secara otomatis. 3. Menggunakan kerangka kerja (<i>Framework</i>) laravel. 4. Pengguna aplikasi terdiri dari dua yaitu admin dan pasien.
2	Sihombing & Irawan, 2019. Perancangan Sistem Informasi Pendaftaran Pasien Menggunakan <i>Vb.Net</i> Pada Klinik Pratama Rbg Rz Bandung.	1. Bertujuan untuk membantu permasalahan pendaftaran pasien puskesmas yang masih manual. 2. Menggunakan metode pengembangan sistem model <i>Waterfall</i> .	1. Terdapat fitur limit berobat disetiap poli yang dapat diubah dan rekap laporan pasien yang bisa dicek baik perhari, perminggu, perbulan maupun pertahun kemudian dapat menjumlahkan pendapatan pembayaran pendaftaran secara otomatis. 2. Menggunakan kerangka kerja (<i>Framework</i>) laravel.
3	Zahratonnisa, dkk, 2022. Aplikasi pendaftaran pasien secara	1. Bertujuan untuk membantu permasalahan pendaftaran pasien puskesmas yang masih manual.	1. Terdapat fitur limit berobat disetiap poli yang dapat diubah, pembayaran online dan rekap laporan pasien yang bisa dicek baik perhari,

	<p><i>online</i> dan sistem pelayanan administrasi kesehatan pada puskesmas tapin utara.</p>	<p>2.Menggunakan metode pengembangan sistem model <i>Waterfall</i>.</p> <p>3.Menggunakan MySQL sebagai penyimpanan data/database.</p> <p>4.Menggunakan bahasa pemrograman PHP.</p>	<p>perminggu, perbulan maupun pertahun kemudian dapat menjumlahkan pendapatan pembayaran pendaftaran secara otomatis.</p> <p>2.Menggunakan kerangka kerja (<i>Framework</i>) laravel.</p>
4	<p>Rochman ,dkk, 2018.</p> <p>Perancangan Sistem Informasi Profil dan Pendaftaran Pasien Rawat Jalan Pada RSIA Pratiwi.</p>	<p>1.Bertujuan untuk membantu permasalahan pendaftaran pasien puskesmas yang masih manual dan berfokus pada pendaftaran pasien rawat jalan.</p> <p>2.Menggunakan metode wawancara, observasi dan studi pustaka dalam memperoleh data pada penelitian yang dilakukan.</p> <p>3.Menggunakan MySQL sebagai penyimpanan data/database.</p> <p>4.Menggunakan bahasa pemrograman PHP.</p>	<p>1.Aplikasi sudah bisa digunakan dipuskesmas yang berkaitan.</p> <p>2.Terdapat fitur limit berobat disetiap poli yang dapat diubah, pembayaran online dan rekap laporan pasien yang bisa dicek baik perhari, perminggu, perbulan maupun pertahun kemudian dapat menjumlahkan pendapatan pembayaran pendaftaran secara otomatis.</p> <p>3.Menggunakan kerangka kerja (<i>Framework</i>) laravel.</p>
5	<p>Eko, dkk, 2021.</p> <p>Aplikasi Sistem Informasi Puskesmas Bagian Registrasi Pendaftaran Pasien.</p>	<p>1.Bertujuan untuk membantu permasalahan pendaftaran pasien puskesmas yang masih manual dan berfokus pada bagian pendaftaran pasien.</p> <p>2.Menggunakan metode pengembangan sistem model <i>Waterfall</i>.</p> <p>3.Menggunakan MySQL sebagai penyimpanan data/database.</p>	<p>1.Terdapat fitur limit berobat disetiap poli yang dapat diubah, pembayaran online dan rekap laporan pasien yang bisa dicek baik perhari, perminggu, perbulan maupun pertahun kemudian dapat menjumlahkan pendapatan pembayaran pendaftaran secara otomatis.</p> <p>2.Menggunakan kerangka kerja (<i>Framework</i>) laravel.</p>

		4.Menggunakan bahasa pemrograman PHP.	
--	--	---------------------------------------	--

2.2 Aplikasi

Menurut Siregar dalam jurnal (Azis, 2023) Aplikasi merupakan penerapan, menyimpan sesuatu hal, data, permasalahan, pekerjaan ke dalam suatu sarana atau media yang dapat digunakan untuk diterapkan menjadi sebuah bentuk yang baru. Pengertian aplikasi secara umum adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya aplikasi merupakan suatu perangkat komputer yang siap pakai bagi *user*. Aplikasi adalah sebuah perangkat lunak yang berisi sebuah *coding* atau perintah yang dimana bisa diubah sesuai dengan keinginan. Aplikasi adalah sebuah perangkat lunak yang dimana tujuannya adalah agar bisa melayani setiap aktivitas komputerisasi yang dilakukan oleh pengguna (Syani & Werstantia, 2018).

2.3 Website

Menurut Zufria dan Azhari dalam jurnal yang ditulis oleh (Zahir, 2019) *Website* adalah sebuah komponen yang terdiri dari teks, gambar, suara, animasi sehingga menjadi media informasi yang menarik dikunjungi oleh orang lain. Website adalah salah satu aplikasi yang berisikan dokumen-dokumen *multimedia* (*text*, gambar, suara, animasi, *video*) di dalamnya yang menggunakan protokol HTTP (*Hypertext Transfer Protocol*) dan untuk mengaksesnya menggunakan perangkat lunak yang disebut *browser*. *Browser* (perambah) adalah aplikasi yang mampu menjalankan dokumen-dokumen *web* dengan cara diterjemahkan (Hasugian, 2018). Tiap-tiap halamannya memberi informasi atau interaksi yang beraneka ragam. Informasi dan interaksi itu bisa berupa tulisan, gambar atau bahkan dapat ditampilkan dalam bentuk *video*, animasi, suara, dan lain-lain. Halaman *website* biasanya berupa dokumen yang ditulis dalam *format HyperText Markup Language* (HTML), yang bisa diakses melalui HTTP, HTTP adalah suatu protokol yang menyampaikan berbagai informasi dari *server website* untuk ditampilkan kepada para *user* atau pemakai melalui *web browser* (Nofyat, dkk., 2018).

2.4 Pendaftaran

Dalam Kamus Besar Bahasa Indonesia dalam jurnal yang ditulis oleh (Mardiana, dkk., 2021), Pendaftaran adalah pencatatan nama, alamat dan sebagainya dalam sebuah daftar. Menurut Departemen Pendidikan dan Kebudayaan dalam jurnal yang ditulis oleh (Mardiana, dkk., 2021). Pendaftaran menurut Kamus Besar Bahasa Indonesia (KBBI) merupakan sebuah proses, cara, perbuatan mendaftar (mendaftarkan). Pendaftaran sendiri berasal dari kata dasar daftar, yang mempunyai definisi sebagai catatan sejumlah nama atau hal (tentang kata –kata, nama orang, barang, dan sebagainya) yang disusun berderet dari atas ke bawah. Menurut DEPDIKBUD “Pendaftaran adalah proses, cara, perbuatan mendaftar yaitu pencatatan nama, alamat, dan sebagainya dalam daftar”. Jadi, pendaftaran adalah proses pencatatan identitas pendaftaran kedalam sebuah media penyimpanan yang digunakan dalam proses pendaftaran (Balqis, 2023).

2.5 Pasien

Menurut Undang-Undang Nomor 29 Tahun 2004 tentang Praktik Kedokteran dalam jurnal yang ditulis oleh (Awaliah & Arianto, 2020) menyebutkan bahwa Pasien adalah setiap orang yang melakukan konsultasi masalah kesehatannya untuk memperoleh pelayanan kesehatan yang diperlukan baik secara langsung maupun tidak langsung kepada dokter atau dokter gigi. Menurut (Fadila, 2022) Pasien merupakan pelanggan layanan kesehatan, tetapi pasien dalam hal ini hanya merupakan salah satu jenis pelanggan. Pelanggan layanan kesehatan merupakan semua orang yang sehari-harinya melakukan kontak dengan layanan kesehatan. Berdasarkan pasal 1 ayat 4 Undang-Undang RI Nomor 44 Tahun 2009 tentang Rumah Sakit, bahwa pasien adalah setiap orang yang melakukan konsultasi masalah kesehatannya untuk mendapatkan pelayanan kesehatan yang dibutuhkan, baik itu secara langsung maupun tidak langsung di rumah sakit. Sedangkan pasien merupakan seseorang yang menerima perawatan secara medis oleh perawat atau tenaga kesehatan baik itu memiliki kelemahan secara fisik maupun secara mental sehingga memercayakan kepada pihak lain.

2.6 Puskesmas

Peraturan Menteri kesehatan Republik Indonesia nomor 75 tahun 2014

tentang Pusat Kesehatan Masyarakat dalam jurnal yang ditulis oleh (Ramadhan, dkk.,2021), Puskesmas adalah fasilitas pelayanan kesehatan yang menyelenggarakan upaya kesehatan masyarakat dan upaya kesehatan perseorangan tingkat pertama, dengan lebih mengutamakan upaya promotif dan preventif, untuk mencapai derajat kesehatan masyarakat yang setinggi-tingginya di wilayah kerjanya. Dalam jurnal (Firmansyah & Dedi, 2021), menurut Depkes RI yang dikutip oleh Nor Sanah Mendefinisikan “Puskesmas adalah unit pelaksana teknis dinas kesehatan Kabupaten/Kota yang bertanggung jawab menyelenggarakan pembangunan kesehatan di suatu wilayah kesehatan”. Menurut ilham Akhsanu Ridho yang dikutip oleh Nor Sanah mendefinisikan “Puskesmas adalah suatu unit organisasi yang bergerak dalam bidang pelayanan kesehatan yang berada di garda terdepan dan mempunyai misi sebagai pusat pengembangan pelayanan kesehatan, yang melaksanakan pembinaan dan pelayanan kesehatan secara menyeluruh dan terpadu untuk masyarakat di suatu wilayah kerja tertentu yang telah ditentukan secara mandiri dalam menentukan kegiatan pelayanan namun tidak mencakup aspek pembiayaan”.

2.7 Rawat Jalan

Rawat jalan adalah pelayanan terhadap orang yang masuk rumah sakit, puskesmas, klinik atau fasilitas kesehatan lainnya, untuk keperluan observasi diagnose, pengobatan, rehabilitasi medik dan pelayanan kesehatan lainnya tanpa tinggal di ruang rawat inap dan pelayanan rawat jalan adalah pelayanan yang diberikan di unit pelaksanaan fungsional rawat jalan terdiri dari poliklinik umum dan poliklinik spesialis serta unit gawat darurat. Pelayanan rawat jalan (*ambulatory services*) adalah salah satu bentuk Dari pelayanan kedokteran. Secara sederhana yang dimaksud dengan rawat jalan adalah pelayanan kedokteran yang disediakan untuk pasien tidak dalam bentuk rawat inap ke dalam pengertian pelayanan rawat jalan ini termasuk tidak hanya yang diselenggarakan oleh sarana pelayanan kesehatan yang telah lazim dikenal seperti rumah sakit atau klinik, tetapi juga yang diselenggarakan di rumah pasien (*home care*) serta di rumah perawatan (*nursing homes*) bentuk pertama dari pelayanan rawat jalan adalah yang diselenggarakan oleh klinik yang ada kaitannya dengan Rumah Sakit (*hospital-based ambulatory*

care) (Andriyanto, 2023).

2.8 Poliklinik

Poliklinik adalah salah satu unit pelayanan masyarakat yang bergerak pada bidang kesehatan. Sebuah poliklinik yang menawarkan fasilitas perawatan kesehatan yang dikhususkan untuk perawatan pasien rawat jalan. Tujuan utama dari Poliklinik adalah melayani para pasien yang akan berobat dan merupakan kegiatan utama, untuk itu penyediaan informasi yang dapat mendukung dan meningkatkan sistem sangatlah dibutuhkan (Sanjaya, 2019).

Pada Puskesmas Pesisir Bukit memiliki sebanyak tujuh Poliklinik yaitu Poliklinik KIA, Poliklinik Anak, Poliklinik Umum, Poliklinik Lansia, Poliklinik Gigi, Poliklinik Mata dan Poliklinik PTM. Untuk masing-masing poliklinik memiliki jumlah batasan kuota yang dapat ditampung, mengingat jumlah tenaga pegawai dan fasilitas yang terbatas menyebabkan diberlakukan batasan kuota berobat pasien pada setiap poli agar pelayanan puskesmas dapat berjalan dengan baik.

2.9 Basis Data (Database)

Menurut Fauzi, et al., (2021), Basis data (*database*) merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data didalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Menurut (Dalimunthe, 2022) *Database* merupakan kumpulan *file-file* yang saling berkaitan dan berinteraksi, relasi tersebut bila ditunjukkan dengan kunci dari tiap - tiap *file* yang ada. Satu *database* menunjukkan kumpulan data yang dipakai dalam suatu lingkup perusahaan, instansi. Pengolahan database merupakan suatu cara yang dilakukan terhadap *file-file* yang berada di suatu instansi yang mana file tersebut dapat disusun, diurutkan, diambil sewaktu-waktu serta dapat ditampilkan dalam bentuk suatu laporan sehingga dapat mengolah *file-file* yang berisikan informasi tersebut secara rapi. DBMS komponen-komponen:

- A. Antarmuka, yang didalamnya terdapat bahasa penyelesaian data (manipulasi data bahasa).
- B. Bahasa definisi data (*datadefinition language*) untuk skema eksternal, skema konseptual dan skema internal.
- C. Sistem kontrol basis data (*Database Control System*) yang mengakses basis data karena adanya perintah dari bahasa manipulasi data.

2.10 MySQL

Menurut (Wahyuni, dkk 2020) *MySQL* merupakan turunan salah satu turunan *SQL (Structured Query Language)* dimana sebuah konsep pengoperasian *database*, terutama untuk pemilihan/seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. *MySQL* merupakan *multiuser database* yang menggunakan bahasa *Structured Query Language (SQL)*. *SQL* (dibaca “*ess-que-el*”) adalah bahasa standar *ANSI (American National Standards Institute)*, yang digunakan untuk mengakses *server database* atau lebih dikenal dengan *relational database management system (RDBMS)*.

2.11 PHP (Hypertext Preprocessor)

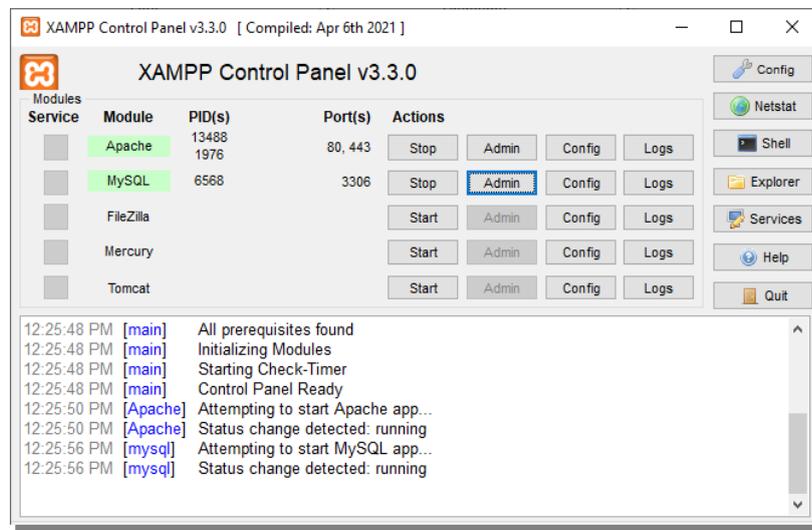
Hypertext Preprocessor (PHP) adalah suatu bahasa pemrograman yang digunakan untuk menterjemahkan basis data kode programer jadi kode mesin yang dapat dimengerti oleh komputer yang bersifat *server-side* yang ditambahkan ke *HTML* (Supono & Putratama, 2018). *PHP* merupakan kependekan dari *Hypertext Preprocessor* yang bahasa pemrograman *web* yang dapat disisipkan dalam skrip *HTML* dan bekerja di sisi *server* (Abdulloh, 2018). Berikut merupakan *syntax* dasar *PHP* seperti yang ditunjukkan pada gambar 2.1.

```
1  <?php
2  |      //code
3  ?>
4  |
```

Gambar 2.1 *Syntax* bahasa *PHP* (Sumber: www.simplilearn.com)

2.12 XAMPP

Iqbal dalam jurnal yang ditulis oleh (Putra, 2019) menyatakan XAMPP merupakan *software server apache* dimana dalam XAMPP yang telah tersedia *database server* seperti MySQL dan PHP *programming*. XAMPP memiliki keunggulan yaitu cukup mudah dioperasikan, tidak memerlukan biaya serta mendukung instalasi pada *Windows* dan *Linux*. Keuntungan lain yang didapatkan adalah hanya dengan melakukan instalasi cukup satu kali kemudian didalamnya tersedia MySQL, *apache web server*, *database server PHP support* (PHP 4 dan PHP 5) dan beberapa modul lainnya. XAMPP ialah *software* yang di dalamnya terdapat *server MySQL* dan didukung oleh PHP sebagai bahasa pemrograman untuk membuat *website* dinamis serta terdapat *web server apache* yang dapat dijalankan di beberapa *platform* seperti OS X, *Windows*, Linux, Mac, dan Solaris (Putra, 2019). Berikut merupakan tampilan XAMPP seperti yang ditunjukkan pada gambar 2.2.



Gambar 2.2 Tampilan XAMPP (Sumber: teknologiterupdate.com)

2.13 HTML (Hypertext Markup Language)

Hypertext Markup Language (HTML) merupakan bahasa dasar pembuatan *web*. HTML menggunakan tanda (*mark*), untuk menandai bagian-bagian dari *text*. HTML disebut sebagai bahasa dasar, karena dalam membuat *web*, jika hanya menggunakan HTML maka tampilan *web* terasa hambar (Rerung, 2018). HTML disimpan dengan ekstensi *.html, jika di dalamnya tidak mengandung skrip PHP.

Jika di dalamnya mengandung skrip PHP, maka disimpan dengan ekstensi *.php. Dalam penulisan HTML ada beberapa *tag* yang wajib dituliskan dengan struktur yang sudah ditentukan (Abdulloh, 2018). Berikut merupakan *syntax* dasar HTML seperti yang ditunjukkan pada gambar 2.3.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Document</title>
  </head>
  <body>
    <!-- Tulis disini -->
  </body>
</html>
```

Gambar 2.3 *Syntax* dasar bahasa pemrograman HTML (Sumber: cahya-rzi.com)

2.14 CSS (Cascading Style Sheet)

CSS atau singkatan dari *Cascading Style Sheet* adalah suatu aturan untuk mengatur tampilan dari *website* sehingga tampilan dalam *web* lebih terstruktur. CSS sendiri bukanlah bahasa pemrograman, CSS lebih seperti konfigurasi tampilan dari suatu *tag* pada *website*. CSS dapat merubah *text*, warna, *background* dan posisi dari suatu *tag* (Marlina & Masnur, 2021). Menurut (Taqrudian, 2018). CSS singkatan dari *cascading style sheets*, berisi rangkaian instruksi yang menentukan bagaimana suatu teks akan tertampil di halaman *web*. Perancangan desain teks dapat dilakukan dengan mendefinisikan *fonts* (huruf), *colors* (warna), *margins* (ukuran), latar belakang (*background*), ukuran *font* (*font size*) dan lain-lain. Elemen-elemen seperti *colors* (warna), *fonts* (huruf), *sizes* (ukuran) dan *spacing* (jarak) disebut juga *styles*. *Cascading Style Sheets* juga bisa berarti meletakkan *styles* yang berbeda pada lapisan yang berbeda. CSS terdiri dari *style sheet* yang memberitahukan *browser* bagaimana suatu dokumen akan disajikan. Fitur-fitur baru pada halaman *web* lama dapat ditambahkan dengan bantuan *style sheet*. Saat menggunakan CSS, tidak perlu lagi untuk menulis *font*, *color* atau *size* pada setiap *paragraf* atau pada setiap dokumen.

2.15 Bootstrap

Bootstrap merupakan salah satu *framework* CSS yang paling populer dari

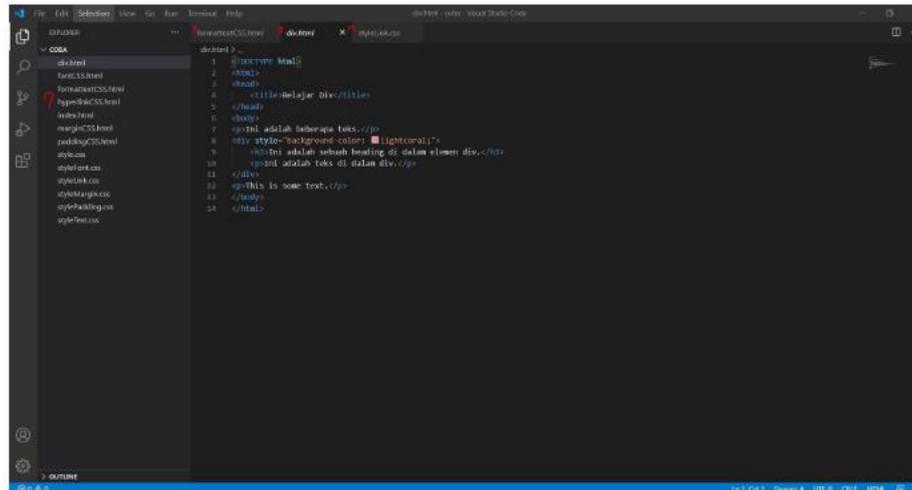
sekian banyak framework CSS yang ada. *Bootstrap* memungkinkan desain sebuah *website* menjadi *responsive* sehingga dapat dilihat dari berbagai macam ukuran device dengan tampilan tetap menarik. *Bootstrap* juga membuat proses pengaturan desain menjadi lebih cepat karena tidak perlu lagi banyak menulis CSS, bahkan hampir tidak perlu, kecuali memerlukan pengaturan desain yang berbeda dengan *style bootstrap*. *Bootstrap* telah didukung oleh hampir semua *browser* baik pada desktop maupun *mobile* (Abdullah, 2018).

2.16 JavaScript

Menurut (Marlina & Masnur, 2021) *Javascript* adalah bahasa pemrograman yang mendekati bahasa manusia atau bisa dikatakan bahasa tingkat tinggi, maka dari itu *javascript* mudah di pelajari. *Javascript* sendiri tujuannya di buat untuk memperkaya fitur pada *website* agar lebih dinamis, seperti untuk menampilkan dan menghilangkan objek-objek ada *website* kemudian dengan fungsi *javascript* dapat memanggil kembali objek yang di hilangkan tersebut. *JavaScript* merupakan bahasa pemrograman *website* yang pemrosesnya dilakukan di sisi *client*. Karena berjalan disisi *client*, *JavaScript* dapat dijalankan hanya dengan menggunakan *browser*. Berbeda dengan PHP yang bekerja disisi *server*, untuk menjalankan *script JavaScript* tidak memerlukan *refresh* pada *browser*. *JavaScript* biasanya dijalankan ketika ada *event* tertentu yang terjadi pada halaman *website*. Baik event yang dilakukan oleh *user*, maupun *event* yang terjadi karena adanya perubahan pada halaman *website* (Abdulloh, 2018).

2.17 Visual Studio Code

Menurut (Permana & Romadlon, 2019), *Visual Studio Code (VS Code)* ini adalah sebuah teks editor ringan dan handal yang dibuat oleh *Microsoft* untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi *Linux*, *Mac*, dan *Windows*. Teks editor ini secara langsung mendukung bahasa pemrograman *JavaScript*, *Typescript*, dan *Node.js*, serta bahasa pemrograman lainnya dengan bantuan plugin yang dapat dipasang via marketplace *Visual Studio Code* (seperti *C++*, *C#*, *Python*, *Go*, *Java*, dan seterusnya). Berikut merupakan tampilan *Visual Studio Code* seperti yang ditunjukkan pada Gambar 2.4.



Gambar 2.4 Tampilan *Visual Studio Code* (Sumber: Kotakode.com)

2.18 Laravel

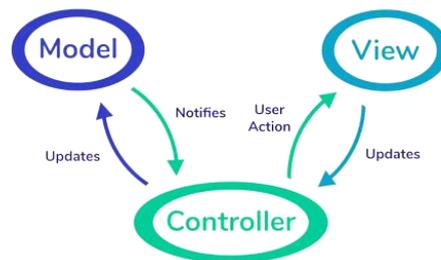
Menurut (Amir, dkk., 2020), Laravel adalah sebuah *framework* PHP yang dirilis dibawah lisensi MIT dan dikembangkan pertama kali oleh Taylor Otwell, dibangun dengan konsep MVC (*Model View Controller*). Sedangkan Menurut (Naista, 2017) mengemukakan bahwa *framework* adalah suatu struktur konseptual dasar digunakan untuk memecahkan atau menangani suatu masalah yang bersifat kompleks. Singkatnya, *framework* merupakan suatu kerangka kerja dari sebuah *website* yang akan dibangun. Dengan menggunakan kerangka tersebut, waktu yang perlukan dalam membangun sebuah *website* menjadi lebih singkat dan memudahkan dalam proses perbaikan.

Menurut (Abdulloh, 2018) mengatakan bahwa terdapat beberapa keunggulan yang dimiliki Laravel yaitu sebagai berikut.

1. Laravel memiliki banyak fitur yang tidak dimiliki oleh *framework* lain.
2. Laravel merupakan *framework* PHP yang ekspresif, artinya sintaks pada Laravel menggunakan bahasa yang mudah dimengerti sehingga programmer pemula sekalipun akan mudah memahami kegunaan dari suatu sintaks meskipun *programer* tersebut belum mempelajarinya.
3. Laravel memiliki dokumentasi yang cukup lengkap, bahkan setiap versinya memiliki dokumentasi tersendiri mulai dari cara instalasi hingga penggunaan fitur-fiturnya.
4. Laravel digunakan oleh banyak programmer sehingga banyak *library* yang

- mendukung Laravel yang diciptakan para programmer pecinta Laravel.
5. Laravel didukung oleh *Composer* sehingga *library-library* diperoleh dengan mudah dari internet menggunakan *Composer*.
 6. Laravel memiliki template *engine* tersendiri yang diberi nama *blade* yang memudahkan dalam menampilkan data pada template HTML.

2.19 Arsitektur Sistem MVC



Gambar 2.5 Arsitektur Sistem MVC (Sumber: mohsai.com)

Menurut (Setiawan, 2021), *Model View Controller* atau yang dapat disingkat MVC adalah sebuah pola arsitektur dalam membuat sebuah aplikasi dengan cara memisahkan kode menjadi tiga bagian yang terdiri dari:

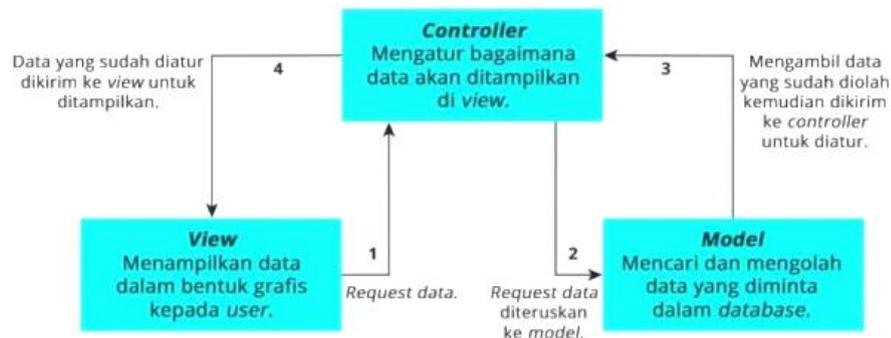
- a. *Model* merupakan bagian yang bertugas untuk menyiapkan, mengatur, memanipulasi, dan mengorganisasikan data yang ada di *desktop*.
- b. *View* merupakan bagian yang bertugas untuk menampilkan informasi dalam bentuk *Graphical User Interface (GUI)*.
- c. *Controller* merupakan bagian yang bertugas untuk menghubungkan serta mengatur *model* dan *view* agar dapat saling terhubung.

- Alur *Model View Controller*

Setelah penjelasan mengenai MVC dan komponennya, terdapat alur *model* proses dari MVC itu sendiri yaitu:

- a. Proses pertama adalah *view* akan meminta data untuk ditampilkan dalam bentuk grafis kepada pengguna.
- b. Permintaan tersebut diterima oleh *controller* dan diteruskan ke *model* untuk diproses.
- c. *Model* akan mencari dan mengolah data yang diminta di dalam *desktop*.
- d. Setelah data ditemukan dan diolah, *model* akan mengirimkan data tersebut kepada *controller* untuk ditampilkan di *view*.

e. *Controller* akan mengambil data hasil pengolahan *model* dan mengaturnya di bagian *view* untuk ditampilkan kepada pengguna.



Gambar 2.6 Alur Model View Controller (Sumber: Dicoding.com)

Alasan dari penggunaan MVC untuk aplikasi pendaftaran ini adalah sebagai berikut:

a. Proses pengembangan aplikasi menjadi lebih efisien

Penggunaan MVC dapat mempercepat pengembangan aplikasi karena kode dapat dikerjakan oleh beberapa *developer*. Contohnya dalam kasus pengembangan aplikasi *web*, bagian *model* dan *controller* dapat dikerjakan oleh *back-end developer* sedangkan bagian *view* dapat dilakukan oleh *front-end developer*.

b. Dapat melakukan *testing* dengan lebih mudah

Untuk memastikan seluruh aplikasi bekerja sesuai dengan rencana maka langkah testing atau uji coba wajib dilakukan. Dengan menggunakan *model view controller* ini, maka proses uji coba dapat dilakukan pada setiap bagian.

c. Perbaiki bug atau *error* lebih cepat untuk diselesaikan

Penggunaan MVC dapat memudahkan *developer* untuk memperbaiki *error* atau *bug* yang terjadi. *Developer* dapat fokus untuk menemukan dan memperbaiki masalah yang terjadi karena kode dituliskan pada bagian-bagian terpisah.

d. Mempermudah pemeliharaan

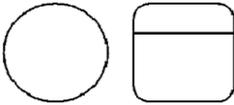
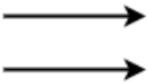
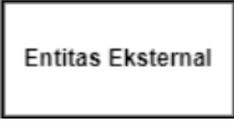
Konsep MVC ini dapat mempermudah pemeliharaan aplikasi, karena *script* atau kode yang lebih rapi dan terstruktur sehingga mempermudah *developer* dalam proses pemeliharaan aplikasi. Kesimpulannya MVC adalah

metode dalam membuat aplikasi dengan cara memisahkan kode menjadi tiga bagian, yaitu *Model, View, dan Controller*. Karena memiliki tiga bagian yang saling berkaitan satu sama lain *model,view,controller* ini jadi memiliki banyak keuntungan untuk pengembangan dan pemeliharaan aplikasi.

2.20 Data Flow Diagram (DFD)

Menurut (Mutoffar, dkk., 2019) Data flow diagram (DFD-DAD/Diagram Alir Data), memperlihatkan hubungan fungsional dari nilai yang di hitung oleh sistem, termasuk nilai masukan, nilai keluaran, serta penyimpanan tempat internal. DFD merupakan alat yang di gunakan pada metodologi pengembangan sistem yang terstruktur. DFD merupakan alat yang cukup populer sekarang ini, karena dapat menggambarkan arus data didalam dengan terstruktur jelas. Berikut ini adalah simbol *Data Flow Diagram* yang dapat dilihat pada tabel 2.2.

Tabel 2.2 Simbol *Data Flow Diagram (DFD)*

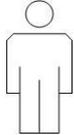
NO	SIMBOL	KETERANGAN
1		Entitas eksternal, dapat berupa orang/unit terkait yang berinteraksi dengan sistem tetapi diluar sistem.
2	Proses 	Orang/ unit yang mempergunakan atau melakukan transformasi data. Komponen fisik tidak diidentifikasi.
3	Aliran Data 	Aliran data dengan arah khusus dari sumber ke tujuan.
4		Entitas eksternal, dapat berupa orang/unit terkait yang berinteraksi dengan sistem tetapi diluar sistem.

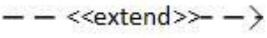
2.21 Use Case Diagram

Menurut (Sukamto & Shalahuddin, 2018) “*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 di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu”. Berikut simbol-simbol yang digunakan dalam *Use Case Diagram* disertai dengan keterangan fungsinya sebagaimana dijelaskan pada Tabel 2.3.

Tabel 2.3 Simbol *Use Case Diagram*

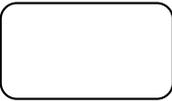
NO	SIMBOL	NAMA	KETERANGAN
1		Actor	Aktor adalah <i>Abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem.
2		<i>Use case</i>	<i>Use Case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktif, yang dinyatakan dengan menggunakan kata kerja
3		<i>Generalization</i>	Hubungan <i>generalisasi</i> dan <i>spesialisasi</i> (umum-khusus) antara dua buah <i>use case</i> , dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
4		<i>Assosiation</i>	Komunikasi antar <i>aktor</i> dan <i>use case</i> yang berpartisipasi pada <i>use case</i> , atau <i>use case</i> memiliki interaksi dengan <i>aktor</i> .

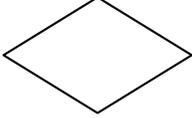
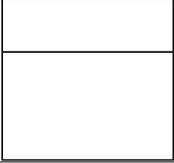
NO	SIMBOL	NAMA	KETERANGAN
5		<i>Include</i>	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan <i>fungsi</i> dari <i>use case</i> lainnya.
6		<i>Extend</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.

2.22 Diagram Aktivitas (Activity Diagram)

Menurut (Sukamto & Shalahuddin, 2018) “diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan. Berikut simbol-simbol yang digunakan dalam *Activity Diagram* disertai dengan keterangan fungsinya sebagaimana dijelaskan pada Tabel 2.4.

Tabel 2.4 Simbol *Activity Diagram*

NO	SIMBOL	NAMA	KETERANGAN
1		Status Awal	<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktivitas.
2		Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3		Status Akhir	<i>End Point</i> , akhir aktivitas.

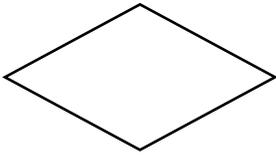
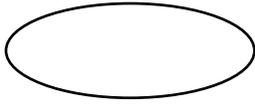
NO	SIMBOL	NAMA	KETERANGAN
4		Decision/percabangan	<i>Decision Points</i> , menggambar kan pilihan untuk pengambilan keputusan, <i>true</i> atau <i>false</i> .
5		Penggabungan/ Join	Simbol yang digunakan dimana ketika ada lebih dari satu aktivitas yang digabungkan menjadi satu.
6		Swimlane	<i>Swimlane</i> , pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa.

2.23 Entity Relationship Diagram (ERD)

Menurut (Lia & Guna, 2021) *Entity Relationship Diagram (ERD)* merupakan gambaran grafis dari suatu model data yang menyertakan deskripsi detail dari seluruh *entity*, *relationship*, dan *constraint*. Ada beberapa elemen penyusun ERD yaitu: Entitas (*Entity*) Entitas merupakan suatu yang nyata atau abstrak dimana kita akan menyimpan data. Relasi (*Relationship*) Relasi merupakan hubungan alamiah yang terjadi antar satu atau lebih. Atribut (*attribute*) Atribut merupakan ciri umum atau sebagian besar intisari pada entitas tertentu. Berikut simbol-simbol yang digunakan dalam ERD disertai dengan keterangan fungsinya sebagaimana dijelaskan pada tabel 2.5.

Tabel 2.5 Simbol *Entity Relationship Diagram (ERD)*

NO	SIMBOL	KETERANGAN
1		Entitas merupakan suatu yang nyata atau abstrak dimana kita akan menyimpan data

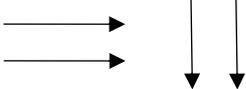
NO	SIMBOL	KETERANGAN
2		Relasi merupakan hubungan alamiah yang terjadi antar satu atau lebih.
3		Atribut merupakan ciri umum atau sebagian besar intisari pada entitas tertentu.
4		Garis merupakan penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut.

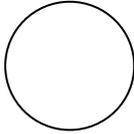
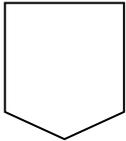
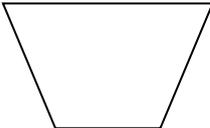
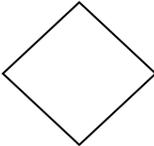
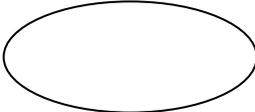
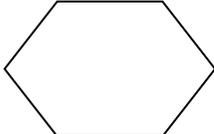
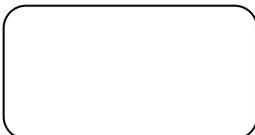
Dari ERD di atas terdapat derajat kardinalitas, Derajat kardinalitas merupakan penjelasan dari tingkat hubungan antar entitas. Ukuran derajat kardinalitas dibagi menjadi tiga macam yaitu *one-to-one*, *one-to-many*, *many-to-one* dan *many-to-many*.

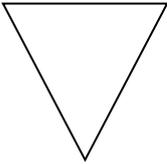
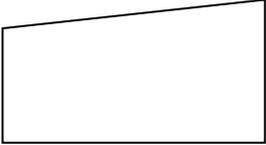
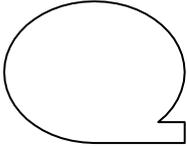
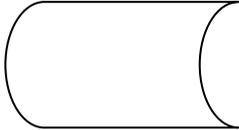
2.24 Flowchart

Diagram Alir (*Flowchart*) adalah bagan (*chart*) yang menunjukkan alir atau arus (*flow*) di dalam program atau prosedur sistem secara logika (Rusmawan, 2019). *Flowchart* atau diagram alir merupakan *chart* (bagan) yang menunjukkan hasil (*flow*) dalam program atau prosedur sistem secara logika. Digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi. Diagram ini bisa memberi solusi selangkah demi selangkah untuk penyelesaian masalah yang ada di dalam proses atau algoritma tersebut. Berikut simbol-simbol yang digunakan dalam *Flowchart* disertai dengan keterangan fungsinya sebagaimana dijelaskan pada Tabel 2.6.

Tabel 2.6 Simbol *Flowchart*

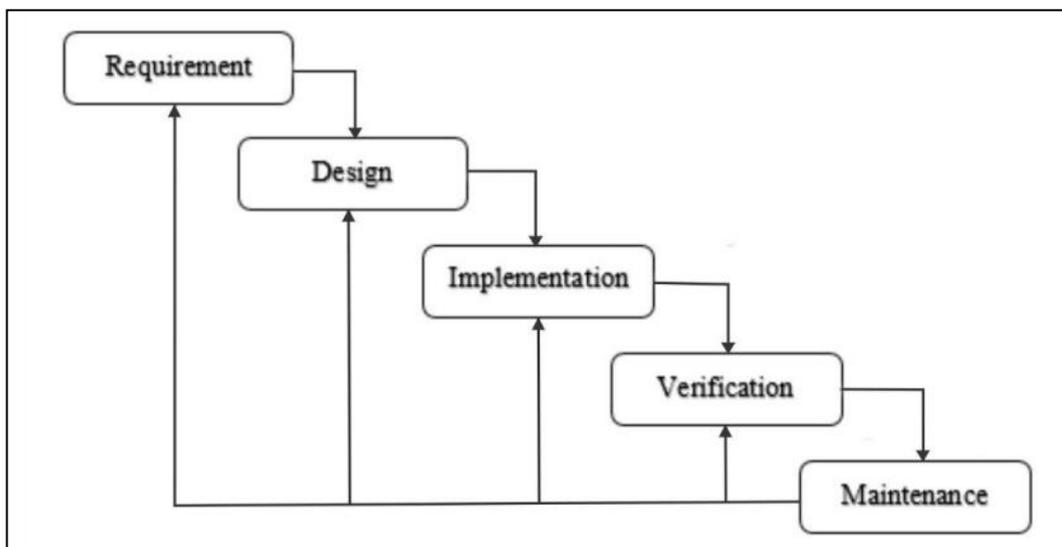
NO	SIMBOL	KETERANGAN
1		Simbol arus/ <i>flow</i> , berfungsi untuk menyatakan jalannya arus suatu proses

NO	SIMBOL	KETERANGAN
2		Simbol <i>connector</i> , berfungsi untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama
3		Simbol <i>offline connector</i> , berfungsi untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda
4		Simbol <i>process</i> , berfungsi untuk menyatakan suatu tindakan (proses) yang dilakukan oleh komputer
5		Simbol <i>manual</i> , berfungsi untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh computer
6		Simbol <i>decision</i> , berfungsi untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban: ya/tidak
7		Simbol <i>terminal</i> , berfungsi untuk menyatakan permulaan atau akhir suatu program
8		Simbol <i>predefined process</i> , berfungsi untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal
9		Simbol <i>keying operation</i> , berfungsi untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai <i>keyboard</i>

NO	SIMBOL	KETERANGAN
10		Simbol <i>offline-storage</i> , berfungsi untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu
11		Simbol <i>manual input</i> , berfungsi untuk memasukkan data secara manual dengan menggunakan <i>online keyboard</i>
12		Simbol <i>input/output</i> , berfungsi untuk menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung jenis peralatannya
13		Simbol <i>magnetic tape</i> , berfungsi untuk menyatakan <i>input</i> berasal dari pita magnetis atau <i>output</i> disimpan ke pita magnetis
14		Simbol <i>disk storage</i> , berfungsi untuk menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i> .
15		Simbol <i>document</i> , berfungsi untuk mencetak keluaran dalam bentuk dokumen (melalui <i>printer</i>)
16		Simbol <i>punched card</i> , berfungsi untuk menyatakan <i>input</i> berasal dari kartu atau <i>output</i> ditulis ke kartu

2.25 Metode Diagram Alir (Waterfall)

Model *Waterfall* merupakan salah satu model SDLC yang sering digunakan dalam pengembangan sistem informasi atau perangkat lunak. Model ini menggunakan pendekatan sistematis dan berurutan. Tahapan dalam model ini dimulai dari tahap perencanaan hingga tahap pengelolaan (*maintenance*) dan dilakukan secara bertahap. Pengembang perlu mengetahui lebih lanjut tentang bagaimana proses pengembangan sistem jika menggunakan model waterfall dan juga karakteristik dari model waterfall tersebut (Wahid, 2020). Berikut merupakan tahapan pada metode *waterfall* seperti yang ditunjukkan pada Gambar 2.7.



Gambar 2.7 Tahapan Metode *Waterfall*

Berikut merupakan tahapan dari metode waterfall menurut (Pressman, 2012) dalam jurnal (Wahid, 2020):

- *Requirement*

Tahap ini pengembang sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut. Informasi dapat diperoleh melalui wawancara, diskusi atau survei langsung. Informasi dianalisis untuk mendapatkan data yang dibutuhkan oleh pengguna.

- *Design*

Pada tahap ini, pengembang membuat desain sistem yang dapat membantu menentukan perangkat keras (hardware) dan sistem persyaratan dan juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.

- *Implementation*

Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut unit, yang terintegrasi dalam tahap selanjutnya. Setiap unit dikembangkan dan diuji untuk fungsionalitas yang disebut sebagai unit testing.

- *Verification*

Pada tahap ini, sistem dilakukan verifikasi dan pengujian apakah sistem sepenuhnya atau sebagian memenuhi persyaratan sistem, pengujian dapat dikategorikan ke dalam unit testing (dilakukan pada modul tertentu kode), sistem pengujian (untuk melihat bagaimana sistem bereaksi ketika semua modul yang terintegrasi) dan penerimaan pengujian (dilakukan dengan atau nama pelanggan untuk melihat apakah semua kebutuhan pelanggan puas).

- *Maintenance*

Ini adalah tahap akhir dari metode waterfall. Perangkat lunak yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya.

2.26 Pengujian Black Box

Black Box Testing merupakan pengujian yang dapat dilakukan dengan melakukan pengamatan, pada hasil eksekusi melalui beberapa data uji dan memeriksa fungsional yang terdapat pada perangkat lunak. Jadi dapat kita dianalogikan seperti halnya kita melihat ke dalam kotak hitam, sehingga kita hanya bisa melihat tampilan luarnya saja tanpa kita tau apa yang ada didalam kotak hitam tersebut (Susanto, dkk, 2022). Metode *Blackbox Testing* merupakan salah satu metode yang mudah digunakan karena hanya memerlukan batas bawah dan batas atas dari data yang di harapkan, Estimasi banyaknya data uji dapat dihitung melalui banyaknya *field* data entri yang akan diuji, aturan entri yang harus dipenuhi serta kasus batas atas dan batas bawah yang memenuhi (Rahmawati & Angga, 2022).