

## **BAB II**

### **TEORI DAN TINJAUAN PUSTAKA**

Pada bab 2 akan dibahas tentang dasar teori dan tinjauan pustaka yang digunakan dalam pembuatan Proyek Akhir ini.

#### **2.1 Penelitian Terdahulu**

Untuk mendapatkan hasil penelitian yang optimal, dilakukan kajian dari penelitian-penelitian terdahulu, sehingga dapat dijadikan referensi dalam penelitian dengan tujuan agar diperoleh perbandingan kelebihan dan kekurangan pada masing-masing perancangan.

Pada penelitian terdahulu menurut Ramadhan dan Saputra (2021) dalam jurnal yang berjudul Strategi Pengembangan Website untuk UMKM di Kota Batu memiliki hasil penelitian yang mengungkapkan bahwa website untuk UMKM di Kota Batu meningkatkan aksesibilitas dan promosi produk lokal. Implementasi strategi digital terbukti meningkatkan penjualan dan jangkauan pasar. Disarankan adanya kolaborasi dengan pemerintah untuk peningkatan infrastruktur teknologi.

Menurut Wijaya dan Hartono (2020) dalam jurnal yang berjudul Peran Pemerintah dalam Mendukung UMKM melalui Website Resmi, memiliki hasil penelitian yang menunjukkan bahwa dukungan pemerintah melalui website resmi sangat penting dalam membantu UMKM berkembang. Dengan menyediakan platform informasi dan layanan online, UMKM dapat lebih mudah mendapatkan informasi tentang perizinan dan peluang investasi.

Menurut Haryanto dan Susilo (2019) dalam jurnal yang berjudul Efektivitas Website dalam Peningkatan Kualitas Layanan UMKM, memiliki hasil penelitian yang menyoroti bahwa penggunaan website dapat meningkatkan kualitas layanan UMKM dengan memberikan informasi yang akurat dan real-time kepada

pelanggan. Hasil menunjukkan peningkatan kepuasan pelanggan dan efisiensi operasional.

Menurut Yusuf dan Anwar (2018) dalam jurnal yang berjudul Pengaruh Digitalisasi terhadap Pertumbuhan UMKM, memiliki hasil penelitian yang menemukan bahwa digitalisasi, termasuk pengembangan website, berdampak positif terhadap pertumbuhan UMKM dengan memperluas akses pasar dan memperbaiki manajemen operasional.

Menurut Yusuf dan Anwar (2018) dalam jurnal yang berjudul Pengaruh Digitalisasi terhadap Pertumbuhan UMKM, memiliki hasil penelitian yang menemukan bahwa digitalisasi, termasuk pengembangan website, berdampak positif terhadap pertumbuhan UMKM dengan memperluas akses pasar dan memperbaiki manajemen operasional.

Dari penelitian terdahulu yang telah penulis paparkan, berikut hasil penelitian terdahulu dalam bentuk table:

**Tabel 2.1 Tabel Penelitian Terdahulu**

No.	Nama Penulis dan Tahun	Judul Jurnal	Hasil Penelitian
1.	(Ramadhan & Saputra, 2021)	Strategi Pengembangan Website untuk UMKM di Kota Batu	Penelitian ini mengungkapkan bahwa pengembangan website untuk UMKM di Kota Batu meningkatkan aksesibilitas dan promosi produk lokal. Implementasi strategi digital terbukti meningkatkan penjualan dan jangkauan pasar. Disarankan adanya kolaborasi dengan pemerintah untuk peningkatan infrastruktur teknologi.
2.	(Wijaya & Hartono, 2020)	Peran Pemerintah dalam Mendukung UMKM melalui Website Resmi	Hasil penelitian menunjukkan bahwa dukungan pemerintah melalui website resmi sangat penting dalam membantu UMKM berkembang. Dengan menyediakan platform informasi dan layanan online, UMKM dapat lebih mudah mendapatkan informasi tentang perizinan dan peluang investasi.

**Tabel 2.1 Tabel Penelitian Terdahulu (Lanjutan)**

No.	Nama Penulis dan Tahun	Judul Jurnal	Hasil Penelitian
	(Haryanto & Susilo, 2019)	Efektivitas Website dalam Peningkatan Kualitas Layanan UMKM	Penelitian ini menyoroti bahwa penggunaan website dapat meningkatkan kualitas layanan UMKM dengan memberikan informasi yang akurat dan real-time kepada pelanggan. Hasil menunjukkan peningkatan kepuasan pelanggan dan efisiensi operasional.
	(Yusuf & Anwar, 2018)	Pengaruh Digitalisasi terhadap Pertumbuhan UMKM	Studi ini menemukan bahwa digitalisasi, termasuk pengembangan website, berdampak positif terhadap pertumbuhan UMKM dengan memperluas akses pasar dan memperbaiki manajemen operasional.
	(Kurniawan & Santoso, 2020)	Implementasi Sistem B2B untuk UMKM di Kota Surabaya	Penelitian ini membahas implementasi sistem B2B untuk UMKM di Surabaya, menunjukkan bahwa penggunaan platform B2B dapat memperluas jaringan bisnis dan meningkatkan efisiensi dalam proses pemasaran dan distribusi. Ditemukan bahwa kolaborasi antara UMKM dan perusahaan besar melalui platform B2B meningkatkan pertumbuhan bisnis.

## **2.2 Landasan Teori**

### **2.2.1 Strategi Pengembangan Website untuk UMKM**

Pengembangan website untuk UMKM merupakan langkah strategis untuk meningkatkan aksesibilitas dan promosi produk lokal. Penelitian oleh Ramadhan dan Saputra (2021) menunjukkan bahwa implementasi strategi digital mampu meningkatkan penjualan dan jangkauan pasar UMKM di Kota Batu. Kolaborasi dengan pemerintah dalam peningkatan infrastruktur teknologi sangat disarankan untuk mendukung optimalisasi strategi ini.

### **2.2.2 Peran Pemerintah dalam Mendukung UMKM melalui Website Resmi**

Dukungan pemerintah sangat penting dalam membantu perkembangan UMKM melalui website resmi. Wijaya dan Hartono (2020) menemukan bahwa dengan menyediakan platform informasi dan layanan online, UMKM dapat dengan mudah mendapatkan informasi tentang perizinan dan peluang investasi. Hal ini menunjukkan bahwa peran aktif pemerintah dalam mendukung digitalisasi UMKM sangat krusial.

### **2.2.3 Efektivitas Website dalam Peningkatan Kualitas Layanan UMKM**

Penggunaan website oleh UMKM dapat meningkatkan kualitas layanan dengan memberikan informasi yang akurat dan real-time kepada pelanggan. Haryanto dan Susilo (2019) menyoroti bahwa hasil penelitian mereka menunjukkan peningkatan kepuasan pelanggan dan efisiensi operasional UMKM yang menggunakan website. Ini menunjukkan bahwa digitalisasi layanan dapat meningkatkan daya saing UMKM.

### **2.2.4 Pengaruh Digitalisasi terhadap Pertumbuhan UMKM**

Digitalisasi, termasuk pengembangan website, berdampak positif terhadap pertumbuhan UMKM. Studi oleh Yusuf dan Anwar (2018) menemukan bahwa digitalisasi dapat memperluas akses pasar dan memperbaiki manajemen operasional. Ini menunjukkan bahwa penerapan teknologi digital merupakan faktor penting dalam mendukung pertumbuhan dan keberlanjutan UMKM.

### **2.2.5 Implementasi Sistem B2B untuk UMKM**

Implementasi sistem B2B (Business to Business) untuk UMKM dapat memperluas jaringan bisnis dan meningkatkan efisiensi dalam proses pemasaran dan distribusi. Kurniawan dan Santoso (2020) membahas bahwa penggunaan platform B2B membantu kolaborasi antara UMKM dan perusahaan besar, yang pada akhirnya meningkatkan pertumbuhan bisnis. Ini menunjukkan bahwa integrasi dengan sistem B2B dapat menjadi strategi efektif dalam meningkatkan performa UMKM.

## **2.3 Bahasa Pemrograman**

### **2.3.1 Kotlin**

Implementasi sistem B2B (Business to Business) untuk UMKM dapat memperluas jaringan bisnis dan meningkatkan efisiensi dalam proses pemasaran dan distribusi. Kurniawan dan Santoso (2020) membahas bahwa penggunaan platform B2B membantu kolaborasi antara UMKM dan perusahaan besar, yang pada akhirnya meningkatkan pertumbuhan bisnis. Ini menunjukkan bahwa integrasi

dengan sistem B2B dapat menjadi strategi efektif dalam meningkatkan performa UMKM.

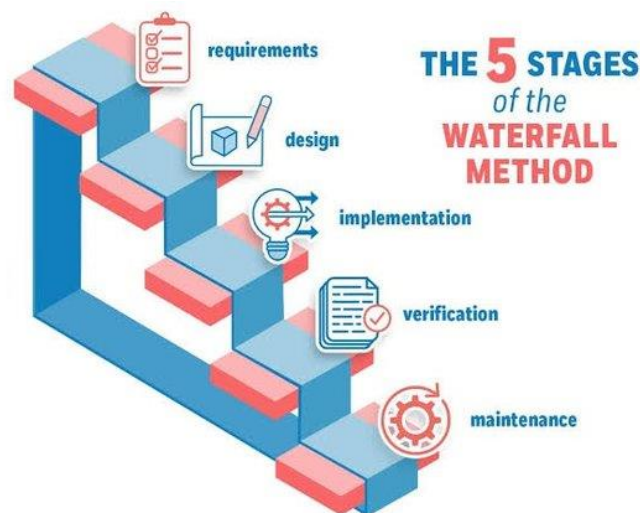
### 2.3.2 Java Script

Menurut (David Flanagan, 2018) JavaScript merupakan bahasa pemrograman yang paling umum digunakan untuk pengembangan aplikasi web. Bahasa ini bersifat client-side, yang berarti kode JavaScript dieksekusi oleh web browser di sisi klien. JavaScript digunakan untuk membuat halaman web menjadi interaktif dengan menyediakan fitur seperti validasi formulir, manipulasi DOM, animasi, dan komunikasi dengan server melalui AJAX. Dengan perkembangan modern, JavaScript juga dapat digunakan di server-side melalui platform seperti Node.js.

## 2.4 Perancangan

### 2.4.1 Waterfall

Menurut (Widianti & Darussalam, 2023), metode perancangan waterfall merupakan model pengembangan sistem informasi secara terorganisir dan berurutan. Keuntungan menggunakan metode ini adalah untuk meminimalisir kesalahan karena prosesnya dilakukan secara linear.



Gambar 2.1 Metode Perancangan Waterfall

Terdapat 5 tahapan dalam metode waterfall sesuai dengan Gambar 2.1:

a. Analisis Kebutuhan

Analisis kebutuhan merupakan tahap penting dalam pengembangan sistem, di mana hasil dari konsultasi pengguna digunakan untuk menentukan layanan sistem, batasan, dan tujuan. Informasi yang diperoleh kemudian dijelaskan secara rinci dan diformulasikan menjadi spesifikasi sistem.

b. Perancangan aplikasi

Tahap perancangan sistem melibatkan alokasi kebutuhan sistem baik untuk perangkat keras maupun perangkat lunak. Proses perancangan perangkat lunak melibatkan identifikasi dan karakterisasi abstraksi sistem fundamental serta hubungan di antara mereka. Perancangan perangkat lunak pada sistem baru dilakukan dengan merancang keluaran sesuai kebutuhan pengguna. Desain antarmuka dibuat sesuai kebutuhan dan dirancang sebaik mungkin agar pengguna dapat menggunakannya dengan mudah.

c. Implementasi

Pada tahap ini desain perangkat lunak diimplementasikan sebagai kumpulan program atau komponen program oleh pengembang.

d. Integrasi dan Pengujian

Setelah melakukan implementasi, akan dilakukan pengujian untuk menentukan apakah unit program individu atau program memenuhi persyaratan perangkat lunak, mereka digabungkan dan dinilai sebagai sistem keseluruhan. Perangkat lunak dapat diserahkan kepada pelanggan setelah pengujian. Penerapan sistem dilakukan dengan mempublikasikan aplikasi berbasis web melalui pendaftaran pada perusahaan penyedia layanan internet atau penyedia *hosting*.

e. Pemeliharaan

Tahapan ini adalah tahapan dimana kesalahan yang terlewatkan pada fase sebelumnya diperbaiki, unit sistem diimplementasikan dengan lebih baik, dan layanan sistem ditingkatkan sebagai respon terhadap perubahan

kebutuhan. Pemeliharaan dilakukan secara berkesinambungan berdasarkan analisis berjalan karena kebutuhan masyarakat akan terus berubah dan berkembang secara dinamis.




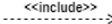
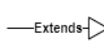

### 2.4.2 Relasi Tabel

Menurut PrepBytes (2023) relasi tabel dalam basis data adalah hubungan logis yang terbentuk antara tabel-tabel yang ada untuk menghubungkan data sehingga informasi dapat disusun lebih terstruktur dan efisien. Tiga jenis utama relasi tabel adalah satu-ke-satu(*one to one*), satu-ke-banyak(*one to many*), dan banyak-ke-banyak(*many to many*). Relasi ini biasanya ditentukan oleh kunci primer (*primary key*) dan kunci tamu (*foreign key*) yang memastikan integritas referensial, sehingga hubungan antar tabel tetap konsisten.

### 2.4.3 Use case Diagram

Menurut (Ruli, 2019), use case diagram merupakan pemodelan interaksi sistem untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi tersebut dan siapa saja yang berhak melakukan fungsi tersebut. Simbol-simbol khusus yang digunakan dalam pembuatan *use case diagram* dijelaskan pada gambar tabel 2.2.

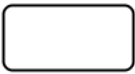






**Tabel 2.2 Detail Simbol Use case Diagram**

No	Simbol	Nama Simbol	Keterangan
1	 Actor	Actor	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case
2		Generalization	Hubungan dimana objek anak ( <i>descendent</i> ) berbagai perilkudan struktur data dari objek yang ada diatasnya objek induk ( <i>ancestor</i> )
3		Association	Apa yang menghubungkan antara objek satu dengan objek yang lainnya
4		Include	Menspesifikasikan bahwa use case sumber secara eksplisit
5		Extend	Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada suatu titik yang diberikan
6		Use case	Deskripsi dari urutan aksi aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor

## 2.4.4 Activity Diagram

Menurut (Kurniawan & Syarifuddin, 2020), activity diagram menggambarkan berbagai aliran aktivitas dalam sistem yang sedang di rancang, bagaimana masing-masing aliran berawal, percabangan yang mungkin terjadi dan bagaimana mereka berakhir. Simbol-simbol khusus yang digunakan dalam pembuatan activity diagram dijelaskan pada gambar tabel 2.3.

**Tabel 2.3 Detail Simbol Activity Diagram**

No	Simbol	Nama Simbol	Keterangan
1		<i>Activity</i>	Memperlihatkan bagaimana masing masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Actions</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali
4		<i>Initial Final Node</i>	Bagaimana objek dibentuk dan diakhiri
5		<i>Decisions</i>	Digunakan untuk menggambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu
6		<i>Line Connections</i>	Digunakan untuk menghubungkan satu simbol dengan simbol yang lainnya
7		<i>Swimlane</i>	Pembagian <i>activity diagram</i> untuk membedakan aktor dengan sistem

## 2.4.5 Sequence Diagram

Menurut (Kurniawan & Syarifuddin, 2020), sequence diagram adalah diagram yang digunakan untuk menggambarkan interaksi antar objek di dalam dan di sekitar sistem yang berupa pesan yang digambarkan terhadap waktu. Simbol-simbol khusus yang digunakan dalam pembuatan sequence diagram dijelaskan pada gambar tabel 2.4.



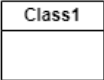


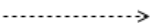


**Tabel 2.4 Detail Simbol Sequence Diagram**

No	Simbol	Nama Simbol	Keterangan
1	 Actor	Actor	Menggambarkan seseorang atau sesuatu yang berinteraksi dengan sistem
2		Boundary	Menggambarkan interaksi antara satu atau lebih aktor dengan sistem, memodelkan bagian dari sistem yang bergantung pada pihak lain disekitarnya dan merupakan pembatas sistem dengan dunia luar
3		Control	Menggambarkan "Perilaku mengatur", mengkoordinasikan perilaku sistem dan dinamika dari suatu sistem, menangani tugas utama dan mengontrol alur kerja suatu sistem
4		Entity	Menggambarkan informasi yang harus disimpan oleh sistem (struktur data dari sebuah sistem)
5		Object message	Menggambarkan pesan/hubungan antar objek yang menunjukkan urutan kejadian yang terjadi
6		Message to self	Menggambarkan pesan/hubungan objek itu sendiri, yang menunjukkan urutan kejadian yang terjadi
7		Return message	Menggambarkan pesan/hubungan antar objek yang menunjukkan urutan kejadian yang terjadi

### 2.4.6 Class Diagram

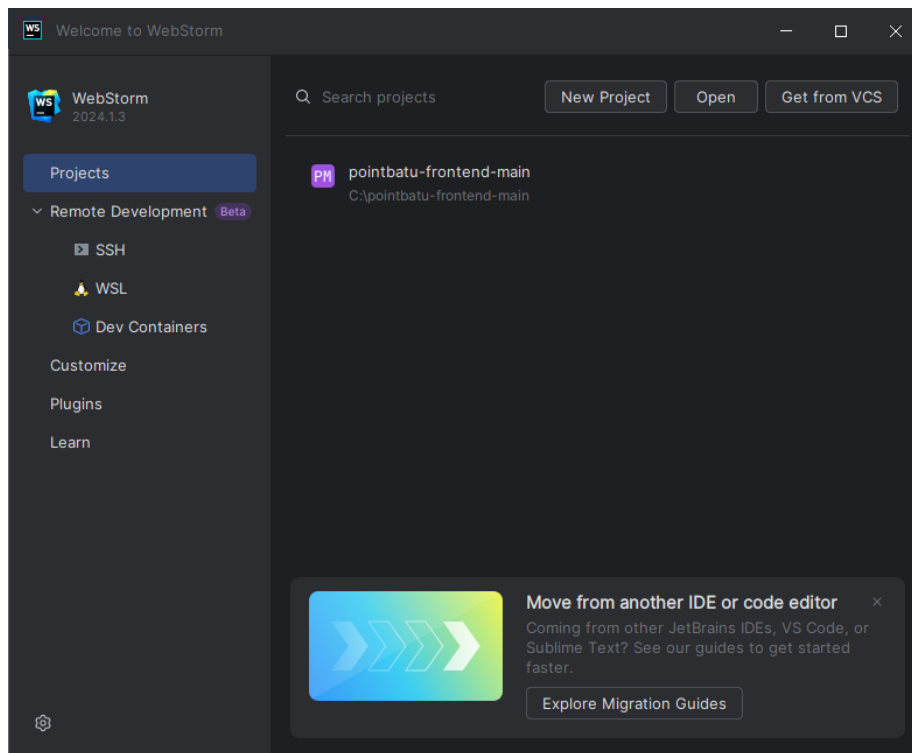
Menurut (Putra & Andriani, 2019), *class diagram* merupakan gambaran struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. *Class diagram* terdiri dari atribut dan operasi dengan tujuan pembuat pembuat program dapat membuat hubungan antara dokumentasi perancangan dan perangkat lunak sesuai. Simbol-simbol khusus yang digunakan dalam pembuatan *class diagram* dijelaskan pada gambar tabel 2.5.

**Tabel 2.5 Detail Simbol Class Diagram**

No	Simbol	Nama Simbol	Keterangan
1		<i>Class</i>	Himpunan dari objek objek yang berbagi atribut serta operasi yang sama
2		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagai perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> )
3		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek yang lainnya
4		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
5		<i>Realization</i>	Operasi yang benar benar dilakukan oleh suatu objek
6		<i>Collaboration</i>	Interaksi aturan aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen elemennya

### 2.4.7 WebStorm

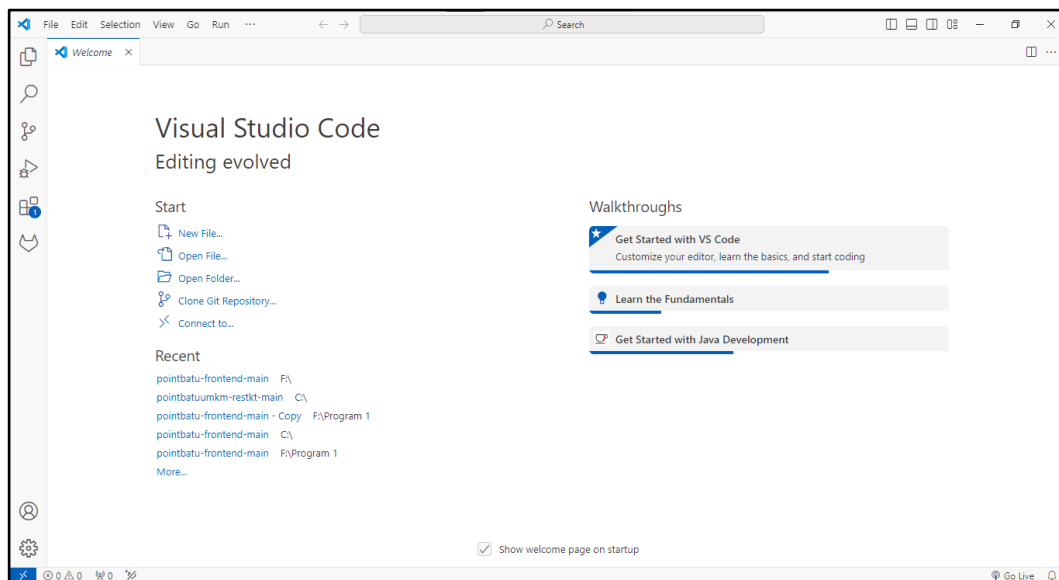
Menurut pengembang kode editor yaitu JetBrains (2024), Webstorm adalah Integrated Development Environment (IDE) yang dirancang khusus untuk pengembangan web oleh JetBrains. IDE ini mendukung banyak teknologi web seperti JavaScript, TypeScript, HTML, CSS, dan berbagai framework seperti React, Angular, Vue.js, dan Node.js. Tampilan WebStorm dapat dilihat pada Gambar 2.2. WebStorm menyediakan berbagai fitur canggih seperti intelligent code completion, code navigation, refactoring, debugger, version control, dan testing. WebStorm membantu meningkatkan produktivitas pengembang web dengan menyediakan alat yang lengkap dan efisien untuk pengembangan dan debugging aplikasi web.



**Gambar 2.2 Tampilan WebStrom IDE**

#### **2.4.8 Visual Studio Code**

Visual Studio Code (VS Code) adalah editor kode sumber terbuka yang dikembangkan oleh Microsoft. Editor ini mendukung berbagai bahasa pemrograman dan dilengkapi dengan fitur-fitur canggih seperti debugging, kontrol versi Git, penyorotan sintaks, IntelliSense (penyelesaian kode cerdas), dan ekstensi yang dapat diunduh untuk menambahkan fungsionalitas baru. VS Code dirancang untuk bekerja pada berbagai sistem operasi termasuk Windows, macOS, dan Linux, sehingga menjadi pilihan populer di kalangan pengembang perangkat lunak, dapat dilihat tampilan halaman visual studio code seperti pada Gambar 2.3.



**Gambar 2.3 Tampilan Visual Studio Code**

## **2.5 Database**

### **2.5.1 MySQL**

Menurut (Noviana, 2022), MySQL adalah sistem manajemen basis data (DBMS) sumber terbuka yang mendukung penggunaan *multiuser*, *multithreaded*, populer, dan gratis. Berdasarkan teori di atas, dapat disimpulkan bahwa SQL adalah bahasa permintaan basis data tertentu di mana subbahasa tersebut memungkinkan pembuatan dan manipulasi data dalam basis data. SQL digunakan untuk menjalankan tugas-tugas seperti melakukan pembaruan terhadap basis data, yang merujuk pada konsep Sistem Manajemen Basis Data Relasional (RDBMS).

### **2.5.2 XAMPP**

Menurut (Noviana, 2022), XAMPP adalah singkatan dari (X-platform, Apache, MySQL, PHP, Perl). Perangkat lunak berbasis web server yang bersifat open source (bebas), serta mendukung di berbagai sistem operasi, baik Windows, Linux, atau Mac OS. XAMPP digunakan sebagai stand alone server (berdiri sendiri) atau biasa disebut dengan localhost.

## **2.6 Pengujian**

### **2.6.1 Black Box**

Menurut (Yahya Dwi & Muna Wardah, 2021), Merupakan pengujian kualitas perangkat lunak yang berfokus pada fungsionalitas perangkat lunak dengan tujuan untuk menemukan fungsi yang tidak benar, kesalahan antarmuka, kesalahan pada struktur data, kesalahan performansi, kesalahan inisialisasi dan terminasi. Pengujian *Black Box* bertumpu pada memastikan setiap proses berfungsi sesuai dengan kebutuhan yang diharapkan. Dengan menggunakan himpunan kondisi masukan, penguji menjalankan pengujian pada fungsi spesifik dari sistem. Tujuannya adalah untuk menemukan kesalahan dalam program dan memperbaikinya sehingga sistem dapat dianggap layak untuk digunakan.