

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Teknologi informasi sangat membantu dalam kemudahan serta kecepatan pengiriman, penyampaian dan penerimaan informasi. Mulai dari perusahaan, sekolah, perguruan tinggi, dan lembaga atau organisasi lainnya telah banyak memanfaatkan aplikasi berbasis *website* dalam kegiatan penjualan, promosi, belajar dan kegiatan lainnya dibutuhkan pengiriman, penyebaran dan penerimaan informasi sehingga memberikan kemudahan bagi pengguna (*user*) yang membutuhkan (Mukrodin & Sugiamto, 2020).

Penerimaan Peserta Didik Baru (PPDB) merupakan langkah atau proses awal yang dilakukan penyelenggara pendidikan baik tingkat dasar ataupun tinggi. Berdasarkan peraturan Menteri Pendidikan dan Kebudayaan nomor 44 Tahun 2019, Bab I Pasal 2. Tentang Penerimaan Peserta Didik Baru pada Taman Kanak-Kanak, Sekolah Dasar, Sekolah Menengah Pertama, Sekolah Menengah Atas dan Sekolah Menengah Kejuruan, PPDB dilakukan berdasarkan: a. nondiskriminatif; b. objektif; c. transparan; d. akuntabel; dan e. berkeadilan. (2) Nondiskriminatif sebagaimana dimaksud pada ayat (1) dikecualikan bagi Sekolah yang secara khusus melayani peserta didik dari kelompok gender atau agama tertentu (Hidayat, Muttaqin, & Djamaludin, 2020).

Pondok Pesantren Darul Furqon merupakan pondok pesantren yang terfavorit di masyarakat Rokan Hulu khususnya masyarakat di wilayah Kota Tengah. Pada saat ini pengelolaan informasi dan penerimaan peserta didik baru oleh *admin* tata usaha di Pondok Pesantren Darul Furqon masih menggunakan metode yang sederhana, yaitu dengan menuliskan informasi dan data calon peserta didik pada buku arsip.

Metode tersebut memiliki kelemahan antara lain pengisian informasi tentang sekolah dan informasi data calon peserta didik baru membutuhkan banyak waktu dan tenaga, buku mudah rusak apabila disimpan terlalu lama, membutuhkan banyak penyimpanan, dan membutuhkan waktu yang cukup lama untuk merekap data calon peserta didik yang baru.

Terkait permasalahan-permasalahan tersebut, Pondok Pesantren Darul Furqon sangat memerlukan sebuah sistem informasi pendaftaran peserta didik baru berbasis *website* di Pondok Pesantren Darul Furqon sebagai solusi yang baik. Oleh karena itu, penulis mengambil judul penelitian yaitu **“SISTEM INFORMASI PENERIMAAN PESERTA DIDIK BARU PONDOK PESANTREN DARUL FURQON DENGAN MEMANFAATKAN *FRAMEWORK CODEIGNITER 3*”**.

1.2 Rumusan masalah

Berdasarkan uraian latar belakang, rumusan masalah penelitian ini adalah sebagai berikut:

1. Bagaimana merancang dan membuat sistem informasi penerimaan peserta didik baru Pondok Pesantren Darul Furqon dengan menggunakan *framework Codeignitier 3* ?
2. Bagaimana sistem informasi penerimaan peserta didik baru Pondok Pesantren Darul Furqon dapat mempermudah peserta untuk mendaftarkan diri lebih efektif dan efisien?
3. Bagaimana petugas pendaftaran Pondok Pesantren Darul Furqon lebih mudah mengelola informasi berita dan data peserta didik baru dan disimpan dalam waktu yang cukup lama?

1.3 Batasan masalah

Adapun batasan masalah penelitian ini yaitu sebagai berikut:

1. Sistem informasi penerimaan peserta didik baru Pondok Pesantren Darul Furqon hanya diterapkan di Pondok Pesantren Darul Furqon yang beralamatkan di Kota Tengah.
2. Sistem akan dirancang dengan menggunakan *framework Codeignitier 3* dengan menggunakan database MySQL.

3. Sistem yang dirancang dengan menggunakan manajemen basis model UML atau (*Unified Modelling Language*).

1.4 Tujuan Penelitian

Tujuan dari penelitian ini yaitu:

1. Menganalisis dan mengumpulkan data-data yang diperlukan untuk membuat dan merancang sistem informasi penerimaan peserta didik baru pondok pesantren Darul Furqon.
2. Membuat Sistem informasi penerimaan peserta didik baru yang dapat mempermudah calon peserta didik mendaftarkan diri ke pondok pesantren Darul Furqon.
3. Membuat sistem informasi penerimaan peserta didik baru yang dapat mempermudah petugas PPDB dalam memanagemen data pendaftaran peserta didik baru pondok pesantren Darul Furqon lebih efektif dan efesien.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini yaitu menambah pengetahuan peneliti maupun pembaca untuk merancang dan membuat sistem informasi peserta didik baru dengan menggunakan *framework codeigniter 3*.

1.6 Metode Pengumpulan Data

1. Pengamatan (Observasi)

Observasi adalah metode penelitian yang melibatkan pengamatan langsung terhadap suatu objek, peristiwa, atau fenomena. Tujuan utama dari observasi adalah melakukan eksplorasi untuk mengumpulkan data atau informasi secara sistematis dan obyektif mengenai apa yang diamati. Pengamatan dilakukan secara langsung di Pondok Pesantren Darul Furqon di Kota Tengah.

2. Wawancara (*Interview*)

Wawancara adalah proses komunikasi di mana seorang wawancara mengajukan pertanyaan kepada responden dengan tujuan untuk mendapatkan informasi, pendapat, atau pandangan mereka tentang suatu topik. Wawancara dilakukan secara langsung kepada pihak tata usaha Pondok Pesantren Darul Furqon.

3. Studi Kepustakaan (*Library Research*)

Pengumpulan data yang sesuai dan mempelajari masalah yang berkaitan dengan judul penelitian dan yang berhubungan dengan pembuatan program menggunakan *Framework Codeignitier* dengan menggunakan basis data MySQL.

1.7 Sistematika Penulisan

Sistematika penulisan laporan penelitian ini terdiri dari enam bab. Penggambaran dari tiap-tiap bab sebagai berikut:

BAB I PENDAHULUAN

Pada bab ini berisi tentang latar belakang, rumusan masalah, Batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

BAB II LANDASAN TEORI

Pada bab ini berisi tentang dasar-dasar teori yang digunakan sebagai landasan dalam pembuatan Sistem Informasi PPDB Pondok Pesantren Darul Furqon.

BAB III METODOLOGI PENELITIAN

Pada bab ini menjelaskan metode-metode yang digunakan untuk pembuatan Sistem Informasi PPDB Pondok Pesantren Darul Furqon.

BAB IV ANALISA DAN PERANCANGAN

Pada bab ini berisi bagaimana menganalisa cara kerja aplikasi yang akan dibangun, dan menjelaskan tahap perancangan aplikasi berdasarkan hasil analisis agar dimengerti pengguna.

BAB V IMPLEMENTASI DAN PENGUJIAN

Pada bab ini berisi bagaimana mengimplementasi dan pentestingan aplikasi/perangkat lunak berdasarkan analisa dan perancangan pada bab sebelumnya

BAB VI PENUTUP

Pada bab ini berisi mengenai kesimpulan dan saran. Kesimpulan merupakan jawaban dari rumusan masalah yang dibuat pada bab 1. Saran berisi kemungkinan pengembangan terhadap sistem yang dibuat penelitian selanjutnya.

BAB 2

LANDASAN TEORI

2.1 Sistem

Sistem (*system*) adalah kumpulan dari sub-sub sistem, elemen-elemen, prosedur-prosedur yang saling berinteraksi untuk Mencapai tujuan tertentu. Seperti informasi, target atau goal. Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu (Irvai et al., 2022). Sistem adalah himpunan suatu “benda” nyata (*set of atau think*) yang abstrak terdiri dari bagian-bagian atau komponen-komponen yang saling berkaitan, berhubungan, berketergantungan, dan saling mendukung, yang secara keseluruhan bersatu dalam kesatuan (*unity*) untuk mencapai tujuan tertentu secara efisien dan efektif (Sriyanti, Febrianto, & Artanto, 2023).

Penjelasan dari para ahli di atas, dapat disimpulkan bahwa Sistem adalah suatu jaringan kerja dari prosedural yang saling berhubungan dan berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu. Sistem informasi juga dapat didefinisikan sebagai kumpulan elemen yang saling berhubungan satu sama lain yang membentuk satu kesatuan untuk mengintegrasikan data memproses dan menyimpan serta mendistribusikan informasi.

2.2 Informasi

Informasi adalah data yang telah diolah menjadi bentuk yang memiliki arti bagi si penerima, dan bermanfaat bagi pengambilan keputusan saat ini atau mendatang (Irvai et al., 2022). Informasi adalah data yang telah diolah ataupun diklasifikasikan untuk digunakan dalam proses pengambilan keputusan. Informasi memiliki nilai yang ditentukan oleh dua hal yaitu manfaat dan biaya, suatu informasi bisa dikatakan bernilai apabila manfaat yang diberikan lebih berharga dibanding biaya untuk mendapatkannya (Faqih & Wahyudi, 2022). Informasi ialah fakta yang sudah dikategorikan atau diolah atau diartikan untuk digunakan dalam proses mengambil keputusan. Sistem pengolahan informasi akan mengolah fakta menjadi informasi atau mengolah fakta dari bentuk tidak berguna menjadi berguna untuk yang menerimanya (Veryal, Dewi, & Khambali, 2023). Dari pernyataan dari para ahli tersebut, dapat disimpulkan bahwa informasi atau embaran adalah pesan atau kumpulan pesan yang terdiri dari *order sekuens* dari simbol, atau makna yang dapat ditafsirkan dari pesan atau kumpulan pesan. Informasi dapat direkam atau ditransmisikan. Hal ini dapat dicatat sebagai tanda-tanda, atau sebagai sinyal berdasarkan gelombang. Sebagai contoh, dokumen berbentuk *spreadsheet (Microsoft Excel)* sering kali digunakan untuk membuat informasi dari data yang ada di dalamnya.

2.3 Sistem Informasi

Sistem adalah suatu jaringan prosedur yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan pokok perusahaan. Informasi adalah sekumpulan data atau fakta yang diorganisasi atau diolah dengan cara tertentu sehingga mempunyai arti bagi penerima. Sistem informasi merupakan sistem yang mempunyai kemampuan untuk mengumpulkan informasi dari semua sumber dan menggunakan berbagai media untuk menampilkan informasi (Pardede, Sihaloho, Tarigan, & Simanjuntak, 2022). Sistem informasi harus dapat menyediakan informasi untuk orang yang tepat dalam waktu yang tepat serta dalam format dan jumlah yang tepat dan sesuai dengan kebutuhan penerima informasi (Arief Selay et al., 2010). Sistem informasi menggunakan database sangat mendukung terhadap penyimpanan informasi secara cepat, tepat, seragam, dan mudah disesuaikan (Pratama & Rusliyawati, 2023).

Berdasarkan pendapat dari para ahli tersebut, dapat disimpulkan bahwa Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi organisasi yang bersifat manajeral dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan laporan-laporan yang diperlukan oleh pihak luar tertentu . Informasi adalah sekumpulan data atau fakta yang diorganisasi atau diolah dengan cara tertentu sehingga mempunyai arti bagi penerima.

2.4 PPDB

Tahapan awal untuk memulai jenjang pendidikan formal yaitu tahap Penerimaan Peserta Didik (PPDB). PPDB adalah proses penarikan calon peserta didik untuk dijadikan input sekolah. Kegiatan ini rutin dilakukan sekolah setiap tahun ajaran baru. Tahap PPDB harus dikelola dan dilaksanakan sesuai standar yang telah ditetapkan pemerintah (Rohmah, Wahyudi, & Pamungkas, 2020). Pendayagunaan teknologi informasi pada alur proses PPDB adalah aksi nyata dalam memfasilitasi terbangunnya *good governance* dalam dunia pendidikan (Yudahana, Riadi, & Elvina, 2023). Berdasarkan Peraturan Menteri Pendidikan dan Kebudayaan nomor 4 tahun 2019 tentang penerimaan peserta didik baru pada taman kanak-kanak. Sekolah dasar, sekolah menengah pertama, sekolah menengah atas dan kejuruan dilakukan berdasarkan nondiskriminatif, objektif, transparan, akuntabel, dan berkeadilan (Ismail & Yulvianda, 2022).

Berdasarkan dari penjelasan dari para ahli tersebut, dapat disimpulkan bahwa Penerimaan Peserta Didik (PPDB) adalah kegiatan langkah awal calon siswa baru untuk melanjutkan ke jenjang berikutnya. Penerimaan peserta didik baru ini terdapat beberapa tahapan-tahapan seperti tes atau yang sudah ditentukan oleh pihak sekolah tersebut. Penerimaan siswa baru ini adalah proses penarikan atau pencarian yang dibutuhkan oleh sekolah sesuai dengan dayaampungnya yang tersedia melalui jalur penyeleksian yang telah ditentukan oleh pihak pemerintah maupun pihak sekolah.

2.5 Website

Website adalah kumpulan halaman *web* dan semua *file* yang ditautkan bersama. *Web* terdiri dari satu atau lebih halaman dan serangkaian halaman yang disebut *homepage*. *Homepage* pada bagian posisi atas dengan halaman terkait berada di bawahnya. Setiap halaman di bawah *homepage* (*subsite*) biasanya berisi *hyperlink* ke halaman lain di web (Syahfitri, Mulya, Larasati, & Anwar, 2023). Berdasarkan pendapat ahli tersebut, dapat disimpulkan bahwa *Website* merupakan kumpulan halaman *web* dan semua file yang ditautkan bersama. *Homepage* pada bagian posisi atas dengan halaman terkait berada di bawahnya.

2.6 Alat Bantuan Perancangan Program

2.6.1 Framework

Framework secara sederhana dapat diartikan kumpulan dari fungsi-fungsi/prosedur-prosedur dan kelas-kelas untuk tujuan tertentu yang sudah siap digunakan sehingga bisa lebih mempermudah dan mempercepat pekerjaan seorang programmer, tanpa harus membuat fungsi atau kelas dari awal (Apriantsani Budiman & Maulana Nugraha, 2019). Dari pendapat ahli tersebut, dapat disimpulkan bahwa *Framework* merupakan sekumpulan fungsi, *class*, dan aturan-aturan yang bersifat mengatur bagaimana kita membangun aplikasi. *Framework* memungkinkan kita membangun aplikasi dengan lebih

cepat karena sebagai *developer* kita akan lebih memfokuskan pada pokok permasalahan.

Setiap *framework* memiliki kelebihan dan kelemahan sendiri, dan pilihan tergantung pada kebutuhan proyek dan preferensi pribadi. Penting untuk mempertimbangkan faktor seperti skala proyek, jenis aplikasi yang akan dikembangkan, serta level pengalaman dan kenyamanan dengan menggunakan *framework* tersebut. Berikut adalah beberapa *framework* populer untuk pengembangan web menggunakan bahasa pemrograman PHP yaitu **Laravel**: *Laravel* adalah salah satu *framework* PHP yang paling populer dan kuat. Ini menyediakan sintaks yang elegan, mendukung pola desain *Model-View-Controller* (MVC), serta menawarkan banyak fitur termasuk sistem rute yang kuat, ORM (*Eloquent*), autentikasi, dan banyak lagi. *Laravel* juga memiliki komunitas yang besar dan dokumentasi yang baik. **Symfony**: *Symfony* adalah *framework* PHP yang kuat dan modular. Ini memungkinkan Anda memilih komponen yang ingin digunakan dan membangun aplikasi sesuai kebutuhan. *Symfony* juga menekankan pada praktik-praktik pengembangan yang baik dan mengikuti standar industri. **CodeIgniter**: *CodeIgniter* adalah *framework* PHP ringan yang fokus pada kinerja dan kemudahan penggunaan. Ini adalah pilihan yang baik untuk proyek-proyek yang lebih kecil atau aplikasi web yang sederhana. Meskipun *CodeIgniter* 4 telah dirilis, versi 3 masih digunakan oleh banyak orang. **Yii**: *Yii* adalah *framework* PHP yang cepat dan efisien, terutama cocok untuk pengembangan aplikasi web yang kompleks. Ini memiliki banyak

fitur seperti sistem *cache* yang kuat, sistem autentikasi, ORM (*Active Record*), dan masih banyak lagi. **Phalcon**: *Phalcon* adalah *framework* PHP yang unik karena sebagian besar komponennya ditulis dalam bahasa pemrograman C untuk meningkatkan kinerja. Ini adalah pilihan yang baik jika Anda mencari kecepatan eksekusi yang tinggi. **Slim**: *Slim* adalah *framework* PHP yang dirancang untuk pengembangan aplikasi *web API* dan *mikroservis*. Ini ringan dan fokus pada kecepatan serta memudahkan pembuatan *RESTful API*. **CakePHP**: *CakePHP* adalah *framework* PHP yang mengikuti pendekatan konvensi lebih dari konfigurasi. Ini menyediakan banyak fitur seperti sistem rute, ORM (*CakeORM*), autentikasi, dan masih banyak lagi. **Laminas (sebelumnya Zend Framework)**: *Laminas* adalah *framework* PHP yang bersifat modular dan dapat diintegrasikan dengan berbagai komponen. Ini cocok untuk pengembangan proyek yang besar dan kompleks.

Pada tahap pembuatan sistem informasi penerimaan peserta didik baru di Pondok Pesantren darul Furqon dengan menggunakan *Framework Codeigniter*

3. Ada beberapa alasan penulis memilih untuk menggunakan *Codeigniter*:

1. **Ringan dan cepat**: *Codeigniter* dirancang untuk menjadi ringan dan cepat. Ini berarti bahwa aplikasi yang dibangun dengan *Codeigniter* cenderung memiliki performa yang baik dan dapat menangani lalu lintas yang tinggi dengan efisien.
2. **Dokumentasi yang baik**: *Codeigniter* memiliki dokumentasi yang komprehensif dan mudah dipahami. Ini sangat membantu pengembang

baru atau yang sudah berpengalaman dalam memahami cara kerja *framework* dan cara menggunakannya.

3. **Struktur kode yang jelas:** *Codeigniter* mendorong penggunaan struktur kode yang jelas dan terorganisir serta sangat mudah dipahami. Dengan menggunakan konsep *Model-View-Controller* (MVC), pengembang dapat memisahkan logika bisnis, tampilan, dan interaksi pengguna dengan mudah.
4. **Kebebasan dalam memilih komponen:** *Codeigniter* adalah *framework* yang tidak memaksa Anda untuk menggunakan semua komponen yang disediakan. Anda dapat memilih untuk menggunakan bagian-bagian tertentu sesuai kebutuhan Anda, sehingga memberikan fleksibilitas dalam pengembangan.
5. **Penggunaan *database* yang mudah:** *Codeigniter* menyediakan fitur-fitur yang memudahkan pengaksesan dan penggunaan database. Pengembang dapat dengan mudah berinteraksi dengan database melalui *Active Record* atau menggunakan *Query Builder* untuk membuat kueri SQL dengan cara yang lebih terstruktur.
6. **Banyak fitur bawaan:** *Codeigniter* memiliki banyak fitur bawaan yang siap digunakan, seperti validasi formulir, manajemen sesi, manajemen file, dan lain-lain. Ini dapat mempercepat proses pengembangan karena Anda tidak perlu membangun fitur-fitur ini dari awal.

7. **Komunitas dan dukungan:** *Codeigniter* memiliki komunitas yang cukup besar dan aktif. Ada banyak sumber daya online, tutorial, forum, dan tempat berdiskusi lainnya yang dapat membantu pengembang dalam mengatasi masalah dan belajar lebih lanjut tentang *framework* ini.
8. **Pembaruan dan peningkatan:** Meskipun ada banyak *framework* PHP modern lainnya, *Codeigniter* tetap aktif dalam merilis pembaruan dan peningkatan. Ini menunjukkan bahwa *framework* ini masih mendapatkan perhatian pengembang dan diperbarui untuk menjaga kelangsungan dan keamanannya.

2.6.2 *Codeigniter*

Codeigniter adalah salah satu *framework* PHP bahkan *framework* PHP yang paling powerfull saat ini karena di dalamnya terdapat fitur lengkap aplikasi *web* di mana fitur-fitur tersebut sudah dikemas menjadi satu. Selain itu, *codeigniter* juga saat ini banyak digunakan khususnya bagi *developer web* untuk mengembangkan aplikasi berbasis *web*-nya tersebut. *Framework* dapat diartikan sekumpulan perintah / program dasar di mana perintah dasar tersebut dapat digunakan lagi untuk menyelesaikan masalah yang lebih kompleks sehingga dapat digunakan untuk membantu membuat aplikasi baru atau aplikasi kompleks tanpa harus membuat program dari awal (Ria, Lorenza, & Mukti, 2020).

Berdasarkan pakar di atas, dapat disintesis bahwa *Codeigniter* merupakan aplikasi *open source* berupa *framework* PHP dengan model MVC

untuk membangun aplikasi *web* dinamis dengan cepat dan mudah. *Codeigniter* memiliki desain dan struktur file yang sederhana, didukung dengan dokumentasi yang lengkap sehingga *framework* ini lebih mudah dipelajari. *Codeigniter* adalah sebuah *framework* yang digunakan untuk membuat sebuah aplikasi berbasis *web* yang disusun dengan menggunakan bahasa PHP.

Untuk membangun sistem informasi penulis menggunakan *framework Codeigniter 3* beberapa alasan mengapa seseorang mungkin memilih untuk menggunakan *Codeigniter 3* yaitu **Stabilitas**: *Codeigniter 3* adalah versi yang sudah mapan dan stabil. Jika Anda memiliki proyek yang sudah berjalan dengan menggunakan versi ini dan tidak memerlukan fitur-fitur terbaru dari versi yang lebih baru, Anda mungkin ingin tetap menggunakan *Codeigniter 3* untuk menjaga stabilitas. **Kompatibilitas**: Beberapa proyek mungkin masih menggunakan kode dan struktur yang dibangun dengan *Codeigniter 3*. Pergantian ke versi yang lebih baru mungkin memerlukan penyesuaian besar-besaran dalam kode yang sudah ada. Jika proyek Anda telah berhasil menggunakan *Codeigniter 3* dan tidak perlu mengadopsi fitur-fitur terbaru, Anda mungkin ingin tetap konsisten dengan versi yang sudah ada. **Dukungan Komunitas**: Meskipun versi terbaru *Codeigniter* telah dirilis, masih ada sebagian besar komunitas pengguna yang masih menggunakan *Codeigniter 3*. Ini berarti Anda masih dapat mencari bantuan dan dukungan dari komunitas dalam mengatasi masalah atau mencari solusi. **Ketersediaan Sumber Daya**: Karena *Codeigniter 3* telah ada dalam waktu yang cukup lama, ada banyak sumber daya

online, tutorial, dan dokumentasi yang tersedia untuk membantu Anda memahami dan menggunakan *framework* ini dengan baik.

2.6.3 Data

Pengertian data ada bermacam-macam, secara umum menurut Kamus Umum Bahasa Indonesia (KUBI), “Data adalah bukti yang ditemukan dari hasil penelitian yang dapat dijadikan dasar kajian atau pendapat”. Secara teknis, data lebih berkaitan dengan pengumpulannya secara empiris. Dengan demikian, data merupakan satuan terkecil yang diwujudkan dalam bentuk simbol angka, simbol huruf, atau simbol gambar yang menggambarkan nilai suatu variabel tertentu sesuai dengan kondisi data di lapangan. Simbol angka, huruf atau gambar sering disebut dengan data mentah atau besaran yang belum menunjukkan suatu ukuran terhadap suatu konsep atau gejala tertentu. Besaran data tersebut belum memiliki arti apa pun jika belum dilakukan pengolahan atau analisis lebih lanjut dalam bentuk informasi atau indikator pendidikan (Baihaqi & Fansyuri, 2022).

Dari pendapat pakar tersebut, dapat disimpulkan bahwa data adalah catatan atas kumpulan fakta. Dalam penggunaan sehari-hari, data berarti suatu pernyataan yang diterima secara apa adanya. Pernyataan ini adalah hasil pengukuran atau pengamatan suatu variabel yang bentuknya dapat berupa angka, kata-kata, atau citra.

2.6.4 Basis Data

Basis data merupakan komponen mendasar suatu sistem informasi dimana pengembangan atau penggunaannya harus dilihat dari perspektif yang lebih luas berdasarkan kebutuhan organisasi. Dengan sistem informasi, memungkinkan terjadinya proses pengumpulan data, pengaturan, Pengawasan dan penyebaran informasi keseluruhan organisasi (Nancy Margareta S & Hendra Putra, 2022). Basis data ialah himpunan yang terorganisasi dari data-data yang berkaitan sedemikian rupa sehingga mudah disimpan, diubah, juga dipanggil oleh *user* (Veryal et al., 2023).

Dari uraian pendapat para ahli tersebut, dapat disimpulkan bahwa Basis data didefinisikan sebagai kumpulan data yang terintegrasi dan diatur sedemikian rupa sehingga data tersebut dapat dimanipulasi, diambil, dan dicari secara cepat. Pengumpulan dan analisis kebutuhan merupakan proses mengumpulkan dan menganalisa informasi tentang organisasi yang akan didukung oleh aplikasi basis data dan menggunakan informasi tersebut untuk mengidentifikasi kebutuhan *user* terhadap sistem baru. Pemanfaatan database memungkinkan untuk dapat menyimpan data atau melakukan perubahan dan menampilkan kembali data tersebut dengan cepat dan mudah .

2.6.5 Aliran sistem informasi (ASI)

Aliran sistem informasi sangat berguna untuk mengetahui permasalahan yang ada pada suatu sistem. Dari sini dapat diketahui apakah sistem informasi

tersebut masih layak dipakai atau tidak, masih manual atau komputerisasi. Jika sistem informasinya tidak layak lagi maka perlu adanya perubahan dalam pengolahan datanya sehingga menghasilkan informasi yang cepat dan akurat serta keputusan yang lebih baik (Muhammad Saed Novendri, Ade Saputra, & Firman, 2019).

Simbol	Keterangan
	Simbol dokumen Menunjukkan dokumen <i>input</i> dan <i>output</i> .
	Simbol kegiatan manual Menunjukkan kegiatan atau pekerjaan manual
	Simbol proses Menunjukkan kegiatan proses dari operasi program komputer.
	Simbol keyboard Menunjukkan <i>input</i> yang menggunakan <i>on-line keyboard</i> .
	Simbol harddisk Menunjukkan <i>input</i> ataupun <i>output</i> menggunakan <i>harddisk</i> .
	Simbol garis alir Menunjukkan arus dari setiap proses.
	Simbol penghubung Menunjukkan penghubung ke halaman yang masih sama atau ke halaman lain.
	Simbol arsip Menunjukkan pengarsipan <i>file</i> tanpa menggunakan komputer.
	Simbol keputusan Digunakan untuk suatu penyeleksian kondisi di dalam program.

Gambar 2.1 Simbol Aliran Sistem Informasi (ASI)

Berdasarkan pendapat yang dijelaskan para ahli tersebut, dapat disimpulkan bahwa Aliran Sistem Informasi (ASI) merupakan suatu bagan yang menggambarkan arus logika dari data yang akan diproses dari awal suatu

program sampai akhir program. Bagan alir terdiri dari simbol-simbol yang akan dikerjakan.

2.6.6 UML (*Unified Modelling Language*)

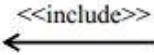
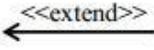
UML yang berarti bahasa pemodelan standar. Ketika kita membuat model menggunakan konsep UML ada aturan-aturan yang harus diikuti. Bagaimana elemen pada model-model yang kita buat berhubungan satu dengan yang lainnya harus mengikuti standar yang ada. UML bukan hanya sekedar diagram tetapi juga menceritakan konteksnya. UML diaplikasikan untuk maksud tertentu biasanya antara lain: merancang perangkat lunak, sarana komunikasi antara perangkat lunak dengan proses bisnis, menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan system, mendokumentasi sistem yang ada, proses-proses dan organisasinya (Mukrodin & Sugiamto, 2020). UML (*Unified Modelling Language*) adalah salah standar bahasa yang banyak di gunakan di dunia industri untuk mendefinisikan requirement, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Sriyanti et al., 2023). Teori dasar UML (Unified Modeling Language) dalam penggambaran pada rancangan awal dari sistem yang akan dibuat. Dalam membuat bentuk dari berbagai macam aplikasi perangkat lunak yang dapat beroperasi dengan baik dengan sistem operasi dan jaringan apapun, serta dapat menggunakan bahasa pemrograman apapun dapat menggunakan

UML (*Unified Modeling Language*). Banyaknya jenis model yang dimiliki UML, hanya beberapa yang akan dibahas dan digunakan dalam penulisan (Yulianti, 2023).

Dari uraian di atas, dapat disimpulkan bahwa UML merupakan bahasa pemodelan standar. Ketika kita membuat model menggunakan konsep UML ada aturan-aturan yang harus diikuti. Bagaimana elemen pada model-model yang kita buat berhubungan satu dengan yang lainnya harus mengikuti standar yang ada.

2.6.6.1 Use Case Diagram

Use Case atau Diagram *Use Case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu (Badrul, Ardy, Nusa Mandiri Jl Jatiwaringin Raya No, & Cipinang Melayu Jakarta Timur, 2021). Dari opini yang disampaikan oleh pakar tersebut, dapat disimpulkan bahwa *Use Case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

Simbol	Keterangan
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i>
	<i>Use case</i> : Abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan <i>use case</i>
	<i>Generalisasi</i> : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i>
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya
	Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi

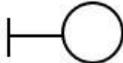
Gambar 2.2 Simbol-Simbol *Use Case Diagram*.

2.6.6.2 *Sequency Diagram*

Sequence Diagram mampu menggambarkan kolaborasi secara dinamis yang terdiri dari beberapa objek dan dimensi vertikal serta dimensi horizontal (Primadasa & Juliansa, 2020). Menggambarkan interaksi antara sejumlah objek dalam urutan waktu. Kegunaannya untuk menunjukkan rangkaian pesan yang

dikirim antara objek juga interaksi antar objek yang terjadi pada titik tertentu dalam eksekusi sistem (Mufida, Rahmawati, & Hertiana, 2019).

Dari beberapa penjelasan pakar di atas, dapat disimpulkan bahwa *Sequence Diagram* merupakan gambaran interaksi antara sejumlah objek dalam urutan waktu. Kegunannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antar objek yang terjadi pada titik tertentu dalam eksekusi sistem.

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menggambar orang yang sedang berinteraksi dengan sistem.
2		<i>Entity Class</i>	Menggambarkan hubungan yang akan dilakukan
3		<i>Boundary Class</i>	Menggambarkan sebuah gambaran dari foem
4		<i>Control Class</i>	Menggambarkan penghubung antara boundary dengan tabel
5		<i>A focus of Control & A Life Line</i>	Menggambarkan tempat mulai dan berakhirnya message
6		<i>A message</i>	Menggambarkan Pengiriman Pesan

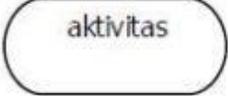
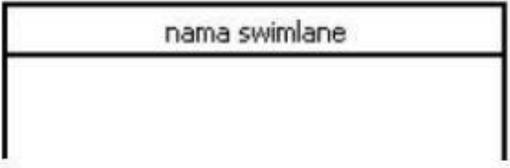
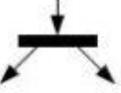
Gambar 2.3 Simbol-Simbol *Sequence Diagram*.

2.6.6.3 *Activity Diagram*

Activity diagram menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis atau menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Setiawansyah,

Titian Lestari, & Ayu Megawaty, 2022). *Activity diagram* mampu mengurutkan aktivitas proses dalam sistem dan membantu memahami proses secara keseluruhan dari sebuah atau beberapa *use case* (Primadasa & Juliansa, 2020).

Dari pengertian tersebut dapat disimpulkan bahwa *Activity Diagram* menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis atau menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Simbol	Deskripsi
status awal 	status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
aktivitas 	aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
percabangan / <i>decision</i> 	asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
penggabungan / <i>join</i> 	asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
status akhir 	status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
swimlane 	memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi
<i>fork,</i> 	digunakan utk menunjukkan kegiatan yg dilakukan secara paralel
<i>join,</i> 	digunakan utk menunjukkan kegiatan yg digabungkan

Gambar 2.4 Simbol-Simbol *Activity Diagram*.

2.6.6.4 Class Diagram

Use Case Diagram merupakan pemodelan untuk kelakuan sistem informasi yang akan dibuat. Use case bekerja dengan mendeskripsikan tipikal interaksi antara user sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sistem itu dipakai (Wira, Putra, & Andriani, 2019). Berdasarkan gagasan dari ahli tersebut, dapat disimpulkan bahwa Diagram Kelas (*Class Diagram*) juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan constraint yang berhubungan dengan objek yang dikoneksikan. *Class Diagram* secara khas meliputi Kelas (*Class*), Relasi *Assosiations*, *Generalitation* dan *Aggregation*, atribut (*Attributes*), operasi (*operation/method*) dan *visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut.

SIMBOL CLASS DIAGRAM

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	<u>Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor</u>
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	<u>Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri</u>
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

Gambar 2.5 Simbol-Simbol *Class Diagram*.

2.6.7 Alat Pemrograman Web

2.6.7.1 HTML (*Hyper Text Markup Language*)

HTML adalah bahasa standard yang digunakan untuk menampilkan halaman *web*. Hal-hal yang bisa dilakukan dengan HTML, yaitu mengatur tampilan dari halaman *web* dan isinya, membuat tabel dalam halaman *web*, mempublikasikan halaman *web* secara online, membuat form yang bisa menangani registrasi dan transaksi via *web*, menambahkan objek-objek seperti audio, video, animasi, *java applet* dalam

halaman *web*, dan menampilkan area gambar (*canvas*) di browser (Ria et al., 2020). Penjelasan yang diungkapkan ahli tersebut, dapat disimpulkan bahwa HTML digunakan untuk mengelola serangkaian data dan informasi sehingga suatu dokumen dapat diakses dan ditampilkan di Internet melalui layanan *web*. Fungsi HTML yang lebih spesifik yaitu : Membuat halaman *web*.

2.6.7.2 Java Script

Java Script adalah bahasa pemrograman berbentuk kumpulan *script* yang berjalan pada suatu dokumen HTML. *JavaScript* dapat menyempurnakan tampilan dan sistem pada halaman *web-based application* yang dikembangkan. Adapun karakteristik dari bahasa pemrograman *JavaScript* adalah: (1) Bahasa pemrograman berjenis *high-level programming*; (2) Bersifat *client-side*; (3) Berorientasi pada objek, dan (4) Bersifat *loosely typed*. Aplikasi atau *tools* standar yang digunakan dalam mengimplementasikan *JavaScript* adalah: (1) *Software text-editor*: seperti *Notepad++*, *Adobe Dreamweaver* dan *NetBeans*; (2) *Web browser*, seperti: *Mozilla Firefox*, *Google Chrome*, *Internet Explorer*, *OperaMini*, *Safari*, dan lain sebagainya; (3) HTML. Kelemahan dari bahasa pemrograman *JavaScript* ini sendiri adalah *JavaScript* tidak didukung oleh *browser* versi lama (Mariko, 2019).

Dari beberapa penjelasan pakar di atas, dapat disimpulkan bahwa *JavaScript* adalah bahasa pemrograman yang digunakan developer untuk membuat halaman *web* yang interaktif. *Javascript* bisa digunakan untuk membuat fitur beragam seperti *drag, drop* komponen yang semuanya bisa bermanfaat untuk meningkatkan tampilan (*interface*) dan pengalaman menggunakan *web*. Selain itu, programmer juga bisa memperluas fungsi halaman *web* dengan menulis *snippet Javascript* untuk *add-on* pihak ketiga,

2.6.7.3 CSS (*Cascading Style Sheet*)

CSS merupakan kependekan dari *Cascading Style Sheet* yang berfungsi untuk mengatur tampilan dengan kemampuan jauh lebih baik dari tag maupun atribut standar HTML (*Hypertext Markup Language*). CSS sebenarnya adalah suatu kumpulan atribut untuk fungsi format tampilan dan dapat digunakan untuk mangontrol tampilan banyak dokumen secara bersamaan. Keuntungan menggunakan CSS yaitu jika ingin mengubah format dokumen, maka tidak perlu mengedit satu persatu (Muhammad Saed Novendri et al., 2019).

Berdasarkan uraian gagasan ahli di atas, dapat disimpulkan bahwa *Cascading Style Sheets* (CSS) merupakan bahasa pemrograman yang digunakan untuk menentukan bagaimana dokumen dan *website* akan

disajikan. CSS berfungsi untuk menjelaskan dan menata tampilan elemen yang tertulis pada bahasa *markup*, salah satunya adalah HTML. Dalam arti lain, CSS-lah yang menjelaskan bagaimana elemen HTML di balik layar ditampilkan sedemikian rupa di layar *website*.

2.6.7.4 PHP

PHP (*Hypertext Pre-Processor*) adalah Bahasa program *open source* yang di gunakan untuk membuat sebuah aplikasi *web*. Kode program PHP akan dieksekusi oleh *server* dan hasil eksekusi tersebut akan ditampilkan kepada *client*. Selain itu, PHP merupakan *open source* yang dapat digunakan di banyak sistem operasi (*windows, linux, mac, solaris, dan lain-lain*) (Waidah & Fatmala, 2022). PHP yang saat ini dikelola oleh The PHP Group dapat berintegrasi dengan berbagai sistem basis data, seperti *MySQL, PostgreSQL, Oracle*, dan banyak lagi. Ini memungkinkan pengembang untuk mengakses, mengelola, dan memanipulasi data dalam aplikasi *web* menggunakan bahasa SQL (Satria, Ramadhani, & Sari, 2023).

Dari uraian gagasan ahli di atas, dapat disimpulkan bahwa, PHP (*Hypertext Pre-Processor*) adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk memrogram situs *web* dinamis. Bahasa ini digunakan untuk mengelola konten dinamis, database, *session tracking*, dan bahkan keseluruhan situs *e-*

commerce. Pada awalnya PHP merupakan kependekan dari *Personal Home Page (Situs personal)*.

2.6.7.5 MySQL

MySQL adalah program database server yang dapat menerima dan mengirim data dengan sangat cepat, dengan menggunakan perintah standar SQL (*Structured Query Language*). MySQL juga merupakan database server, yang bersifat *Open Source*, sehingga dalam penggunaannya dapat dipergunakan secara bebas tanpa harus membeli atau membayar lisensinya (Syahfitri et al., 2023).

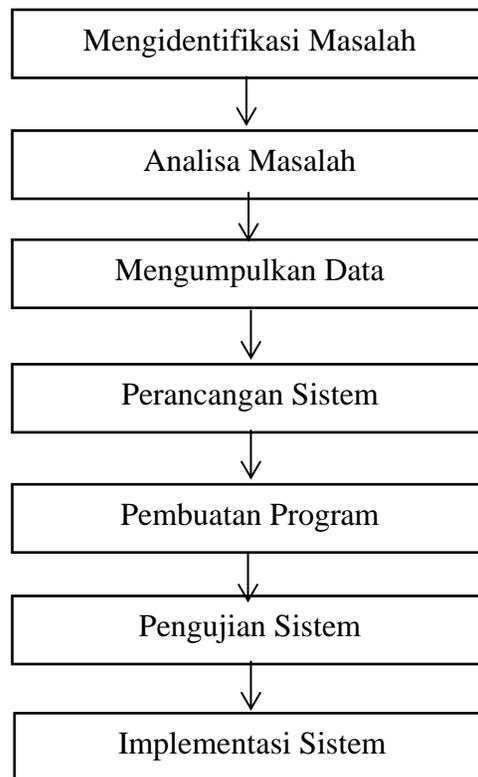
Dari pendapat pakar tersebut, dapat disimpulkan bahwa MySQL merupakan sistem manajemen database yang bersifat *open-source* yang menggunakan perintah dasar atau bahasa pemrograman yang berupa *Structured Query Language (SQL)* yang cukup populer di dunia teknologi. MySQL berguna sebagai *database*. Kelebihan dari MySQL adalah mendukung berbagai macam data yang bisa digunakan di MySQL. Contohnya *float, integer, date, char, text, timestamp, double*, dan lain sebagainya.

BAB 3

METODOLOGI PENELITIAN

3.1. Kerangka Kerja Penelitian

Pada bab ini diuraikan juga kerangka kerja penelitian yang digunakan dalam penyelesaian penelitian ini. Kerangka kerja ini merupakan tahap-tahap yang akan dilakukan dalam penyelesaian. Uraian kerangka kerja dalam penelitian ini bertujuan sebagai pedoman agar pelaksanaan penelitian berjalan terstruktur dan jelas. Berikut struktur kerangka kerja penelitian dijelaskan pada gambar 3.1 berikut ini.



Gambar 3.1 Kerangka Kerja Penelitian.

3.2. Tahapan Metode Penelitian

Berdasarkan kerangka kerja pada gambar 3.1 maka masing-masing langkahnya dapat dijabarkan seperti berikut ini:

1. Mengidentifikasi Masalah

Kerangka kerja ini dimulai dari melakukan penelitian di Pondok Pesantren Darul Furqon yang terdiri dari penelitian perpustakaan (*library research*), penelitian lapangan (*field research*), dan penelitian laboratorium (*laboratory research*).

2. Analisa Masalah

Setelah mengidentifikasi masalah selesai, selanjutnya dilakukan analisis terhadap permasalahan-permasalahan yang terjadi dalam proses pendaftaran peserta didik baru di Pondok Pesantren Darul Furqon. Hal ini bertujuan untuk melakukan pengelompokan terhadap data tersebut sehingga akan memudahkan penulis di dalam melakukan analisis berikutnya.

3. Mengumpulkan Data

Pengumpulan data dilakukan untuk memperoleh informasi yang dibutuhkan dalam rangka mencapai tujuan penelitian. Teknik pengumpulan data yang digunakan yaitu sebagai berikut:

- a. Pengamatan (*Observasi*), yaitu melakukan pengamatan langsung terhadap objek yang diteliti untuk mengetahui secara langsung

bagaimana sistem pendaftaran peserta didik baru di Pondok Pesantren Darul Furqon.

- b. Wawancara (*Interview*), yaitu melakukan tanya jawab langsung dengan pihak yang berhubungan dengan permasalahan untuk memperoleh informasi dari objek yang diteliti.

4. Analisis

Analisis dilakukan untuk kebutuhan sistem serta mengidentifikasi informasi yang dibutuhkan melalui analisis SWOT sebagai landasan bagi identifikasi masalah ataupun hambatan. Hasil identifikasi awal dijabarkan dalam bentuk narasi dan grafis serta analisis SWOT tersebut didiskusikan dengan pihak Pondok Pesantren Darul Furqon agar ditetapkan proyeksi sistem yang akan diterapkan.

5. Perancangan Sistem

Perancangan sistem merupakan tahap merancang sistem yang akan dibuat berdasarkan tahapan sebelumnya yang mengacu pada kebutuhan sistem serta pengembangan sistem yang dihasilkan sesuai dengan yang diinginkan. Dalam perancangan sistem peneliti menggunakan Aliran Sistem Informasi (ASI), *Unified Modelling Language (UML)*.

6. Pembuatan Sistem

Setelah tahapan perancangan sistem selesai dilakukan, maka selanjutnya adalah pembuatan sistem. Pembuatan sistem meliputi bagaimana sistem yang akan dibuat sehingga menghasilkan sebuah sistem yang sesuai dengan tujuan.

7. Pengujian Sistem

Pengujian merupakan sebuah tahapan yang memperlihatkan apakah peramalan tingkat akurasi dari penelitian sesuai dengan yang dibutuhkan atau tidak. Pengujian dilakukan dengan menggunakan metode *Black Box*, hal ini dilakukan meminimalisir terjadinya kesalahan dan yang dihasilkan sesuai dengan apa yang diinginkan. Pengujian *Black Box* adalah pengujian aspek fundamental sistem tanpa memperhatikan struktur logika internal perangkat lunak. Metode ini digunakan untuk mengetahui apakah perangkat lunak berfungsi dengan benar atau tidak.

8. Implementasi Sistem

Setelah lolos dari tahapan pengujian, maka tahapan selanjutnya adalah implementasi atau penerapan dari sistem yang dibuat. Namun dalam perjalannya, akan ada banyak perubahan dan kebutuhan baru sesuai dengan perkembangan.