

BAB II

TINJAUAN PUSTAKA

2.1 Aplikasi

2.1.1 Definisi Aplikasi

Menurut Kamus Besar Bahasa Indonesia (KBBI) adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman. Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari *user* (pengguna).

Menurut Roni Habibi dan Riki Karnovi (2020) Aplikasi adalah sebuah program siap pakai yang bisa dipakai untuk menjalankan sejumlah perintah dari pemecahan masalah yang memakai salah satu teknik pemrosesan data aplikasi pada sebuah komputerisasi atau smartphone dengan tujuan untuk memperoleh hasil lebih akurat dan sesuai dengan tujuan pembuatan aplikasi .

Menurut (Alfiansyah & Junianto, 2023) aplikasi secara umum adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya aplikasi merupakan suatu perangkat komputer yang siap dipakai bagi *user*. Aplikasi adalah koleksi *windows* dan objek yang menyediakan fungsi untuk aktivitas *user*, seperti pemasukan data, proses dan pelaporan. Aplikasi berisi

1. Suatu antarmuka : *Menu, window dan control window* Dimana *user* berinteraksi langsung dengan aplikasi.
2. Proses logika aplikasi : Kejadian (*event*) dan fungsi skrip yang dibuat sebagai logika aplikasi, validasi dan proses lainnya.

Menurut (Pane dkk., 2020) Aplikasi adalah perangkat lunak (*software*) atau program komputer yang beroperasi pada sistem tertentu yang diciptakan dan dikembangkan untuk melakukan perintah tertentu istilah aplikasi sendiri diambil dari bahasa inggris "*application*" yang dapat diartikan secara harfiah, aplikasi merupakan suatu penerapan perangkat lunak atau *software* yang dikembangkan untuk tujuan melakukan tugas-tugas tertentu. Proses pengembangan aplikasi melibatkan berbagai langkah, termasuk perancangan, pemrograman, pengujian dan implementasi.

2.1.2 Pengembangan Aplikasi

Pengembangan aplikasi adalah serangkaian proses yang dilakukan seorang atau sekompok programmer untuk membuat rangkaian proses dari rancangan awal aplikasi membuat *prototype* aplikasi, implementasi, pengujian akhir aplikasi hingga aplikasi selesai dan siap untuk digunakan

Menurut (Pane dkk., 2020) pengembangan aplikasi dikelompokkan menjadi tiga kelompok, diantaranya:

1. Aplikasi Dekstop, yaitu aplikasi yang hanya dijalankan perangkat PC komputer atau desktop.
2. Aplikasi Web, yaitu aplikasi yang dijalankan menggunakan komputer dan koneksi internet.
3. Aplikasi Mobile, yaitu aplikasi yang menjelakan diperangkat mobile Dimana untuk kategori ini penggunaanya sudah banyak sekali.

2.1.3 Pengelompokkan Aplikasi

Menurut (Pane dkk., 2020) pengelompokkan aplikasi dikelompokkan menjadi lima kelompok, diantaranya:

1. *System Software*, yaitu aplikasi yang dapat mengatur dan mengontrol operasi internet sistem komputer.
2. *Real Time Software*, yaitu aplikasi yang kemampuan untuk memantau, menganalisa dan mengontrol suatu peristiwa aktual yang sedang berlangsung.
3. *Bussines Software*, yaitu aplikasi yang dibuat dan dikmembangkan untuk tujuan seperti mengelola sistem keuangan.
4. *Enginerring and Scientific Software*, yaitu aplikasi yang dibuat dan dikembangkan untuk membatu manusia memecahkan masalah non algoritmik.
5. *Web Base Software*, adalah aplikasi yang bertindak sebagai media penghubung antara pengguna dengan internet.

2.1.4 Aplikasi Berbasis Web

Menurut (Kurniawan, 2022) aplikasi berbasis web adalah aplikasi yang dikembangkan dengan menggunakan bahasa pemograman HTML, PHP, CSS, JS yang membutuhkan *web server* dan *browser* untuk menjalankannya seperti *Chrome*, *Firefox*, *Opera*, *Internet Eksplorer*, *Microsoftedge* dan lain-lainnya. Aplikasi ini

dapat berjalan pada jaringan maupun internet (Jaringan LAN). Data yang terpusat dan kemudahan dalam mengakses adalah ciri utama yang membuat aplikasi web lebih banyak diminati dan lebih mudah diimplementasikan di berbagai bidang kehidupan.

Menurut (A. Saputra, 2020) aplikasi berbasis web merupakan sebuah aplikasi yang dapat diakses menggunakan web browser atau penjelajah web melalui jaringan internet yang memiliki keunggulan bahwa aplikasi berbasis web dapat dengan mudah diakses oleh pengguna tanpa harus melakukan instalasi.

2.2 Penjadwalan Bimbingan Akademik Dan Bimbingan Online

2.2.1 Definisi Bimbingan Online

Bimbingan online meliputi berbagai metode interkasi digital seperti konsultasi melalui platform aplikasi pesan instan, dan forum diskusi online. Teknologi ini tidak hanya meningkatkan ketelibatatan tetapi juga memungkinkan personalisasi layanan bimbingan berdasarkan kebutuhan individu.

2.2.2 Definisi Penjadwalan Akademik

Menurut Kamus Besar Bahasa Indonesia (KBBI) penjadwalan berasal dari kata “jadwal” yang berarti pembagian waktu berdasarkan rencana pengaturan urutan kerja, daftar atau tabel kegiatan dengan pembagian waktu pelaksanaan yang terperinci. Penjadwalan adalah proses, cara, perbuatan menjadwalkan atau memasukkan ke dalam jadwal, pembagian waktu berdasarkan rencana pengaturan urutan kerja atau disebut pembagian waktu pelaksanaan yang terperinci sesuai dengan daftar atau tabel kegiatan.

Menurut Achmad Fikri dan Indra Kanedi (2020) penjadwalan adalah pengaturan waktu dari suatu kegiatan operasi penjadwalan mencakup kegiatan mengalokasikan fasilitas, peralatan ataupun tenaga kerja bagi suatu operasi. Dalam hierarki pengambilan keputusan, penjadwalan merupakan langkah terakhir. Jadwal merupakan suatu informasi yang menunjukkan rencana kegiatan atau pekerjaan yang akan dilakukan dan pelaksana kegiatan pekerjaan yang bertujuan agar kegiatan atau pekerjaan tersebut lebih terorganisir sesuai dengan rencana.

Menurut Putra (2019) Penjadwalan akademik adalah proses menentukan waktu, tempat, dan sumber daya yang dibutuhkan untuk pelaksanaan kegiatan akademik. Penjadwalan akademik adalah proses pengalokasian waktu, tempat, dan sumber daya untuk kegiatan akademik secara optimal dan efektif (Wiyati dkk., 2019). Penjadwalan akademik adalah proses pengaturan kegiatan akademik agar berjalan dengan lancar dan efektif (Syahputri & Nasution, 2023).

2.2.3 Tujuan Penjadwalan Akademik

Menurut (Yusar, 2023) Tujuan bimbingan akademik adalah proses pemberian bantuan kepada mahasiswa dalam merencanakan, melaksanakan dan mengevaluasi studinya. Bimbingan yang dilakukan oleh dosen yang merupakan dosen yang ditunjuk oleh pimpinan perguruan tinggi untuk memberikan bimbingan akademik kepada mahasiswa. Tujuan penjadwalan bimbingan akademik adalah sebagai berikut:

1. Meningkatkan efisiensi dan efektivitas kegiatan bimbingan akademik
Penjadwalan yang baik akan memastikan bahwa kegiatan bimbingan akademik dapat berjalan dengan lancar tanpa hambatan. Hal ini dapat dicapai dengan meminimalisir terjadinya bentrok pada penjadwalan.
2. Meminimalkan bentrok jadwal
Bentrok jadwal dapat menyebabkan mahasiswa tidak dapat mengikuti kegiatan bimbingan akademik secara efektif dan efisien. Oleh karena itu, penjadwalan harus dipastikan bahwa tidak ada dua pertemuan bimbingan akademik yang berlangsung pada waktu yang sama.
3. Memanfaatkan sumber daya secara optimal
Sumber daya dapat dimanfaatkan dalam penjadwalan bimbingan akademik antara lain ruangan bimbingan, dosen dan mahasiswa. Penjadwalan yang baik akan memanfaatkan sumber daya tersebut secara optimal.
4. Memenuhi kebutuhan dan persyaratan dosen, mahasiswa dan admin Penjadwalan harus mempertimbangkan kebutuhan dan persyaratan dari berbagai pihak yang terlibat dalam kegiatan bimbingan akademik.

2.2.4 Faktor Penjadwalan Akademik

Menurut (Rukmana dkk., n.d.) Penjadwalan adalah proses pengalokasian waktu, tempat dan sumber daya untuk kegiatan akademik. Seperti perkuliahan, penelitian dan pengabdian masyarakat. Proses penjadwalan akademik harus mempertimbangkan berbagai faktor, baik dari sisi institusi pendidikan, mahasiswa, dosen maupun staf akademik lainnya. Faktor-faktor tersebut dapat dikelompokkan menjadi tiga kategori, yaitu:

1. Faktor Akademis

Faktor akademis merupakan faktor yang berkaitan dengan kegiatan akademik seperti perkuliahan, penelitian dan pengabdian masyarakat. Faktor ini penting untuk mempertimbangkan agar kegiatan akademik berjalan dengan lancar. Faktor akademis memiliki dua bagian yaitu :

a. Visi dan misi insititusi pendidikan

Visi dan misi institusi pendidikan merupakan landasan dalam penyusunan kurikulum dan kegiatan akademik lainnya. Oleh karena itu, faktor ini penting untuk dipertimbangkan dalam penjadwalan akademik.

b. Kurikulum

Kurikulum merupakan seperangkat mata kuliah yang harus ditempuh oleh mahasiswa untuk menyelesaikan studinya. Oleh karena itu faktor ini penting untuk dipertimbangkan dalam penjadwalan akademik agar mahasiswa dapat mengikuti mata kuliah yang sesuai dengan kurikulum.

c. Ketersediaan sumber daya

Sumber daya diperlukan dalam kegiatan akademik antara lain ruang kuliah, laboratorium, peralatan dan tenaga pendidik. Oleh karena itu faktor ini penting untuk dipertimbangkan dalam penjadwalan akademik agar kegiatan akademik berjalan dengan lancar.

2. Faktor non-akademis

Faktor non-akademis merupakan faktor yang berkaitan dengan kebutuhan dan kemampuan mahasiswa, dosen, dan admin lainnya. Faktor ini penting untuk dipertimbangkan agar proses penjadwalan akademik dapat diterima dari semua pihak

a. Kebutuhan mahasiswa

Kebutuhan mahasiswa antara lain waktu kuliah yang sesuai dengan jadwal kuliah dan aktivitas lainnya, jarak antara tempat tinggal dengan kampus dan ketersediaan transportasi. Oleh karena itu, faktor ini penting untuk dipertimbangkan dalam penjadwalan akademik agar mahasiswa dapat mengikuti kegiatan akademik dengan nyaman.

b. Kemampuan akademik mahasiswa

Kemampuan akademik mahasiswa dapat berbeda-beda tergantung pada latar belakang pendidikan dan kemampuannya masing masing. Oleh karena itu, faktor penting untuk dipertimbangkan dalam penjadwalan bimbingan akademik agar mahasiswa dapat mengikuti kegiatan akademik sesuai dengan kemampuannya.

c. Kebutuhan dosen

Kebutuhan dosen antara lain waktu mengajar yang sesuai dengan jadwal mengajar dan aktivitas lainnya, serta ketersediaan ruang kuliah. Oleh karena itu, faktor ini penting untuk dipertimbangkan dalam penjadwalan akademik agar dosen dapat memberikan pengajaran dengan efektif.

d. Kebutuhan admin dan lainnya

Kebutuhan staf akademik lainnya antara lain waktu untuk melakukan penelitian dan pengabdian kepada masyarakat. Faktor ini penting untuk dipertimbangkan dalam penjadwalan akademik agar admin lainnya dapat melaksanakan tugas secara efektif.

2.2.5 Manfaat Penjadwalan Akademik

Menurut (Rukmana dkk., n.d.) penjadwalan bimbingan akademik yang baik akan memberikan manfaat bagi mahasiswa, dosen dan admin, yaitu:

1. Meningkatkan kemudahan dan kenyamanan dalam mendapatkan bimbingan akademik

Bimbingan akademik yang efektif dapat membantu mahasiswa untuk mendapatkan bimbingan yang mereka butuhkan secara tepat waktu. Hal ini dapat

meningkatkan kenyamanan dan kepuasan mahasiswa dalam mengikuti bimbingan akademik.

2. Meningkatkan peluang untuk memperoleh hasil belajar yang optimal
Penjadwalan bimbingan akademik yang efektif dapat membantu dosen untuk memberikan bimbingan akademik secara efektif dan efisien. Hal ini dapat meningkatkan peluang kepada mahasiswa dalam memperoleh hasil belajar yang optimal.
3. Meningkatkan kepuasan dalam memberikan bimbingan akademik yang efektif dapat membantu dosen untuk merasa lebih puas dalam memberikan bimbingan akademik. Oleh karena itu hal ini dapat meningkatkan motivasi dosen dalam memberikan bimbingan akademik.

2.3 Algoritma FIFO (*First-In-First-Out*)

2.3.1 Definisi Algoritma FIFO

Menurut Wayan (2023) FIFO singkatan dari *First-In-First-Out* dan mengacu pada bagaimana mengatur dan menangani data dalam kaitannya dengan waktu dan prioritas. Pernyataan ini mengacu pada pendekatan mendasar dari pemrosesan antrian atau melayani permintaan yang bertentangan dengan sistem pemesanan berdasarkan *First Come First Served (FICFS)* Dimana individu meninggalkan antrian sesuai urutan kedatangannya atau menunggu antrian.

Menurut (Fadhli dkk., 2023) Algoritma FIFO merupakan sebuah algoritma untuk menerapkan disiplin antrian yang tidak berprioritas. Algoritma FIFO ini menggunakan struktur data. Algoritma FIFO sering di gunakan di berbagai macam pemecahan masalah kehidupan dan berbagai aplikasi serta teknologi yang ada. FIFO sendiri merupakan algoritma yang bersifat berurutan dan bergiliran namun tetap pada alur atau jalurnya sesuai dengan yang pertama kali masuk dan kemudian diproses sesuai dengan giliran dalam artian yang pertama kali masuk maka pertama kali keluar.

Menurut Aditya (2022) Algoritma FIFO merupakan suatu bentuk penjadwalan yang dipengaruhi oleh data yang pertama kali masuk maka data itulah yang akan di proses oleh CPU (*Central Processing Unit*) terlebih dahulu, terlepas dari prioritas data tersebut.

Menurut Hidayat dan Al Amin (2020) FIFO merupakan metode pemecahan pada masalah antrian yang dapat diterapkan dengan cara pertama kali masuk diasumsikan keluar pertama untuk merancang sistem antrian pada sebuah aplikasi.

Menurut (Asrozy dkk., 2022) First In First Out (FIFO) adalah metode yang mendahulukan pelayanan yang masuk lebih dahulu sehingga yang lebih dahulu datang maka lebih dahulu dilayani.

2.3.2 Manfaat Algoritma FIFO

Menurut Wayan (2023) Algoritma FIFO juga memiliki manfaat bagi pengguna pemrograman antrian berdasarkan waktu kedatangannya, yaitu:

1. Mudah diimplementasikan

Algoritma FIFO adalah algoritma yang sederhana dan mudah untuk diimplementasikan. Algoritma ini dapat diimplementasikan dengan menggunakan berbagai bahasa pemrograman.

2. Efisien dalam penggunaan sumber daya

Algoritma FIFO mudah untuk diverifikasi. Algoritma ini dapat diverifikasi dengan memeriksa apakah tugas-tugas dijadwalkan sesuai dengan urutan kedatangannya.

3. Mengendepankan keadilan

Algoritma FIFO mengendepankan keadilan. Algoritma ini tidak memberikan prioritas kepada tugas-tugas tertentu. Semua tugas dijadwalkan berdasarkan kedatangannya.

4. Mudah untuk diperluas

Algoritma FIFO mudah untuk diperluas. Algoritma ini dapat diperluas untuk mendukung berbagai fitur, seperti prioritas tugas, batas waktu dan penjadwalan berulang.

2.4 Sistem Antrian

2.4.1 Definisi Sistem Antrian

Menurut (Sari dkk., 2022) Proses antrian adalah suatu proses yang berhubungan dengan kedatangan konsumen pada suatu fasilitas pelayanan, kemudian menunggu dalam suatu barisan (antrian) apabila fasilitas pelayanan

sedang sibuk konsumen tersebut akan menunggu dan konsumen akan meninggalkan fasilitas pelayanan tersebut apabila sudah mendapatkan pelayanan.

Menurut Asrul Sani (2021) antrian adalah hal dapat terjadi ketika kebutuhan akan suatu layanan melebihi jumlah pelayanan yang tersedia. Rata-rata lamanya waktu menunggu (*waiting time*) sangat tergantung kepada rata-rata tingkat kecepatan pelayan (*rate of service*). Hal ini terlihat jika ada pelanggan yang mengantri suatu pelayanan karena pelayan tersebut sedang melayani pelanggan lain sehingga tidak dapat melayani lebih dari suatu pelanggan dalam waktu bersamaan.

Menurut Sari (2024) Sistem antrian bimbingan akademik adalah sistem pengaturan mahasiswa yang akan menemui dosen untuk mendapatkan bimbingan akademik. Sistem antrian bertujuan untuk meningkatkan kepuasan mahasiswa dan efisiensi dosen.

Secara umum, sistem antrian bimbingan akademik dapat didefinisikan sebagai sistem pengaturan mahasiswa yang akan menemui dosen untuk mendapatkan bimbingan akademik. Sistem antrian ini bertujuan untuk meminimalkan waktu tunggu mahasiswa, meningkatkan kepuasan mahasiswa dan meningkatkan efisiensi dosen.

2.4.2 Faktor Antrian

Sistem antrian memiliki beberapa faktor, sebagai berikut:

1. Pola kedatangan

Menurut Maman Hilman pola kedatangan menggambarkan distribusi pelanggan memasuki sistem. Distribusi kedatangan terdiri dari *onstant arrival distribution* dan *arrival pattern random*. *Constant arrival distribution* adalah pelanggan yang datang setiap periode tertentu sedangkan *Arrival pattern random* adalah pelanggan yang datang secara acak.

2. Fasilitas Pelayanan

Menurut Asrul Sani (Hilman & Liyanti, 2021) Fasilitas pelayanan terkait dengan baris antrian yang akan dibentuk. Ada tiga bentuk perancangan fasilitas pelayanan, yaitu perancangan stasiun seri *parallel* dan *network station*.

3. Distribusi layanan

Menurut Jamaluddin (2023) distribusi layanan dapat diterapkan berdasarkan salah satunya yaitu, beberapa pelanggan yang dapat dilayani persatuan waktu atau berapa banyak setiap pelanggan bisa melayani. Dalam kasus lain, distribusi mungkin digunakan untuk menentukan waktu rata-rata melayani.

4. Disiplin Antrian

Menurut Rika Listiyani (2019) Disiplin antrian menunjukkan aturan urutan kedatangan dan layanan pelanggan.

2.4.3 Manfaat Antrian Bimbingan Akademik

Menurut Devi Yuliana (Susanto, 2020) Antrian bimbingan yang akan menemui dosen untuk mendapat bimbingan akademik memiliki beberapa manfaat terhadap mahasiswa, dosen, dan institusi, yaitu:

1. Manfaat bagi mahasiswa

a. Meningkatkan kemudahan dan kenyamanan dalam mendapatkan bimbingan akademik

Antrian bimbingan dapat membantu mahasiswa untuk mendapatkan bimbingan akademik secara tepat waktu dan tempat. Hal ini dapat meningkatkan kemudahan dan kepuasan mahasiswa dalam mengikuti bimbingan akademik.

b. Meningkatkan peluang untuk memperoleh hasil belajar yang optimal Antrian bimbingan akademik dapat membantu mahasiswa untuk mengatasi kesulitan yang dihadapi dalam studinya, sehingga dapat peluang mereka untuk memperoleh hasil belajar yang optimal.

c. Meningkatkan kepuasan studi

Antrian bimbingan akademik bisa menjadikan mahasiswa merasa lebih didukung dalam studinya, sehingga dapat meningkatkan kepuasan mereka terhadap studinya, terutama dalam bimbingan akademik.

2. Manfaat bagi dosen

a. Meningkatkan efisien dan efektivitas dalam memberikan bimbingan akademik

Antrian bimbingan akademik dapat membantu dosen untuk memberikan bimbingan akademik secara efisien dan efektif.

b. Meningkatkan kepuasan dalam memberikan bimbingan akademik

Antrian bimbingan akademik dapat membantu dosen merasa lebih puas dalam memberikan bimbingan akademik.

3. Manfaat bagi institusi Pendidikan

a. Meningkatkan efisiensi dan efektivitas dalam pengelolaan bimbingan akademik

Antrian bimbingan akademik dapat membantu institusi Pendidikan untuk mengelola bimbingan akademik secara efisien dan efektif.

b. Meningkatkan kepuasan mahasiswa dan dosen terhadap institusi Pendidikan.

2.5 Website

2.5.1 Definisi Website

Menurut Susilo dan Ernatita (2022) *website* merupakan sekumpulan halaman yang dipergunakan untuk menampilkan informasi tekstual, gambar, animasi, suara dan lainnya. Adapun halaman yang bersifat statis dan dinamis, membentuk serangkaian bangunan yang saling berhubungan. Umumnya *website* disimpan di dalam *server website* dan dapat diakses melalui internet. Singkatnya, *website* adalah kumpulan folder dan file yang berisi banyak perintah dan fitur tertentu, seperti fungsi tampilan, fungsi pemrosesan, penyimpanan data, dan fungsi menampilkan data.

Menurut Sari (2021) *website* merupakan kumpulan halaman digital yang berisi informasi yang berupa teks, animasi, gambar, suara dan video atau gabungan dari semuanya yang terkoneksi oleh internet, sehingga dapat dilihat seluruh pengguna koneksi jaringan internet.

Menurut Elgamar (2023) *website* adalah suatu media yang terdiri dari beberapa halaman yang saling berkaitan satu sama lain dan berfungsi sebagai media untuk menampilkan suatu informasi, baik berbentuk gambar, video, teks, suara ataupun gabungan dari semuanya. Website bersifat *multi-platform* yang artinya dapat dibuka dari segala perangkat atau *device* yang terhubung dengan jaringan

internet. Walaupun teknologi ini sudah cukup banyak sekali digunakan, namun saat ini banyak sekali perusahaan lainnya yang masih menggunakan *website* dalam menampilkan profil perusahaan (*company profile*), menjual produk, ataupun sebagai sistem yang dapat digunakan oleh pelanggan.

2.5.2 Basis Data (*Database*)

Menurut Syaputri (2023) basis data adalah kumpulan informasi yang disimpan di dalam komputer secara sistem sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi basis data tersebut. Kegunaan utama sistem basis data adalah agar tujuan pemakai mampu menyusun suatu pandangan (*view*) abstraksi data. Hal ini bertujuan untuk menyederhanakan interaksi antara pengguna dengan sistemnya dan basis data dapat mempresentasikan pandangan yang berbeda kepada para pengguna, *programmer* dan administrasinya. Ada beberapa peran utama yang dimainkan oleh sistem *database*:

1. Penyimpanan Data

Peran utama sistem *databases* adalah menyimpan data secara terstruktur dan terorganisir.

2. Pengelolaan Data

Sistem *databases* memungkinkan pengelolaan data secara efektif ini termasuk pembuatan, pengeditan dan penghapusan data, memanipulasi data, menyaring data dan menggabungkan data dari beberapa tabel.

3. Keamanan Data

Sistem *databases* menyediakan fitur keamanan untuk melindungi data dari akses yang tidak sah dapat menjaga kerahasiaan data.

4. Konsisten Data

Sistem *databases* memastikan konsistensi data dengan menerapkan aturan integritas data.

5. Pemulihan Data

Sistem *database* menyediakan fitur pemulihan data dalam sistem, seperti kerusakan hardware atau kesalahan menggunakan teknik seperti backup dan

restore, log out, replikasi data, sistem *database* dapat memulihkan data setelah kegagalan.

6. Skalabilitas dan Kinerja

Sistem *database* dirancang untuk mengelola jumlah data yang besar dan memungkinkan skalabilitas *vertical*.

2.6 Bahasa Pemrograman

2.6.1 PHP

Menurut (Sahi, 2020), PHP atau singkatan dari Hypertext Preprocessor merupakan bahasa pemrograman Script Server-side yang disisipkan pada dokumen html dalam pengembangan web. Adapun Kelebihan dari PHP antara lain :

- a. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
- b. Web server yang mendukung PHP dapat ditemukan dimana-mana dari mulai apache, IIS, Lighttpd, nginx, hingga Xitami dengan konfigurasi lebih mudah.
- c. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.
- d. PHP adalah bahasa open source yang dapat digunakan di beberapa mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah system.



Gambar 2.1 Logo PHP

Sumber ([WordPress - Wikipedia](#))

2.6.2 MySQL

Menurut Adnan Buyung Nasution, dkk (2020) MySQL adalah salah satu *Database Management System* (DBMS) dari sekian banyak DBMS seperti *Oracle*,

MySQL, *Postgre SQL* dan lainnya. *MySQL* berfungsi mengolah database menggunakan bahasa *SQL*. *MySQL* bersifat *open source* sehingga kita bisa menggunakannya secara gratis. Pemograman *PHP* juga sangat mendukung dengan database *MySQL*.

Menurut Kusumo (2021) Pemograman *SQL server* akan mendesain dan mengimplementasikan model database relasional. Dia akan memanggil atau memodifikasi data menggunakan *query SQL* juga akan memprogram menggunakan *view*, *trigger*, *stroted procedur*, *fungsi (function)* dan *UDF (User Defined Function)*.



Gambar 2.2 Logo *MYSQL*

Sumber (Sinaga & Samsudin, 2021)

2.6.3 XAMPP

Fungsi *XAMPP* menurut Pratama (2020) adalah sebagai server yang berdiri (*localhost*) yang terdiri dari beberapa program antara lain *apache*, *MYSQL*, *database* dan penerjemah bahasa yang ditulis dengan bahasa pemograman *PHP*. Tanpa memakai *XAMPP*, file aplikasi yang memiliki format *PHP* tidak bisa terbaca, format *PHP* mengidentifikasi bahasa pemograman yang ada.

Menurut Tumini dan Fitriani (2020) *XAMPP* adalah paket instalasi program yang terdiri dari program *apache Hypertext Transporotocol (HTTP) server*, *MYSQL*, *database* penerjemah bahasa dengan bahasa pemograman *PHP* dan *perl*. *XAMPP* adalah *server* yang paling banyak digunakan untuk keperluan belajar HP secara mandiri, terutama bagi pemogrammer pemula.



Gambar 2.3 Logo Xampp

Sumber ([WordPress - Wikipedia](#))

2.6.4 *Framework*

Menurut Medina (2022) *Framework* adalah struktur konseptual dasar yang digunakan untuk memecahkan atau menangani sebuah permasalahan atau isu-isu yang kompleks. Singkatnya *framework* adalah wadah atau kerangka kerja dari sebuah *website* yang akan dibangun. Dengan menggunakan kerangka tersebut waktu yang digunakan dalam membuat *website* lebih singkat dan memudahkan dalam melakukan perbaikan.

Framework adalah struktur konseptual dasar yang berisi kumpulan fungsi untuk tujuan yang sudah siap digunakan sehingga pembuatan aplikasi digunakan dengan lebih cepat karena programnya tidak dibuat dari awal.

Framework adalah struktur konseptual dasar yang berisi kumpulan fungsi tujuan tertentu yang sudah siap digunakan, sehingga pembuatan aplikasi dapat dilakukan dengan lebih cepat karena kode programnya tidak dibuat dari awal. Adapun kelebihan dari *framework*.

1. Mudah dan dokumentasi lengkap strukturnya disediakan dalam dokumentasi.
2. *Open Source Laravel* adalah *framework open source* dapat digunakan secara gratis yang memungkinkan untuk membuat web aplikasi besar dan kompleks dengan mudah. Hanya dengan bermodalkan editor dan web server yang telah terinstal PHP untuk memulai.
3. Arsitektur MVC dengan menggunakan pola MVC kita dapat membuat struktur kode lebih rapi Dimana pola tersebut memisahkan antara pola dan view. MVC dapat meningkatkan performa dan memiliki beberapa fungsi *built-in* (fungsi yang dapat langsung digunakan) dan penulisan dokumentasi yang lebih baik.

4. *Blade Template Blade* dapat menetapkan template yang dimiliki dengan membaginya menjadi beberapa bagian sehingga lebih mudah dimaintenance.
5. Memiliki fitur *migration* yang merupakan fitur utama yang dimiliki *Laravel Migration* yang dapat mempertahankan struktur *database* yang dimiliki tanpa harus membuatnya kembali dan dapat menuliskan kode PHP untuk mengatur *database*.

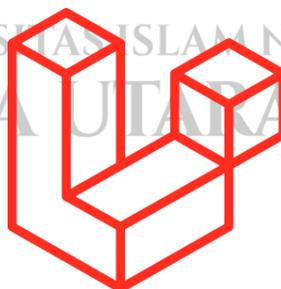
2.6.5 *Laravel*

Laravel ialah *framework* PHP yang dirilis di bawah lisensi MT dengan kode sumber yang sudah disediakan oleh *Github*, sama seperti *framework* yang lainnya. *Laravel* dibangun dengan konsep MVC (*Model-Controller-View*), kemudian *Laravel* dilengkapi dengan *command line*.

Menurut Naista (2021) *Laravel* adalah *framework* berbasis PHP yang sifatnya *open source* dan menggunakan konsep *model-view-controller*. Dalam pengguna *laravel* memiliki beberapa kekurangan salah satunya ukuran file yang cukup besar. Di dalam *laravel* terdapat file yang bersifat *default* seperti *vendor*. File tersebut tidak boleh dihapus sembarangan sehingga ukuran *website* yang dibuka berukuran besar. Selain itu dibutuhkan koneksi internet untuk instalasi dan mengunduh *library PHP* minimal 5.4 untuk menjalankannya.

Laravel dapat dibagi dua cara yaitu:

1. Menggunakan *Composer* dengan mengetik perintah *create-project*
2. *Download source code laravel* melalui *github* dengan Alamat <https://github.com/laravel/laravel>.



Gambar 2.4 *Laravel*

Sumber <https://github.com/laravel/laravel>.

2.6.6 UML (*Unified Modeling Language*)

Menurut Putra dan Andriani (2019) UML juga salah satu standar bahasa yang banyak digunakan dunia perindustrian untuk mendefinisikan *requirement*, membuat analis desain, serta menggambarkan arsitektur dalam pemograman beorientasi objek.

Menurut Abdillah (2020) UML adalah sebuah bahasa pemodelan perangkat lunak yang telah di standarisasi sebagai media penulisan cetak bitu (perangkat lunak, UML bisa saja digunakan visualisasi, spesifikasi, kontruksi dan dokumentasi) beberapa bagian-bagian dari sistem yang ada dalam perangkat lunak. Dalam kata lain seperti halnya seorang arsitektur dalam membuat cetak biru yang digunakan oleh perusaha konstruksi dan membangun sebuah bangunan, arsitek perangkat lunak membuat diagram-diagram UML untuk membantu *programmer* atau *developer* membangun perangkat lunak.

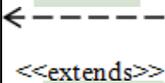
2.6.7 *Use Case Diagram*

Menurut Menurut Triase (2020) *Use Case Diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. *Use case diagram* berfungsi untuk melakukan pekerjaan tertentu dalam menggambarkan bisnis pada proses sistem itu sendiri, *diagram use case* juga dapat digunakan untuk menjelaskan fitur yang akan digunakan.

Tabel 2.1 Simbol-simbol Use Case Diagram

Sumber (Irianto dkk., 2021)

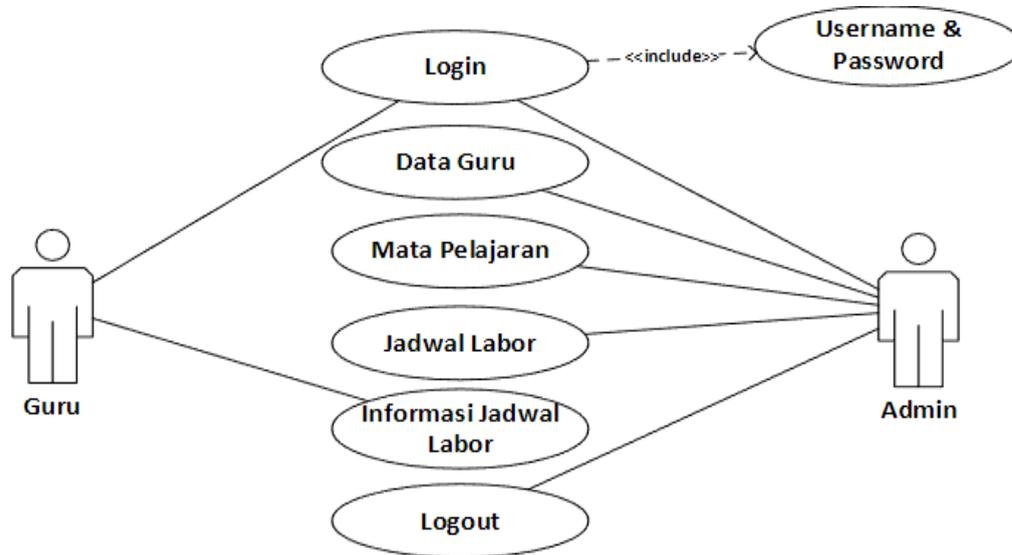
Gambar	Keterangan
	<p><i>Use case</i> menggambarkan fungsionalitas yang disediakan system sebagai unit-unit yang saling bertukar pesan antar unit dengan aktor yang dinyatakan dengan menggunakan kata kerja.</p>

	<p><i>Actor</i> atau Aktor adalah <i>Abstraction</i> dari orang sistem lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktir, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>Use Case</i>, tetapi tidak memiliki kontrol terhadap <i>use case</i>.</p>
	<p>Asosiasi antara <i>actor</i> dan <i>use case</i>, digambarkan dengan garis Tanpa panah megindikasikan siapa atau apa yang meminta interaksi secara langsung bukan mengindikasikan data.</p>
	<p>Asosiasi anantara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.</p>
	<p><i>Include</i> oleh use case lain, contohnya adalah pemanggilan Sebuah fungsi program.</p>
	<p><i>Extends</i>, merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.</p>

Berikut contoh gambar *Use Case Diagram*

Menurut Muhamad Alda (2020) *Use Case Diagram* menggambarkan interaksi yang terjadi antara aktor dengan aplikasi yang dibangun.

SUMATERA UTARA MEDAN



Gambar 2.5 Use Case Diagram

Sumber (Alda, 2020)

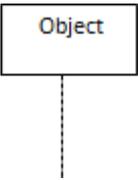
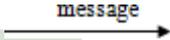
2.6.8 Sequence Diagram

Menurut Putra dan Andriani (2022) *Sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dari pesan yang dikirimkan dan diterima antara objek. Gambar *sequence diagram* dibuat minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada *sequence diagram* sehingga semakin banyak *use case* didefinisikan. Maka *sequence diagram* yang harus dibuat juga semakin banyak.

Menurut (Irianto dkk., 2021) *sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek atau komponen dalam suatu sistem atau proses secara visual. Diagram ini menampilkan urutan tindakan dan komunikasi antara berbagai komponen sistem pengguna, proses, subsistem dan perangkat keras.

Tabel 2.2 Simbol-simbol *Sequence Diagram*

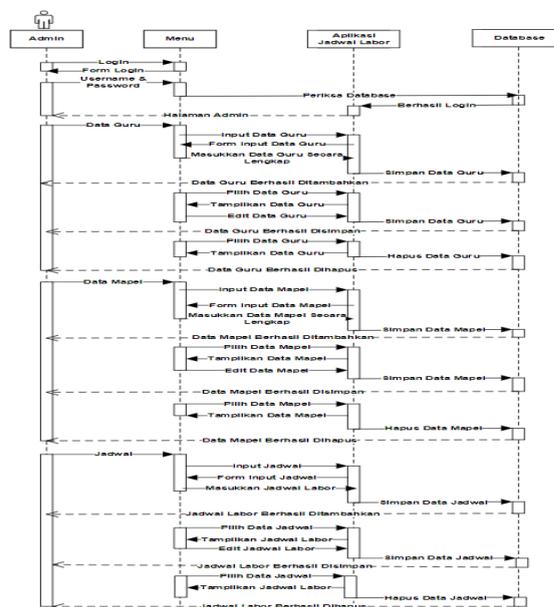
Sumber (Irianto dkk., 2021)

Simbol	Nama	Keterangan
	<i>Object/Actor</i>	Sebuah objek yang berasal dari kelas atau dapat dinamai dengan kelasnya saja. Aktor termasuk objek, garis-garis putus menunjukkan garis suatu objek Sebuah objek berasal dari kelas
	<i>Activity</i>	Untuk memperlihatkan masing-masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Message</i>	Interaksi antara satu objek dengan objek lainnya. Interaksi antar objek ditunjukkan pada bagian operasi pada diagram kelas.

Berikut contoh gambar Sequence Diagram

Sequence diagram adalah diagram yang digunakan untuk menggambarkan urutan interaksi antar objek atau komponen dalam suatu sistem atau proses secara visual. Diagram ini menampilkan urutan Tindakan dan komunikasi antar berbagai komponen sistem seperti pengguna, proses, subsistem dan perangkat keras. Sequence memiliki tiga komponen, yaitu: *Life line* (Garis vertical), *Message* (Panaah komunikasi), *Action box* (Persegi Panjang mewakili waktu aktif objek)

Menurut Muhamad Alda (Alda, 2020) Sequence Diagram menggambarkan bagaimana klien berasosiasi dengan aplikasi untuk mendapatkan data yang dibutuhkan.



Gambar 2.6 *Sequence Diagram*

Sumber (Alda, 2020)

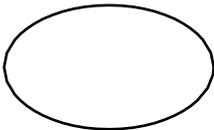
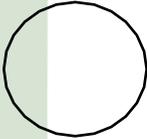
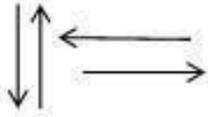
2.6.9 *Activity Diagram*

Activity diagram merupakan diagram yang menggambarkan *workflow* atau aktivitas dari sebuah sistem yang ada pada perangkat lunak. Runtutan proses dari suatu sistem digambarkan secara vertikal. *Activity diagram* merupakan pengembangan dari *use case* yang memiliki alur aktivitas. Alur atau aktivitas bisa berupa menu – menu yang terdapat di dalam sistem tersebut. Dalam pembuatan *activity diagram* hanya dapat dipakai untuk menggambarkan alur kerja atau aktivitas sistem saja (Irianto dkk., 2021). Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

1. Rancangan proses bisnis setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem.
2. Urutan atau pengelompokan tampilan dari sistem /*user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan *interface* tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
4. Rancangan menu yang ditampilkan pada perangkat lunak.

Tabel 2.3 Simbol-simbol *Activity Diagram*

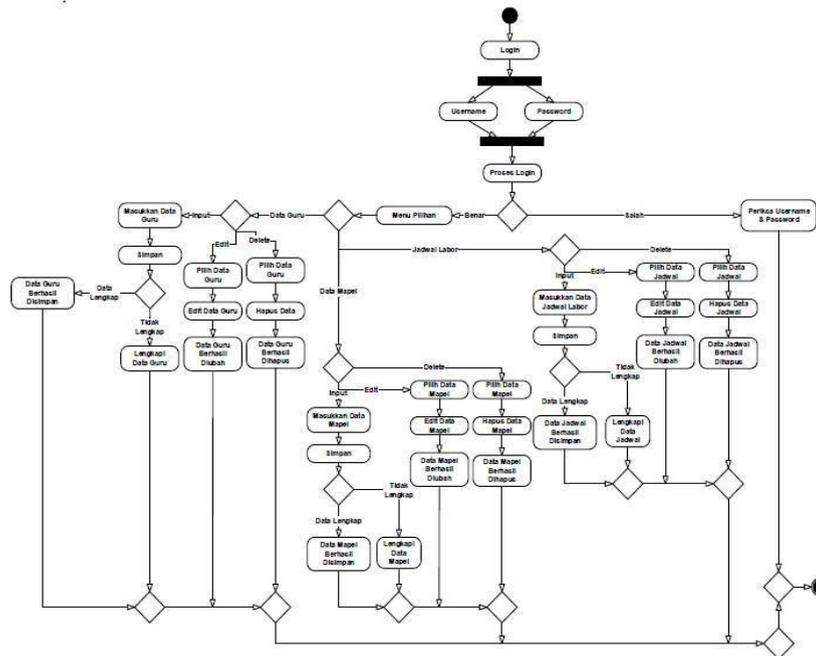
Sumber (Irianto dkk., 2021)

Simbol	Nama	Keterangan
	<i>Activity</i>	Memperlihatkan bagaimana masing masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi.
	<i>Initial Node</i>	Bagaimana objek dibentuk dan diawali.
	<i>Activity Final Node</i>	Bagaimana objek dibentuk dan diakhiri.
	<i>Decision</i>	Digunakan untuk menggambarkan keputusan tindakan yang harus diambil pada kondisi tertentu.
	<i>Line Connector</i>	Digunakan untuk menghubungkan satu simbol dengan simbol lain.

Berikut contoh *Activity Diagram*

Menurut Muhamad Alda (Alda, 2020) Diagram aktivitas menggambarkan urutan alur aktivitas yang digunakan untuk menggambarkan suatu aktivitas yang

terbentuk dalam satu aktivitas, sehingga dapat digunakan juga pada aktivitas lainnya.



Gambar 2.7 Activity Diagram

Sumber (Alda, 2020)

2.6.10 Class Diagram

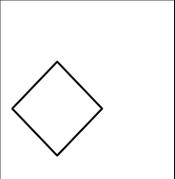
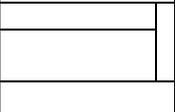
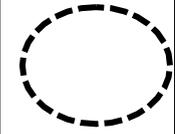
Menurut Muhamad Alda (Alda, 2020) diagram kelas adalah grafik yang memberikan gambaran hubungan dari setiap kelas atau tabel pada sebuah basis data.

Menurut Satri (Imanuel dkk., 2022) *class diagram* adalah diagram yang digunakan untuk menggambarkan stuktur dan perilaku suatu sistem. Secara umum, *class diagram* didefenisikan sebagai diagram yang menggambarkan struktur kelas dan hubungan antar kelas dalam suatu sistem. *Class diagram* digunakan untuk memahami sistem secara keseluruhan dan merancang sistem.

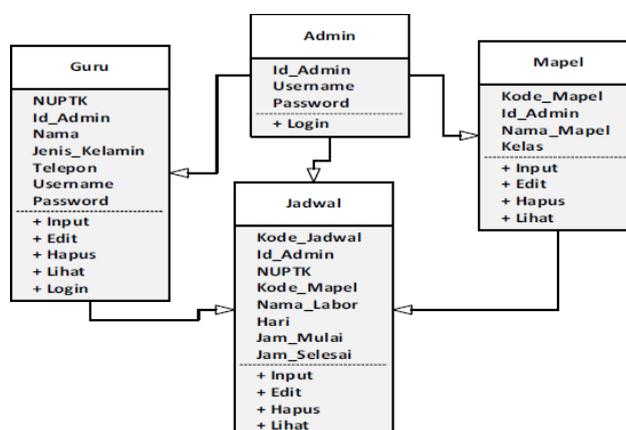
Tabel 2.4 Simbol-simbol *Class Diagram*

Sumber (Irianto dkk., 2021)

No	Gambar	Nama	Keterangan
----	--------	------	------------

1		<i>Generalization</i>	Hubungan dimana objek (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang di atasnya objek (<i>ancestor</i>)
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
3		<i>Class</i>	Himpunan dari urutan objek-objek yang berbagi atribut serta operasi yang sama
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh satu objek
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri akan mempengaruhi elemen yang bergantung elemen tidak mandiri
7		<i>Association</i>	Menghubungkan antara objek satu dan objek lainnya

Berikut contoh gambar *Class Diagram*



Gambar 2.8 *Class Diagram*

Sumber (Alda, 2020)

2.6.11 *Flowchart*

Flowchart atau Diagram Alir merupakan gambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. Diagram alir (*flowchart*) adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika (Irwanto, 2021). Diagram alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi. *Flowchart* terdiri dari lima jenis, setiap jenis memiliki karakteristik yang berbeda dalam penggunaannya, Berikut adalah jenis-jenis *flowchart*:

1. *Flowchart* Dokumen

Flowchart dokumen (*document flowchart*) atau yang juga disebut sebagai *paperwork flowchart*, digunakan untuk menelusuri alur dokumen atau form dari satu departemen atau bagian ke bagian lainnya. Fungsi *flowchart* ini meliputi proses pencatatan, pengolahan, dan penyimpanan dokumen atau laporan yang terjadi di dalam organisasi atau perusahaan.

2. *Flowchart* Program

Flowchart tersebut secara terperinci menjelaskan prosedur dari proses program. Terdapat dua jenis *flowchart* program, yaitu *flowchart* logika program (*program logic flowchart*) dan *flowchart* program komputer terinci (*detailed computer program flowchart*).

3. *Flowchart* Proses

Flowchart proses merupakan teknik visualisasi yang digunakan dalam rekayasa industri untuk memecah dan menganalisis tahapan-tahapan yang akan dilakukan dalam suatu prosedur atau sistem.

4. *Flowchart* Sistem

Flowchart sistem merupakan suatu jenis *flowchart* yang menunjukkan seluruh tahapan atau proses kerja yang terjadi dalam sistem secara menyeluruh. *Flowchart* ini juga menjelaskan urutan dari setiap prosedur yang ada dalam sistem tersebut.

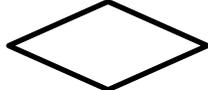
5. *Flowchart* Skematik

Mirip dengan *flowchart* sistem, *flowchart* ini menunjukkan aliran proses dalam suatu sistem. Namun, ada variasi dalam penggunaan simbol untuk mendeskripsikan plot. Untuk membuat *flowchart* skematis lebih mudah, gambar komputer dan alat lain digunakan selain simbol.

Tabel 2.5 Simbol-simbol *Flowchat*

Sumber (Irianto dkk., 2021)

Simbol	Nama	Keterangan
	<i>Terminator</i>	Mulai atau selesai.
	<i>Proses</i>	Menyatakan proses terhadap data.

Simbol	Nama	Keterangan
	<i>Input/Output</i>	Menerima <i>input</i> atau menampilkan <i>output</i>
	Seleksi/Pilihan	Memilih aliran berdasarkan syarat.
	<i>Predefinend data</i>	Definisi awal dari variable atau data.
	<i>Predefined proses</i>	Lambang fungsi atau sub program.
	<i>Connector</i>	Penghubung.
	<i>Off-page connector</i>	Penghubung halaman pada halaman yang berbeda.
	<i>Keyboard</i>	Menunjukkan input data secara manual menggunakan online keyboard.
	<i>Printer</i>	<i>Input</i> berasal dari dokumen dalam bentuk kertas, atau <i>output</i> yang perlu dicetak di atas kertas.
	<i>File/Storage</i>	Untuk menyatakan input yang berasal dari disk atau disimpan ke disk.
	<i>Display/Monitor</i>	<i>Output</i> yang digunakan yaitu layar, plotter, printer dan sebagainya.

Simbol	Nama	Keterangan
	<i>Magnetic Tape</i>	<i>Input</i> berasal dari pita magnetik atau output disimpan ke pita magnetik.
	<i>Magnetic Disk</i>	Untuk input atau output yang menggunakan disk magnetic
	<i>Sorting</i>	Menunjukkan proses pengurutan data diluar proses komputer.
	<i>Extract</i>	Untuk mengekstrak proses pemecahan atau algoritma.
	<i>Merger</i>	Menunjukkan bahwa data dalam simbol ini akan disimpan dalam suatu media tertentu.

2.6.12 Penelitian Terdahulu

Adapun penelitian yang telah dilakukan sebelumnya sebagai berikut :

1. Peneliti yang dilakukan oleh (Noviana dkk., 2018), dengan judul “Aplikasi Bimbingan Mahasiswa Berbasis *Website*”. Penelitian ini bertujuan mengembangkan sebuah sistem bimbingan akademik yang membantu para mahasiswa dan dosen secara online tidak secara *offline* sehingga tidak adanya pertemuan antarmuka antar dosen dan mahasiswa.
2. Peneliti yang dilakukan oleh (Putri dkk., 2020) dengan judul “Rancang Sistem Antrian Pelayanan Mahasiswa di Fakultas Teknik Universitas Persada Y.A.I.”. Penelitian ini bertujuan antri pelayanan bimbingan mahasiswa agar tidak terjadi kerusuhan saat diruangan dosen.
3. Peneliti yang dilakukan oleh (Susanto, 2020) dengan judul “Sistem Antrian Bimbingan Skripsi Pada Fakultas Ekonomi Dan Bisnis Muhammdiyah

Magelang”. Penelitian ini bertujuan mendata nama-nama yang akan mengantri saat bimbingan.

4. Penelitian yang dilakukan oleh (Rumende, 2019) dengan judul “Pembuatan Aplikasi Jadwal Bertemu Dosen Dan Mahasiswa Teknik Informatika Universitas Surabaya Berbasis *Website*”. Penelitian ini hanya menggunakan bahasa pemrograman *MYSQL dan XAMPP*.
5. Penelitian yang dilakukan oleh (Oktafiantoro dkk., 2023) dengan judul “Implementasi Sistem Bimbingan Skripsi Berbasis Web Pada Prodi Teknik” penelitian ini menggunakan metode waterfall dalam pengembangan sistemnya.
6. Penelitian yang dilakukan oleh (Oktafiantoro dkk., 2023) dengan judul “Pengembangan Sistem Informasi Kehadiran Dosen Jurusan Teknik Informatika Fakultas Teknik Universitas Negeri Gorontalo Berbasis Web” dengan hasil sistem kehadiran dosen yang ditampilkan secara realtime yang mempermudah siswa untuk memperoleh informasi kehadiran dosen.
7. Penelitian yang dilakukan oleh (Nazareta dkk., 2021) dengan judul “Metode Antrian First In First Out Berbasis Website Pada Sistem Reservasi Gadget”. Penelitian ini menggunakan metode FIFO untuk sistem reervasi gadget agar lebih cepat.
8. Penelitian yang dilakukan oleh (Nurokhim & Mulyanti, 2023) dengan judul “Sistem Informasi Bimbingan Konseling Tugas Akhir Berbasis (Website)”. Penelitian ini menggunakan metode *Object Oriented Analysis and Design* (OOAD) untuk sistem informasi bimbingan konseling untuk mengatasi masalah jadwal bimbingan tugas akhir.
9. Penelitian yang dilakukan oleh (Fakhri & Delianti, 2021) dengan judul “Perancangan dan Implementasi Sistem Informasi Bimbingan Tugas Akhir Online”. Penelitian ini menghasilkan sistem informasi bimbingan tugas akhir yang mempermudah dosen dan mahasiswa bertukar inforamsi terkait bimbingan tugas akhir.
10. Penelitian ysng dilakukan oleh (Ismail dkk., 2024) dengan juduk “Implementasi Algoritma Fifo Terhadap Sistem Antrian Pasien di Rumah Sakit Berbasis Web Online”. Penelitian ini menerapkan metode FIFO untuk sistem

antrian pasien rumah sakit yang berguna untuk memonitoring pasien yang mengantri berdasarkan nomor antrian yang sedang berjalan sehingga mempermudah admin untuk memanggil nomor antrian pasien.

Perbedaan penelitian yang dilakukan oleh penulis dibandingkan dengan dilakukan peneliti adalah penulis membangun aplikasi dengan algoritma FIFO menggunakan proses antrian penjadwalan akademik.



UNIVERSITAS ISLAM NEGERI
SUMATERA UTARA MEDAN