

## BAB II TINJAUAN PUSTAKA

### 2.1 Penelitian Terdahulu

Penelitian terdahulu mengenai pengembangan sistem berbasis web untuk penjadwalan dan optimasi waktu telah banyak dilakukan, terutama yang berhubungan dengan penggunaan algoritma genetika dalam pengelolaan jadwal mata kuliah. Penelitian pertama oleh Thomas Kristian Jeriko et al. (2022) membahas penerapan algoritma genetika dalam sistem informasi penjadwalan mata kuliah berbasis web di STIKOM Yos Sudarso Purwokerto. Tujuan dari penelitian tersebut adalah untuk meningkatkan efisiensi dalam proses penyusunan jadwal perkuliahan yang sebelumnya dilakukan secara manual. Sistem yang dikembangkan dengan *framework CodeIgniter* dan menggunakan PHP serta MySQL terbukti mampu menyelesaikan penjadwalan secara lebih cepat, dengan pengurangan waktu penyusunan dari rata-rata tujuh hari menjadi 31,26 detik (Jeriko et al., 2022).

Penelitian kedua dilakukan oleh Ryan Eka Wiratna et al. (2023), yang berfokus pada pengembangan aplikasi penjadwalan kuliah berbasis web dengan menerapkan algoritma genetika. Aplikasi ini dikembangkan untuk mengotomatisasi proses penjadwalan yang sebelumnya memakan waktu lama dan sering terjadi bentrokan jadwal. Dengan penerapan tahapan algoritma genetika, seperti inisialisasi, seleksi, *crossover*, dan mutasi, penelitian ini berhasil menghasilkan jadwal yang optimal tanpa adanya bentrokan antar kelas atau waktu, dengan waktu rata-rata penjadwalan hanya 2,8 menit (Wiratna, Ryan Eka, Nurlaili & Rizki, 2023).

Penelitian ketiga oleh Eka Yulia Sari et al. (2023) juga melakukan penelitian tentang optimasi penjadwalan mata kuliah menggunakan algoritma genetika. Penelitian ini dilakukan di Fakultas Teknik Universitas Sarjanawiyata Tamansiswa dengan tujuan mengurangi kompleksitas penjadwalan manual yang dilakukan sebelumnya. Dalam penelitian ini, algoritma genetika digunakan untuk mengoptimalkan penggunaan sumber daya seperti ruang, waktu, dan dosen, yang sebelumnya menimbulkan banyak bentrokan. Hasilnya adalah model sistem

penjadwalan yang lebih efektif dan efisien dibandingkan metode manual yang digunakan sebelumnya (Eka Yulia Sari et al., 2023).

Penelitian keempat oleh Andreas Johan et al. (2024) mengkaji penerapan Algoritma Genetika untuk penjadwalan mata kuliah di Program Studi Teknik Informatika STMIK Widya Cipta Dharma. Studi ini bertujuan untuk menyelesaikan masalah bentrokan jadwal antara mata kuliah wajib dan pilihan serta keterbatasan ruang kuliah. Dengan menggunakan Algoritma Genetika, penelitian ini menyusun jadwal secara otomatis dengan mempertimbangkan parameter seperti jumlah populasi, maksimal generasi, probabilitas *crossover*, dan probabilitas mutasi. Parameter terbaik yang diperoleh adalah jumlah populasi sebanyak 30, maksimal 5.000 generasi, probabilitas *crossover* antara 70-80%, dan probabilitas mutasi sebesar 15%. Implementasi algoritma ini mampu mengurangi waktu penyusunan jadwal dari beberapa hari menjadi sekitar 37 detik (Johan, 2024).

Penelitian kelima oleh Halimil Fathi (2023) di Politeknik Enjinering Indorama juga pernah melakukan penelitian yang berfokus pada pengembangan sistem informasi penjadwalan berbasis Algoritma Genetika. Penelitian ini memecahkan masalah sulitnya menentukan jadwal kuliah yang sesuai dengan jadwal dosen dan kapasitas ruang. Algoritma Genetika digunakan untuk mengintegrasikan komponen-komponen seperti mata kuliah, dosen, ruangan, dan waktu, yang menghasilkan jadwal yang lebih teratur dan terstruktur. Proses optimasi yang diterapkan mencakup pencarian nilai *fitness*, seleksi, *crossover*, dan mutasi. Hasil penelitian menunjukkan bahwa sistem yang dihasilkan lebih unggul daripada metode manual yang sebelumnya diterapkan (Fathi, M.Kom, 2023).

Penelitian keenam oleh Galih Wana Sumantri et al. (2022) di Universitas Sangga Buana menunjukkan bahwa penerapan Algoritma Genetika dalam penjadwalan akademik dapat mengatasi konflik penjadwalan ruang dan waktu dengan lebih efisien. Penelitian ini menyimpulkan bahwa Algoritma Genetika dapat menangani masalah penjadwalan yang kompleks dengan meminimalkan risiko bentrokan, yang pada akhirnya memperbaiki proses manajemen akademik. Penelitian ini juga menyoroti bahwa implementasi Algoritma Genetika memberikan fleksibilitas yang lebih baik dalam penyesuaian jadwal, terutama ketika terdapat

perubahan mendadak yang sering kali sulit diakomodasi oleh metode manual (Sumantri & Laluma, 2022).

Penelitian ketujuh oleh Hidayat et al. (2023) dari Universitas Bina Insan meneliti penggunaan Algoritma Genetika dalam penjadwalan mata kuliah yang sebelumnya dilakukan secara semi-manual. Penelitian ini menunjukkan bahwa Algoritma Genetika tidak hanya mempercepat proses penjadwalan, tetapi juga mengurangi kesalahan dalam penjadwalan yang sering terjadi akibat input manual. Sistem yang dikembangkan berhasil menghasilkan jadwal yang optimal dan bebas dari bentrokan, serta memberikan fleksibilitas lebih dalam hal penyesuaian jadwal (Toyib Hidayat et al., 2023)

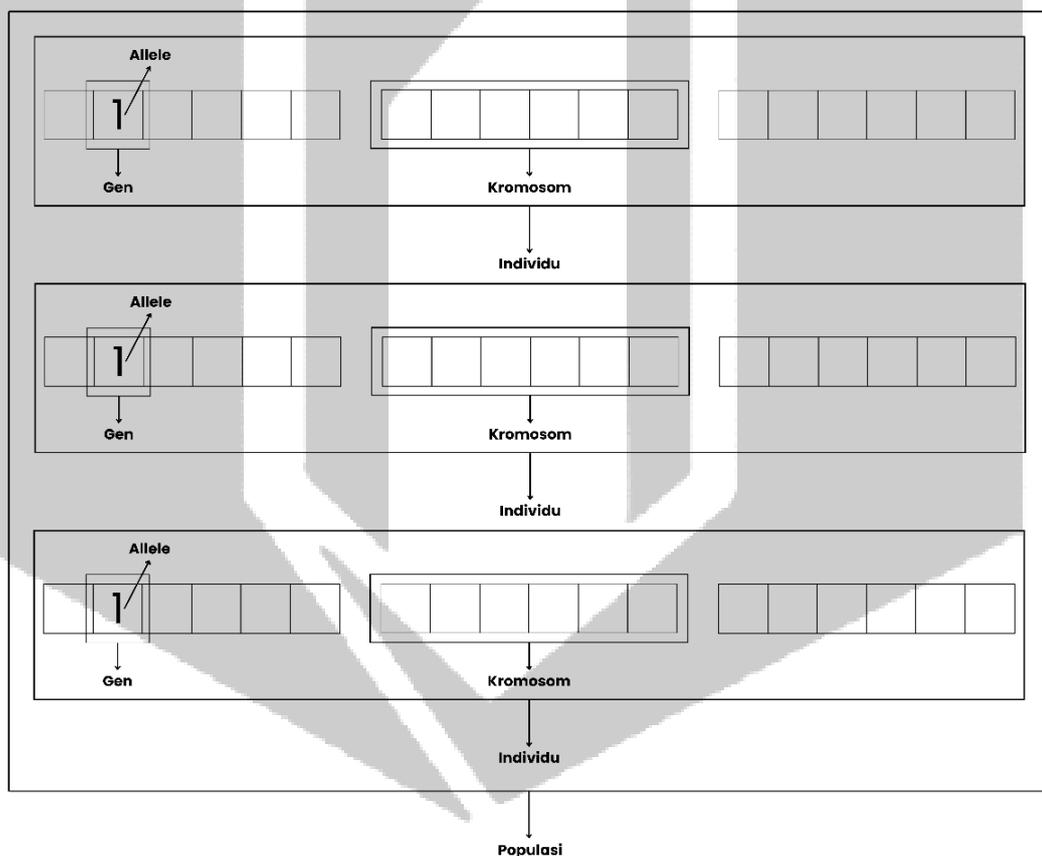
Penelitian ini menghadirkan beberapa pembaruan signifikan dibandingkan penelitian sebelumnya terkait pengembangan sistem penjadwalan berbasis algoritma genetika, dengan studi kasus di UHW Perbanas yang memiliki kompleksitas unik dalam mengelola jadwal mata kuliah untuk berbagai program studi, jenis mata kuliah (teori dan praktikum), serta ruangan khusus. Aplikasi ini dikembangkan menggunakan *framework CodeIgniter*, yang berbeda dari penelitian terdahulu yang umumnya menggunakan *platform* tanpa struktur *Model-View-Controller* (MVC). Penggunaan *CodeIgniter* mendukung pemisahan antara logika bisnis, tampilan, dan data sehingga mempermudah pengelolaan serta memungkinkan penyesuaian data secara *real-time*. Selain menghindari bentrokan jadwal, aplikasi ini memiliki fitur dosen berhalangan mengajar, dimana dengan fitur tersebut dosen memungkinkan penjadwalan sesuai waktu dosen, fitur yang tidak selalu ditemukan dalam penelitian sebelumnya. Algoritma genetika dalam sistem ini juga dirancang dengan parameter yang disesuaikan untuk memenuhi kondisi dan batasan khusus di lingkungan akademik UHW Perbanas. Penelitian ini dapat menghasilkan aplikasi yang tidak hanya lebih praktis dibandingkan metode manual, tetapi juga lebih responsif terhadap kebutuhan akademik spesifik, menunjukkan pembaruan dalam penggunaan *framework CodeIgniter* yang akan memberikan nilai tambah.

## 2.2 Landasan Teori

Landasan teori penelitian ini menjelaskan semua teori terkait yang digunakan dalam penerapan algoritma genetika untuk penjadwalan mata kuliah pada Program Studi Sistem Informasi UHW Perbanas.

### 2.2.1 Algoritma Genetika

Algoritma Genetika (AG) merupakan sebuah metode optimasi dan pencarian yang mengambil inspirasi dari prinsip seleksi alam dan genetika. Tujuan utama dari algoritma ini adalah menemukan solusi terbaik untuk masalah yang kompleks dengan cara mengiterasi populasi solusi, yang disebut sebagai individu atau kromosom, melalui proses evolusi yang melibatkan seleksi, *crossover*, dan mutasi untuk mendapatkan hasil yang lebih optimal (Deb, 1999). Dalam bukunya "Algoritma Genetika dan Penerapannya" (Lado, 2015), dijelaskan beberapa istilah penting yang menjadi dasar dalam penerapan AG untuk penyelesaian masalah. Gambar 2.1 di bawah ini menggambarkan istilah-istilah yang digunakan dalam algoritma genetika, yang kemudian akan dijelaskan secara lebih rinci.



Gambar 2.1 Istilah-istilah pada Algoritma Genetika

### 1. Kromosom

Kromosom adalah kumpulan gen yang bergabung membentuk sebuah entitas yang lebih besar.

### 2. Gen

Gen adalah unit terkecil yang membentuk suatu makna dalam kromosom dan dapat berupa nilai biner, angka desimal, integer, atau kombinasi karakter lainnya.

### 3. Allele

Allele adalah nilai dari suatu gen tersebut.

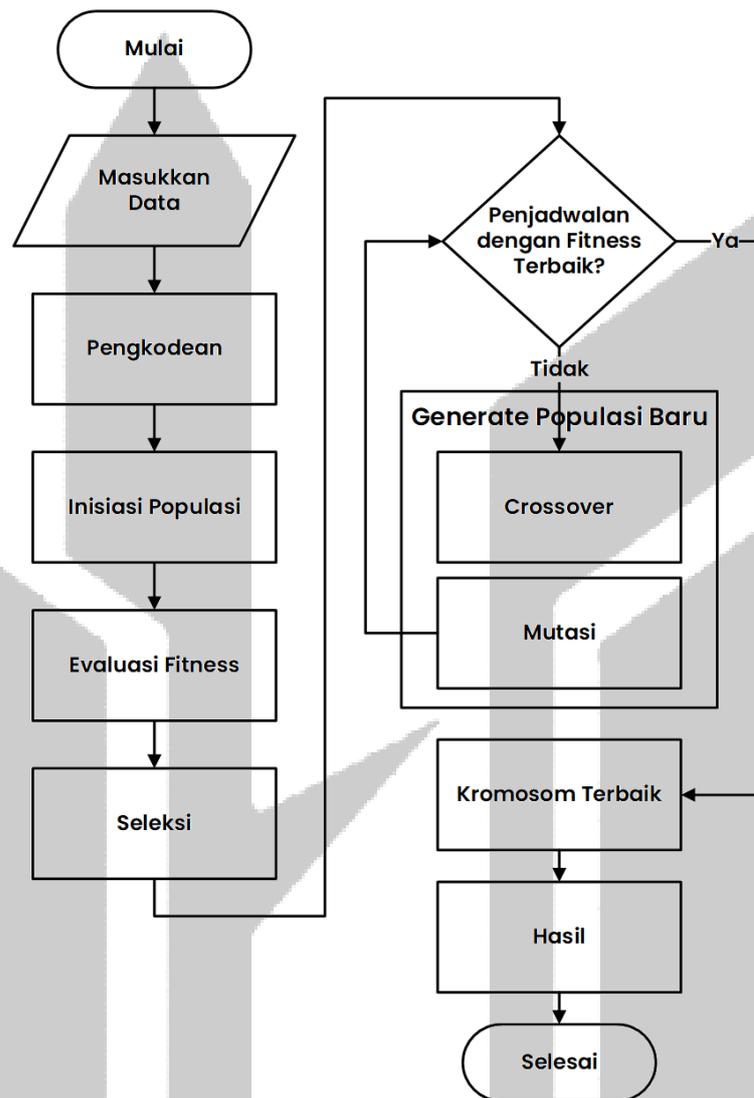
### 4. Individu

Individu adalah kumpulan dari beberapa kromosom.

### 5. Populasi

Populasi adalah sekumpulan individu yang bersama-sama menjalani proses evolusi dalam algoritma genetika.

Algoritma genetika berupaya untuk meningkatkan kualitas solusi yang dihasilkan sesuai dengan elemen-elemen tersebut. Melalui mekanisme evolusi seperti seleksi individu terbaik, persilangan bagian dari kromosom (*crossover*), dan penerapan variasi baru dengan mutasi, algoritma ini meniru proses seleksi alam untuk mencapai solusi yang lebih baik secara berkelanjutan. Beberapa proses umum yang digunakan dalam AG yang ditunjukkan pada gambar 2.2 berikut.



**Gambar 2.2 Alur Algoritma Genetika Penjadwalan Mata Kuliah**

(Sumber: Saud et al., 2017)

### 1. Pengkodean

Pengkodean adalah tahap dimana setiap solusi diubah menjadi semacam "kode" yang disebut kromosom. Pengkodean ini penting untuk memetakan masalah yang ingin diselesaikan ke dalam format yang dapat diproses oleh algoritma (Ardiansyah & Junianto, 2022).

### 2. Inisiasi Populasi

Tahap inisiasi populasi dimulai dengan membuat kumpulan solusi acak, bisa diibaratkan seperti generasi pertama dalam sebuah populasi. Setiap solusi

(atau individu) mewakili kemungkinan cara menyelesaikan masalah yang dihadapi (Ardiansyah & Junianto, 2022).

### 3. Evaluasi *Fitness*

Evaluasi ini mengukur kinerja setiap individu menggunakan fungsi *fitness*, yang diukur dengan rumus *fitness*. *Fitness* menentukan seberapa baik solusi tersebut dalam menyelesaikan masalah yang dihadapi. Nilai *fitness* yang lebih tinggi menunjukkan solusi yang lebih baik (Deb, 1999).

### 4. Seleksi

Dari semua solusi yang ada, langkah selanjutnya adalah memilih beberapa solusi terbaik, yaitu solusi yang memiliki nilai *fitness* paling tinggi. Proses ini bertujuan untuk menentukan kromosom mana yang akan dipertahankan untuk perhitungan pada tahap berikutnya. Salah satu metode yang sering digunakan untuk proses seleksi ini adalah *Roulette Wheel Selection*, di mana peluang atau probabilitas setiap individu untuk dipilih menjadi bagian dari populasi baru dihitung menggunakan rumus tertentu.

### 5. *Crossover* (Penyilangan)

Dari solusi yang terpilih, kita gabungkan sebagian solusi yang telah ada, seperti satu solusi dengan solusi lain. *Crossover* ini dilakukan dengan harapan bisa menciptakan solusi yang lebih baik.

### 6. Mutasi

Setelah *crossover*, dilakukan proses mutasi untuk mengubah gen secara acak untuk menghindari semua solusi menjadi terlalu mirip.

### 7. Kriteria Penghentian

Semua proses yang dilakukan akan terus berulang sampai ditemukan solusi yang cukup baik atau sudah mencapai batas tertentu, misalnya telah mencapai jumlah percobaan yang dilakukan.

## 2.2.2 Penjadwalan

Penjadwalan adalah proses menentukan waktu, sumber daya, atau aktivitas yang akan dilakukan dalam mencapai kelancaran dan keberhasilan suatu kegiatan (Simarangkir, 2021). Pada dasarnya, penjadwalan mengatur alur berbagai kegiatan agar dapat berjalan dengan baik tanpa terjadi konflik. Penjadwalan dapat diterapkan

dalam berbagai konteks, salah satunya dalam lingkup akademik pada perguruan tinggi adalah penjadwalan mata kuliah.

Penjadwalan mata kuliah sendiri merupakan proses penyusunan aktivitas perkuliahan yang dilaksanakan secara rutin setiap pergantian semester (Mone & Simarmata, 2021). Penjadwalan mata kuliah diatur dengan mempertimbangkan berbagai variabel, seperti ketersediaan dosen, ruang kelas, waktu, dan mata kuliah yang ditawarkan dengan tujuan untuk menghindari konflik, seperti kesamaan waktu antar ruangan atau dosen yang dijadwalkan mengajar di dua tempat pada waktu yang sama.

### 2.2.3 Web

*Website* adalah kumpulan halaman yang saling terhubung, umumnya berada dalam satu domain yang sama dan diakses melalui internet menggunakan web browser. *Website* berfungsi sebagai platform penyampaian informasi, layanan, atau interaksi dengan pengguna (Ridho Pamungkas, 2018). Sebuah *website* biasanya berisi informasi, layanan, dan juga penyedia *platform* interaksi antara pemilik dan pengguna. Pada era digital fungsi website saat ini sangat beragam, mulai dari sarana komunikasi pribadi, portal berita yang menyediakan informasi terkini, hingga *platform e-commerce* yang memungkinkan pengguna dalam melakukan transaksi jual beli secara daring (*online*).

Dalam pengembangannya, website dibangun dengan menggunakan bahasa-bahasa pemrograman web standar, seperti HTML (*HyperText Markup Language*) untuk struktur dasar halaman, CSS (*Cascading Style Sheets*) untuk tata letak dan desain visual, serta *JavaScript* untuk membuat elemen-elemen interaktif (Sholikhan, 2022). *Website* yang modern juga sering menggunakan teknologi backend salah satunya adalah PHP untuk memproses permintaan server, menyimpan data pengguna, dan mengelola konten secara dinamis. Penggunaan berbagai teknologi menjadikan *website* sebagai *platform* yang multifungsi, sehingga dapat diakses kapan saja dan dari mana saja oleh pengguna yang terhubung ke internet.

#### **2.2.4 Artificial Intelligence**

Kecerdasan Buatan atau yang dikenal sebagai *Artificial Intelligence* (AI) adalah salah satu cabang dari ilmu komputer yang mempelajari bagaimana menciptakan mesin atau sistem yang mampu meniru kemampuan manusia (Sobron & Lubis, 2021). Dengan menggunakan AI memungkinkan untuk melakukan tugas-tugas yang biasanya memerlukan kecerdasan manusia, seperti pengenalan suara, pengambilan keputusan, pembelajaran, dan pemecahan masalah dalam waktu yang lebih singkat. Seiring dengan kemajuan teknologi, AI telah menjadi bagian integral dari berbagai bidang, seperti kesehatan, keuangan, transportasi, dan manufaktur. Dengan kemampuannya yang terus berkembang, AI dipandang sebagai teknologi masa depan yang memiliki potensi besar untuk meningkatkan efisiensi di berbagai sektor kehidupan manusia.

#### **2.2.5 Framework CodeIgniter**

*CodeIgniter* termasuk kedalam salah satu contoh framework PHP yang *open-source* dengan konsep MVC (*Model View Controller*) yang sering digunakan untuk membangun sebuah sistem (Ikhsan et al., 2023). Dengan menggunakan MVC, pengembang dapat dengan mudah mengelola berbagai komponen aplikasi secara terpisah, sehingga memudahkan proses pengembangan, pemeliharaan, dan perbaikan kode. *CodeIgniter* ini memiliki banyak keunggulan seperti fitur-fitur yang sangat mendukung dalam pengembangan maupun rancang bangun sebuah sistem teknologi informasi. Selain itu, fitur-fitur keamanan yang disediakan, seperti enkripsi data, perlindungan terhadap serangan *SQL injection*, dan validasi input, menjadikan *CodeIgniter* sebagai salah satu pilihan utama dalam pengembangan aplikasi web yang aman (CodeIgniter, 2024).

#### **2.2.6 Basis Data**

Basis data atau *database* adalah kumpulan data atau objek yang disusun dan disimpan secara sistematis sehingga dapat diakses, diubah, dan dikelola dengan mudah (Andaru, 2018). *Database* berfungsi sebagai penyimpanan utama untuk data dalam berbagai aplikasi, salah satunya aplikasi berbasis web. *Database* memungkinkan penyimpanan data dalam jumlah besar dan memberikan dukungan untuk melakukan berbagai operasi, seperti pencarian, pembaruan, dan penghapusan data.

MySQL merupakan salah satu sistem manajemen basis data yang banyak digunakan karena sifatnya yang *open-source* serta kemampuannya untuk berintegrasi dengan berbagai aplikasi berbasis web. MySQL mendukung penyimpanan dan pengelolaan data dalam jumlah besar menggunakan tabel-tabel yang terstruktur, yang memungkinkan pengguna untuk mengelola informasi secara efisien. Basis data ini memanfaatkan SQL (*Structured Query Language*) sebagai bahasa standar untuk melakukan operasi-operasi seperti penyimpanan, pengambilan, dan manipulasi data. Keunggulan MySQL antara lain adalah kemampuannya dalam menangani banyak pengguna (*multi-user*) secara bersamaan serta kecepatannya dalam menerima dan mengirim data, menjadikannya pilihan populer untuk aplikasi web yang dikembangkan dengan bahasa pemrograman seperti PHP. Dengan menggunakan MySQL, pengembang dapat menciptakan sistem penyimpanan data yang stabil, aman, dan mudah diakses. Fleksibilitas yang ditawarkan oleh MySQL memungkinkan pengelolaan berbagai jenis data, baik dalam skala kecil maupun besar, sehingga cocok untuk digunakan dalam berbagai aplikasi web modern (Prasetya et al., 2023).

### **2.2.7 PHP (*Hypertext Preprocessor*)**

PHP adalah bahasa pemrograman yang digunakan terutama untuk pengembangan aplikasi web berbasis server. PHP memungkinkan pembuatan halaman web dinamis, di mana konten dapat diubah secara *real-time* berdasarkan input pengguna atau data dari *database*. PHP dapat digunakan bersama HTML, CSS, dan *JavaScript* untuk membuat aplikasi web interaktif. PHP mendukung berbagai sistem basis data, seperti MySQL dan PostgreSQL, sehingga memungkinkan pengembangan aplikasi berbasis data dengan mudah.

PHP memiliki banyak keunggulan, salah satu keunggulannya adalah kemudahan penggunaan, karena sintaksnya yang sederhana dan dukungan luas dari komunitas pengembang. Selain itu, PHP juga dapat diintegrasikan dengan berbagai *framework*, seperti *CodeIgniter* dan *Laravel*, yang mempercepat proses pengembangan dengan menyediakan fitur-fitur bawaan yang siap digunakan. PHP juga mendukung manipulasi data dan penyusunan logika bisnis dalam satu lingkungan pengembangan, menjadikannya sebagai salah satu bahasa

pemrograman yang paling populer untuk membangun aplikasi berbasis web (PHP.net, 2024).

### **2.2.8 Xampp**

XAMPP adalah perangkat lunak *open-source* yang digunakan sebagai server lokal untuk pengembangan aplikasi berbasis web. XAMPP memiliki tiga komponen penting yang sering digunakan untuk aplikasi berbasis web, yaitu web server Apache, PHP, dan MySQL (Andani et al., 2021). Dengan menggunakan XAMPP, pengembang dapat mengembangkan, menguji, dan menjalankan aplikasi web secara lokal sebelum dipublikasikan ke server *online*. Selain itu, XAMPP juga menyediakan antarmuka pengguna yang sederhana untuk mengelola server, basis data, dan aplikasi secara langsung. Oleh karena itu, XAMPP menjadi pilihan populer di kalangan pengembang untuk mempercepat proses pengembangan dan pengujian aplikasi web.