



BAB 2 LANDASAN TEORI

2.1 Penelitian Terdahulu

Penelitian terdahulu ini adalah sebagai bahan acuan berupa teori-teori atau temuan-temuan melalui berbagai penelitian terdahulu oleh beberapa peneliti yang pernah penulis baca dan yang akan dijabarkan sesuai dengan permasalahan yang sedang dibahas dalam penelitian ini.

Rujukan penelitian pertama yang menjadi referensi yaitu jurnal artikel yang berjudul **“Pengembangan “Sistem Informasi Pembelajaran Sertifikasi Pengadaan Barang dan Jasa Pemerintah” (SIMPENBAJA)”**. Penelitian yang terlibat langsung dengan Lembaga Kebijakan Pengadaan Barang dan Jasa Pemerintah (LKPP) tersebut, menjumpai beberapa kendala yang dirasa dapat mempengaruhi kinerja bagi Aparatur Sipil Negara (ASN) yang terlibat dalam kegiatan proses bisnis yang ada terutama dalam perihal memperoleh materi yang digunakan untuk belajar dalam memperdalam materi guna mengikuti ujian sertifikasi. Dengan adanya Sistem Informasi Pembelajaran Pengadaan Barang dan Jasa Pemerintah (SIMPENBAJA) ini dibuat, yang nantinya aplikasi ini dapat dimanfaatkan guna menambah pemahaman kepada Peserta ujian sertifikasi/ASN dalam mengikuti ujian serta dapat meningkatkan kualitas lulusan bersertifikasi yang nantinya ikut andil dalam proses kegiatan pengadaan barang dan jasa pemerintah. (Pical & Veritawati, 2022)

Rujukan penelitian kedua yang menjadi referensi yaitu jurnal artikel yang berjudul **“Pengembangan Sistem Informasi Eksekutif Unit Layanan Pengadaan Pada Kementerian Ketenagakerjaan”**. Tim Unit Layanan Pengadaan (ULP) Kementerian Ketenagakerjaan Republik Indonesia yang telah melaksanakan pengadaan barang dan jasa dirasa masih belum terlaksana secara maksimal. Selain resiko *kolusi*, *korupsi*, dan

nepotisme (KKN), proses sinkronisasi data masih dilakukan secara manual dikarenakan sumber data dari kedua sistem tersebut yang belum terintegrasi satu sama lain. Tujuan dari penelitian ini adalah untuk menciptakan sistem informasi eksekutif di Kementerian Ketenagakerjaan untuk menyelesaikan masalah saat ini dan mendukung transformasi digital untuk memastikan pengelolaan anggaran yang baik. *Pre-project*, *lifecycle project*, dan *post-project* adalah tiga tahap yang membentuk metode pengembangan sistem dinamis (DSDM). Hasilnya, sistem dapat membantu pencapaian tujuan ULP dengan memantau, menilai, dan menganalisis data untuk PPK dan pihak lain. (Rizki & Ridwan, 2023)

Rujukan penelitian ketiga yang menjadi referensi yaitu jurnal artikel yang berjudul “**Sistem Informasi Arsip Data Kepegawaian Menggunakan Metode Agile Development Dinas Kesejahteraan Sosial**”. Sistem informasi arsip data kepegawaian pada dinas kesejahteraan sosial masih melakukan pengarsipan data kepegawaian secara manual, hal ini dapat mengakibatkan kerumitan dalam pencarian data pegawai dan pembaruan informasi pegawai serta tidak ada sistem pengamanan yang memadai, sehingga meningkatkan risiko kebocoran data atau akses yang tidak sah. Maka dari itulah dibuat sebuah sistem informasi arsip data kepegawaian dengan menggunakan metode agile development yaitu pendekatan pengembangan SDLC (*System Development Life Cycle*) dan melakukan pengujian sistem *black box testing*. Dimana aplikasi/sistem ini dapat lebih membantu seperti pencarian data kepegawaian yang tergolong cepat dan pembaruan data lebih di permudah dari sebelumnya serta memberikan fasilitas pengamanan data yang lebih aman dengan fitur yang dibuat pada aplikasi. (Letsoin & Prayitno, 2023)

Rujukan penelitian keempat yang menjadi referensi yaitu jurnal artikel yang berjudul “**Akuntabilitas Pengadaan Barang atau Jasa E-Katalog di Kelurahan Wonokromo Kota Surabaya**”. Berdasarkan Peraturan Presiden Nomor 12 Tahun



2021 tentang Pengadaan Barang/Jasa Pemerintah, pengadaan barang dan jasa secara elektronik (EGP) atau *E-Procurement* didefinisikan sebagai pengadaan barang dan jasa yang lebih akuntabel. Teori BAPPENAS memiliki tiga indikator, yaitu adanya kesesuaian antara pelaksanaan dan prosedur pelaksanaan yang didukung oleh peraturan perundang-undangan, adanya sanksi untuk kesalahan atau kelalaian yang dipersamakan dengan adanya sistem penilaian penyerapan di tempat penyelenggara, dan adanya output dan hasil yang dapat diukur dan didokumentasikan. (Putri, Sunarya, & Fatah, 2023)

Rujukan penelitian kelima yang menjadi referensi yaitu jurnal artikel yang berjudul **“Tinjauan Pelaksanaan Barang dan Jasa”**. Dalam proses pengadaan barang dan jasa yang dilakukan oleh Divisi *Supply Chain*, ada beberapa masalah yang memperlambat proses pengadaan. Beberapa masalah ini termasuk keterlambatan pengiriman material dan barang, yang dikenal sebagai diameter alat, dan penggunaan harga perkiraan sendiri (HPS) berdasarkan data sebelumnya. Upaya untuk menangani masalah vendor termasuk: perusahaan tidak akan memberikan sanksi jika vendor mengajukan keluhan terkait keterlambatan barang yang masuk akal, tetapi sanksi yang berlaku akan diterapkan jika keterlambatan barang tidak masuk akal. Jika vendor tidak dapat menyelesaikan pesanan barang yang dipesan, vendor tersebut akan dipilih kembali atau dibuat pengadaan tambahan untuk vendor tersebut. Untuk mengatasi masalah Harga Perkiraan Sendiri (HPS), pengguna dapat melakukan survei harga di lapangan atau melalui media elektronik. (Jubaedah & Suprastiyo, 2022)

Rujukan penelitian keenam yang menjadi referensi yaitu jurnal artikel yang berjudul **“Perancangan Sistem Informasi Laboratorium Program Studi Teknologi Rekayasa Komputer Jaringan dengan Metode Agile Development Menggunakan Cloud Computing Berbasis Web”**. *Digitalisasi* adalah masa ketika teknologi digital sangat penting untuk berbagai aspek kehidupan. Adanya layanan *cloud* adalah salah satu keuntungan *digitalisasi*. Informasi, data



siswa, pengolahan nilai, dan berbagai laporan akademik adalah kemudahan yang diperoleh di bidang akademik. Penyimpanan data dilakukan pada titik tertentu melalui penyimpanan *hard copy*, yang memiliki kendala seperti pemulihan data, pencarian data secara manual yang memakan waktu, dan pemborosan kertas. Perancangan Sistem Informasi Laboratorium Program Studi Teknologi Rekayasa Komputer Jaringan menggunakan metode *Agile* dan *Cloud Computing* berbasis web dapat meningkatkan efisiensi dan efektivitas pengelolaan laboratorium dan memberikan kemudahan akses bagi pengguna, terutama untuk membantu mahasiswa dan dosen mengelola laporan laboratorium dan meningkatkan efisiensi pencarian data. (Sukma, Amelia, Arnita, & Zoni, 2023)



Tabel 2. 1 Daftar Penelitian-penelitian Terdahulu yang Terkait

No.	Nama Peneliti dan Judul Penelitian	Metode	Persamaan	Perbedaan	Hasil Output
1.	Yohanes Paulus Pical & Ionia Veritawati (2022) "Pengembangan "Sistem Informasi Pembelajaran Sertifikasi Pengadaan Barang dan Jasa Pemerintah" (SIMPENBAJA) "	Survey (kendala dan kebutuhan), UML dan ERD (desain model), dan Blackbox testing (testing)	Melakukan penelitian berupa pengembangan pada sistem informasi pengadaan barang	Tempat studi kasus yang dijadikan penelitian	Sistem Informasi Pembelajaran Sertifikasi Barang dan Jasa Pemerintah (SimpénBaja) adalah sistem pembelajaran interaktif yang memungkinkan siswa berinteraksi dengan guru. SimpenBaja memiliki materi dan Latihan soal yang membantu Pegawai Negeri Sipil mengetahui lebih banyak tentang Pengadaan Barang dan Jasa Pemerintah (PBJ) dan mengevaluasi kemampuan mereka.
2.	Muhammad Yazir Rizki & Anggraeni	<i>Dynamic System</i>	Melakukan penelitian	<ul style="list-style-type: none"> • Metode yang 	Dalam proses pengadaan barang atau jasa untuk

No.	Nama Peneliti dan Judul Penelitian	Metode	Persamaan	Perbedaan	Hasil Output
	Ridwan (2023) "Pengembangan Sistem Informasi Eksekutif Layanan Pengadaan Pada Kementerian Ketenagakerjaan".	<i>Development Method (DSDM)</i>	berupa pengembangan pada sistem informasi pengadaan barang.	<p>digunakan dalam (pengembangan) penelitian tersebut.</p> <ul style="list-style-type: none"> • Tempat studi kasus yang dijadikan objek penelitian. 	kepentingan pemerintah, situs Sistem Informasi Rencana Umum Pengadaan Lembaga Kebijakan Pengadaan Barang atau Jasa Republik Indonesia (SIRUP LKPP) dan LPSE Naker berdiri sendiri dan tidak terhubung satu sama lain. Dengan sistem ini, proses dokumentasi berkas pengadaan RUP (Rencana Umum Pengadaan) dan RPP (Rencana Pelaksanaan Pengadaan) menjadi lebih mudah. Ini meningkatkan efektivitas, efisiensi, transparansi, dan akuntabilitas penggunaan anggaran Negara.

No.	Nama Peneliti dan Judul Penelitian	Metode	Persamaan	Perbedaan	Hasil Output
					Selain itu, pelaporan secara berkala memudahkan proses audit, sehingga tidak ada lagi komentar audit terkait dokumentasi pengadaan.
3.	Maria Claudia Letsoin & Gunawan Prayitno (2023) "Sistem Informasi Arsip Data Kepegawaian Menggunakan Metode Agile Development Dinas Kesejahteraan Sosial"	Metode <i>Agile Development</i>	Metode yang dilakukan pada penelitian tersebut sama, yaitu <i>agile development methodology</i> .	Tempat studi kasus yang dijadikan penelitian	Aplikasi sistem informasi arsip data kepegawaian berbasis android ini memberikan kemudahan dalam pencarian data lama maupun data baru serta memberikan kemudahan untuk melakukan perubahan data.

No.	Nama Peneliti dan Judul Penelitian	Metode	Persamaan	Perbedaan	Hasil Output
4.	Devanada Anisza Putri, Aris Sunarya, & Zainal Fatah (2023) "Akuntabilitas Pengadaan Barang atau Jasa E-Katalog di Kelurahan Wonokromo Kota Surabaya"	Metode Analisis Data Kualitatif	Membahas mengenai pengadaan barang serta akuntabilitas kinerja pada sistem pengadaan barang atau jasa guna meningkatkan kinerja, performa, dan efisiensi.	<ul style="list-style-type: none"> • Metode yang digunakan dalam (pengembangan) penelitian tersebut. • Tempat studi kasus yang dijadikan objek penelitian. • Fokus atau tujuan utama dalam 	<p>Akuntabilitas pada penerapan pengadaan barang atau jasa e-katalog di Kelurahan Wonokromo Kota Surabaya relatif akuntabel apabila mengacu pada 3 indikator akuntabilitas yang dijelaskan pada teori BAPPENAS, diantaranya:</p> <ol style="list-style-type: none"> I. Pertama harus ada kesesuaian antara pelaksanaan dan prosedur pelaksanaan yang berdasarkan dengan adanya Peraturan Perundang-Undangan yang menjadi landasan hukum dalam

No.	Nama Peneliti dan Judul Penelitian	Metode	Persamaan	Perbedaan	Hasil Output
				penelitian tersebut.	<p>pelaksanaan pengadaan barang/jasa e-katalog.</p> <p>II. Kedua adanya sanksi atas kesalahan atau kelalaian yang dipersamakan dengan adanya sistem penilaian penyerapan anggaran.</p> <p>III. Ketiga adalah adanya output dan outcome yang terukur.</p>
5.	Edah Jubaedah & Efan Suprastiyo (2022) "Tinjauan Pelaksanaan Pengadaan Barang dan Jasa"	Request For Quotation (RFQ)	Membahas mengenai pengadaan barang serta keefektivan kinerja pada	<ul style="list-style-type: none"> • Metode yang digunakan dalam (pengembangan) 	Untuk menangani masalah HPS, Divisi <i>Supply Chain</i> menghubungi user untuk menanyakan masalah penentuan harga barang dan membahas masalah tersebut

No.	Nama Peneliti dan Judul Penelitian	Metode	Persamaan	Perbedaan	Hasil Output
			sistem pengadaan barang atau jasa guna meningkatkan kinerja, performas, dan efisiensi.	<ul style="list-style-type: none"> • penelitian tersebut. • Tempat studi kasus yang dijadikan objek penelitian • Fokus atau tujuan utama dalam penelitian tersebut. 	dengan <i>user</i> . Selanjutnya, <i>user</i> melakukan survey harga barang yang saat ini tersedia di pasaran untuk mengetahui harganya, baik melalui survey online maupun survey di lokasi. Jika harga benar-benar naik, pihak user akan merubah harga dari harga yang lama ke harga yang baru, dan user menetapkan harga yang baru.
6.	Dara Novtah Eka Sukma, Riska Amelia, Arnita, & Mirza Zoni (2023)	Metode Agile Development	Penelitian tersebut menggunakan metode yang	<ul style="list-style-type: none"> • Tempat studi kasus yang dijadikan 	Perancangan Sistem Informasi Laboratorium Program Studi Teknologi Rekayasa Komputer Jaringan menggunakan

No.	Nama Peneliti dan Judul Penelitian	Metode	Persamaan	Perbedaan	Hasil Output
	"Perancangan Sistem Informasi Laboratorium Program Studi Teknologi Rekayasa Komputer Jaringan Dengan Metode Agile Development Menggunakan Cloud Computing Berbasis Web"		sama, yakni <i>Agile Development Methodology</i>	objek penelitian <ul style="list-style-type: none"> Fokus atau tujuan utama dalam penelitian tersebut. 	Metode Agile dengan Cloud Computing berbasis Web, dapat meningkatkan efisiensi dan efektivitas pengelolaan laboratorium serta memberikan kemudahan akses bagi pengguna terutama untuk memudahkan mahasiswa dan dosen dalam pengelolaan laporan laboratorium, serta meningkatkan efisiensi dalam proses pencarian data.



2.2 Kajian Pustaka

Bagian ini memuat rangkuman teori atau literatur dari dari buku yang mendukung penelitian dan penjelasan tentang konsep dan prinsip dasar yang diperlukan untuk memecahkan masalah. Landasan teori dapat berupa uraian kualitatif, model matematis, atau alat yang langsung berkaitan dengan masalah yang diteliti. Semua sumber teori yang disebutkan di bagian ini harus disertakan dalam kalimat atau pernyataan yang diacu dan dicantumkan dalam Daftar Pustaka. Jika sumber tersebut ditulis dalam Landasan Teori, sumber tersebut harus disesuaikan dengan tema yang telah ditentukan sebelumnya.

2.2.1 Pengadaan Barang

Pengadaan barang dan jasa erat kaitannya dengan pembuatan fasilitas baru, jalan, rumah sakit, gedung perkantoran, dan alat tulis disebuah instansi pemerintah. Tender yang sering disebut pengadaan barang dan jasa sebenarnya terjadi bukan hanya di instansi pemerintah. Pengadaan barang dan jasa ini dapat dilakukan oleh perusahaan swasta dan perusahaan nasional maupun internasional. Pengadaan barang dan jasa pada dasarnya dilakukan untuk memenuhi kebutuhan bisnis atau lembaga pemerintah akan barang dan jasa yang dapat meningkatkan kinerja dan performa mereka. (Faisal, Morasa, & Mawikere, 2017)

Aplikasi sistem pengadaan barang adalah sebuah perangkat lunak yang dirancang khusus untuk membantu berjalannya proses pengadaan barang dalam suatu organisasi atau perusahaan. Sistem ini mencakup berbagai tahapan, mulai dari pengajuan permintaan barang, pengadaan, pemilihan vendor atau pemasok, hingga proses pembayaran.

2.2.2 Database

Database atau basis data adalah kumpulan data yang dikelola sedemikian rupa berdasarkan ketentuan tertentu yang saling berhubungan sehingga mudah dalam pengelolaannya.

Melalui pengelolaan tersebut pengguna dapat memperoleh kemudahan dalam mencari informasi, menyimpan informasi dan membuang informasi. (Intern, 2020)

Berikut ini adalah komponen dalam database adalah:

1. **Data**

Data dapat didefinisikan sebagai file yang berisi informasi dalam bentuk teks, gambar, log, atau format lainnya. Struktur tertentu yang mudah dioperasikan diperlukan untuk penyimpanan. Struktur itu antara lain:

- **Field** merupakan satuan informasi rinci, misal nama produk, stok, harga, dan lain-lain.
- **Record** adalah kumpulan field membentuk informasi unik, misal harga suatu produk.
- **Table** adalah kumpulan record, yaitu isi dari file.
- **Database** adalah kumpulan tabel dan file.

2. **Hardware**

Hardware adalah perangkat keras yang digunakan untuk menyimpan dan mengatur data. Untuk penyimpanan lokal pada jaringan tertentu, *hardware* yang digunakan termasuk *disk*, komputer, memori, dan lainnya. Jika data disimpan secara *online*, dapat menggunakan *website* dan *server hosting*.

3. **Sistem Operasi**

Untuk mendukung database yang dibangun, sistem operasi harus memilih Windows atau Linux.

4. **Database Management System**

Banyaknya pilihan aplikasi database (DBMS) memudahkan dalam penginputan dan update data. Contohnya Anda dapat menggunakan MySQL untuk mengelola database di website.



2.2.3 PHP

PHP: Hypertext Preprocessor (yang awalnya disebut dengan *Personal Home Pages*) atau hanya **PHP** saja, merupakan bahasa pemrograman yang umum digunakan untuk pengembangan web. Bahasa ini pertama kali dikembangkan oleh programmer asal Denmark-Kanada **Rasmus Lerdorf** pada tahun 1994. Sekarang PHP Group membuat implementasi referensi PHP. Yang awalnya, PHP merupakan singkatan dari **Personal Home Page**, tetapi sekarang merupakan singkatan dari inisialisasi rekursif **PHP: Hypertext Preprocessor**. (Labdajiwa, 2021)

Sebagian besar, kode PHP diproses di peladen web oleh interpreter PHP yang dijalankan sebagai *daemon*, modul, atau *Common Gateway Interface* (CGI). Di server web, output dari kode PHP yang ditafsirkan dan dieksekusi (yang dapat berupa berbagai jenis data, seperti data gambar biner atau HTML) akan membentuk respons HTTP secara keseluruhan atau sebagian. Beberapa sistem template web, sistem manajemen konten web, dan kerangka kerja web dapat digunakan untuk mengatur atau memudahkan pembuatan respons ini. Selain itu, PHP dapat digunakan untuk berbagai tugas pemrograman di luar konteks web, seperti aplikasi grafis mandiri dan kontrol drone robotik. Kode PHP juga dapat dieksekusi langsung dari baris perintah.

Adapun alasan PHP merupakan bahasa pemrograman yang lebih banyak digunakan programmer dan populer diantara bahasa pemrograman lainnya. (Awwaabiin, 2021) Hal tersebut dikarenakan :

a) Cenderung mudah dipelajari

PHP merupakan bahasan pemrograman yang termasuk sangat mudah dipelajari dibandingkan dengan Bahasa pemrograman terkenal lainnya.

b) Kompatibilitas dengan HTML yang bagus

Penggunaan script PHP tidak merusak HTML, yang justru keduanya saling melengkapi dan selaras.



c) PHP bersifat open

Bahasa pemrograman PHP yang bersifat terbuka sehingga dapat diakses dan digunakan siapa saja tanpa perlu mengeluarkan biaya sama sekali.

d) Fleksibilitas yang tinggi

PHP dapat dikolaborasikan dengan Bahasa pemrograman yang lain, hal itu dapat mempermudah developer dalam proses pengembangan atau penyesuaian sistem yang dibuat dengan kebutuhan yang diperlukan.

e) Mendukung layanan cloud

Meskipun dari segi umur Bahasa pemrograman ini hampir mencapai dua dekade, akan tetapi PHP bisa mendukung layanan cloud dengan cukup baik.

f) Materi pembelajaran yang banyak

PHP bukan termasuk bahasa pemrograman yang baru, dari hal tersebut tentunya banyak sekali forum, komunitas yang saling bertukar informasi, saling mendiskusikan, serta dokumentasi mengenai project maupun panduan.

g) Pilihan database yang banyak

Bahasa pemrograman PHP bisa menggunakan hampir semua jenis database seperti MySQL, hingga non-raltional database.

h) Kecepatan tinggi

PHP telah terbukti dapat meningkatkan kecepatan dalam proses akses (loading) dibandingkan dengan Bahasa pemrograman yang lain.

i) *Multi-platform*

PHP mampu digunakan pada berbagai macam *operating system* (OS). Seperti pada Linux, Windows, hingga MacOS.

j) Selalu diperbarui

Memang PHP pertama kali sudah lama ditemukan semenjak tahun 1995, akan tetapi PHP juga ikut mengalami perkembangan. Bahkan untuk sekarang PHP sudah memiliki versi 8.2.12.



2.2.4 XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung beragam jenis sistem operasi, serta merupakan kompilasi dari beberapa program.

XAMPP merupakan singkatan dari X (tempat sistem operasi apapun), Apache, MySQL, PHP, dan Perl. Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri dari program Apache HTTP Server, database MySQL, dan penerjemah Bahasa yang ditulis dengan Bahasa pemrograman PHP dan Perl. Program ini bebas dan tersedia dalam lisensi umum GNU. XAMPP adalah web server yang mudah digunakan yang dapat menyediakan tampilan halaman web yang dinamis. Anda dapat mengambilnya langsung dari web resminya. (Loudtiago, 2023)

XAMPP merupakan singkatan dari lima komponen utama, yaitu:

- **X – Cross Platform.** X adalah kepanjangan XAMPP yang artinya Cross. Hal itu sebagai penanda bahwa XAMPP adalah aplikasi *Cross Platform* yang kompatibel dengan berbagai jenis OS.
- **A – Apache.** Apache adalah web server default yang dipakai XAMPP. *Web server* adalah software untuk mengatur proses transfer data antara website dan pengunjung. Selain itu, Apache juga bertugas sebagai distributor untuk fitur atau komponen-komponen lain.
- **M – MariaDB.** MariaDB adalah software DBMS default milik XAMPP. DBMS adalah software untuk mengelola database beserta data di dalamnya. MariaDB menggantikan software DBMS sebelumnya, MySQL sejak XAMPP versi 5.5.30 dan 5.6.14.
- **P – PHP.** PHP adalah bahasa pemrograman untuk membangun website atau web app dari sisi *back-end*.
- **P – Perl.** Kepanjangan XAMPP yang terakhir yaitu Perl. Perl adalah contoh bahasa pemrograman untuk keperluan yang lebih kompleks. Misalnya pembuatan aplikasi GUI,



pemrograman jaringan, dan banyak lagi. (Muhammad, 2022)

XAMPP memiliki beberapa komponen penting yang harus Anda ketahui. (Maksum, 2022) Berikut ini beberapa komponen utama XAMPP :

➤ **XAMPP Control Panel**

XAMPP control panel memungkinkan Anda mengaktifkan fungsi Apache, MySQL, Filezilla, Config, Netstat, dan konfigurasi lainnya..

➤ **Htdoccs**

Htdocs merupakan komponen XAMPP yang berbentuk folder. Folder htdocs berfungsi sebagai tempat penyimpanan folder dan file, yang dapat ditampilkan dengan *browser*. Dalam hal *hosting*, htdocs adalah folder *public*. Kapasitas partisi yang ada sebanding dengan folder htdocs. Letak folder htdocs biasanya berada pada path *C:\xampp\htdocs*.

➤ **Config**

Komponen berikutnya adalah konfigurasi. Konfigurasi adalah komponen XAMPP yang digunakan untuk mengatur pengaturan dasar, seperti mengatur editor teks dan *browser* yang akan digunakan secara *default* oleh aplikasi XAMPP.

➤ **Netstat**

Salah satu komponen XAMPP berikut adalah *Netstat*, dapat digunakan untuk mengetahui apakah port yang digunakan oleh XAMPP telah digunakan oleh aplikasi lain. Jika port standar XAMPP digunakan oleh aplikasi lain, maka aplikasi XAMPP tidak akan berfungsi dengan baik. Selain itu, Anda harus mengganti *port* yang bermasalah dengan *port* lain yang belum digunakan.

➤ **PhpMyAdmin**

Dan komponen yang terakhir adalah *PhpMyAdmin*. *PhpMyAdmin* merupakan komponen yang bertugas dalam hal manajemen database melalui browser.



2.2.5 Agile Development Methodology

Metode *Agile* adalah salah satu teknik pengembangan *software* yang berdasarkan atas protokol pengerjaan berulang yang terdiri dari solusi dan aturan yang telah diputuskan. Metode ini digunakan melalui sistem kolaborasi tim yang terorganisir dan terstruktur.

Metode *Agile* sangat cocok untuk proyek jangka pendek karena sangat mudah disesuaikan dengan perubahan dalam proyek. Tim yang dapat membuat keputusan dengan cepat adalah kunci *Agile Development Methodology*.

Keputusan yang diambil tidak hanya cepat dari segi waktu, tetapi juga memiliki kualitas prediksi yang baik. Sehingga dapat mengatasi permasalahan saat ini tanpa menimbulkan masalah baru.



Gambar 2. 1 Tahapan-tahapan pada agile development methodology

Agile Development memiliki 7 tujuan sebagai berikut:

1) **High Value & Working App System**

Tujuan yang pertama adalah *high value and working app system*. Artinya, dalam pembuatan *software* haruslah memperhatikan nilai dari suatu *software* tersebut, berusaha membuat *software* yang bernilai tinggi meskipun dengan *budget* yang seminimal mungkin. Akan tetapi, kualitas serta fungsi dari *software* tersebut tidak kalah penting dan tidak boleh disepelekan.

2) ***Iterative, Incremental, Evolutionary***

Agile adalah salah satu metode pengembangan *software* yang memungkinkan perubahan secara bertahap dan berulang. Dengan kata lain, teknik ini cocok untuk proyek IT jangka pendek karena fleksibel.

3) ***Cost Control & Value Driven Development***

Proses pengembangan *software* dapat disesuaikan dengan kebutuhan pengguna atau *user*. Tim pengembang juga dapat mengontrol biaya dan waktu yang dibutuhkan untuk proses tersebut.

4) ***High Quality Production***

Produk *software* atau perangkat lunak harus tetap berkualitas meskipun menggunakan biaya seminimal mungkin.

5) ***Flexible & Risk Management***

Jadwal pertemuan dengan klien yang dapat diatur kapan saja adalah contoh tujuan *fleksibel*. Oleh karena itu, fungsi perangkat lunak yang sedang dikembangkan masih dapat dilaksanakan. Cara ini juga digunakan untuk mengurangi kesalahan program sebelum proses *deploy aplikasi* selesai.

6) ***Collaboration***

Kolaborasi sangat penting dilakukan oleh tim developer untuk membahas *feedback* dari klien. Oleh karena itu diperlukan koordinasi serta komunikasi yang baik dengan seluruh anggota tim.

7) ***Self Organizing, Self Managing Teams***

Dengan menggunakan metode *Agile*, pihak *developer* dapat memiliki kemampuan untuk mengatur proses pengembangan *software* secara mandiri. Manajer bertanggung jawab sebagai penghubung antara *client* dan *developer*. Dengan begitu, dimungkinkan untuk mengurangi kemungkinan kesalahan dalam komunikasi.

Setelah mengetahui tujuan *agile*, faktor manusianya juga sangat penting untuk diperhatikan, diantaranya adalah:



- a) Kompetensi, yaitu segala sesuatu yang mencakup mengenai pengetahuan dan keterampilan dalam proses pengembangan software.
- b) Fokus, yang berarti setiap anggota tim harus fokus pada satu tujuan meskipun masing-masing dibagi tugasnya sendiri.
- c) Kolaborasi, kerja sama antara anggota tim, manajer, dan *client*.
- d) Kemampuan untuk mengatasi permasalahan dengan cepat dan tepat.
- e) *Fuzzy problem solving* untuk menentukan prioritas masalah yang harus diselesaikan lebih dulu.
- f) Menghormati dan saling percaya satu sama lain sehingga dapat terwujudnya tim yang lebih solid.
- g) Manajemen diri pada tiap tiap masing-masing anggota dalam menyelesaikan proyek, mengatur proses pengembangan sesuai dengan lingkungan, dan memastikan bahwa hasil dikirim sesuai jadwal.



Gambar 2. 2 Alur atau tahapan garis besar agile development methodology

Adapun tahapan-tahapan dalam penggunaan metode *Agile* diantaranya adalah *Planning*, *Design*, *Development*, *Testing*, *Deployment*, *Review*, *Launch* dan *Maintenance* sistem. (Lubis, 2023) Berikut ini adalah penjelasan mengenai tahapan-tahapan tersebut:

1] Tahap *Planning*

Perencanaan merupakan tahapan yang paling awal (*fundamental*) dalam *Agile Development Methodology*. Tujuannya adalah untuk memahami tujuan awal proyek, merencanakan langkah-langkah dalam proses pengembangannya, serta mengidentifikasi kebutuhan yang diperlukan.

2] Tahap *Design*

Tahap perancangan merupakan tahapan lanjutan dari tahapan awal *Agile Development Methodology*, dimana pada tahapan tersebut pihak *developer* akan mengembangkan rancangan awal menjadi lebih terperinci agar lebih jelas lagi mengenai apa saja yang akan dikerjakan selanjutnya. Desain ini mencakup beberapa aspek diantaranya seperti *user interface*, visual, dan struktur keseluruhan *software* (produk).

3] Tahap *Development*

Tahap pengembangan merupakan tahapan dimana pihak *developer* mulai mengeksekusi rencana perancangan yang sudah ada menjadi sebuah *syntax* atau *script* yang berfungsi.

4] *Testing*

Tahap pengujian merupakan langkah yang dilakukan untuk memastikan pada sistem atau *software* yang telah dikembangkan berfungsi sebagaimana mestinya dan tidak ada kendala *error* pada sistem saat dijalankan.

5] Tahap *Deployment*

Tahap penyampaian adalah tahapan yang dilakukan setelah adanya pengetesan pada fitur-fitur yang telah dikembangkan sudah berfungsi sebagaimana mestinya, sistem tersebut akan disampaikan kepada *client* mengenai sistem setelah adanya pengembangan. Penyampaian tersebut dilakukan secara *continues* (berkelanjutan dan berulang-ulang) setiap adanya progress pengembangan yang ada pada sistem.



6] Tahap *Review*

Tahap ulasan adalah tahapan dimana saat semua tim (*developer*) menerima *feedback* dan saran terkait sistem yang telah dikembangkan, baik dari pihak *client*, pengguna (*tester*) maupun anggota tim lainnya.

7] Tahap *Launch*

Tahap peluncuran adalah saat produk akhirnya siap untuk dirilis ke pengguna akhir. Setelah melalui berbagai tahap pengembangan, pengujian, dan perbaikan, produk dianggap telah mencapai kualitas yang memadai untuk digunakan oleh pengguna.

Metode *Agile* memiliki berbagai varian atau *framework* yang dirancang untuk menyesuaikan dengan kebutuhan dan konteks proyek yang berbeda. (Toms, 2023) Berikut adalah beberapa jenis metode *Agile* dan contohnya:

➤ **Scrum**

Scrum adalah *framework Agile* yang paling populer dan sering digunakan, yang menekankan pada iterasi (*sprint*) berdurasi pendek dan pengelolaan proyek yang terstruktur. Dalam *Scrum*, peran seperti *Scrum Master* dan *Product Owner* menjadi penting.

➤ **Kanban**

Kanban adalah metode yang menekankan pada aliran kerja visual dan pengelolaan waktu untuk meningkatkan efisiensi. Papan *Kanban* digunakan untuk melacak progress pekerjaan.

➤ **Lean Software Development**

Lean menekankan pada pengelolaan hambatan dan peningkatan berkelanjutan untuk menciptakan nilai bagi pelanggan dengan efisiensi maksimal.

➤ **Extreme Programming (XP)**

XP adalah metode yang menekankan pada kepuasan pelanggan dan kemampuan untuk beradaptasi terhadap perubahan. Ini melibatkan praktik seperti pengkodean berpasangan dan pengembangan berkelanjutan.



- **Feature Driven Development (FDD)**
FDD fokus pada pengembangan berdasarkan fitur dan membangun sistem dengan pendekatan iteratif dan inkremental.
- **Dynamic System Development Method (DSDM)**
DSDM adalah metode yang menekankan pada kecepatan pengiriman dan kualitas produk, dengan fokus pada kolaborasi antara *stakeholder*.
- **SAFe (Scale Agile Framework)**
SAFe adalah kerangka kerja untuk menerapkan prinsip-prinsip Agile pada skala organisasi yang lebih besar, dengan struktur dan panduan untuk bekerja di tingkat tim, program, dan portofolio.
- **Less (Large-Scale Scrum)**
Less adalah *framework* yang mengadaptasi prinsip-prinsip dan praktik *Scrum* untuk digunakan dalam proyek skala besar dengan banyak tim.
- **Crystal**
Crystal adalah keluarga metodologi *Agile* yang menekankan pada komunikasi dan interaksi antar individu, serta adaptasi terhadap kebutuhan proyek.
- **Agile Unified Process (AUP)**
AUP adalah versi ringan dari *Rational Unified Process* (RUP) dengan fokus pada *simplicitas* dan fleksibilitas.

Agile software development sendiri memiliki empat nilai inti. Berikut ini adalah keempat nilainya:

1] **Interaksi antar individu lebih penting dibandingkan proses dan alat**

Nilai yang menunjukkan bahwa pengembangan perangkat lunak tidak akan berhasil tanpa kolaborasi antara tim pengembang dan klien serta komunikasi yang efektif untuk mencapai hasil maksimal.



2] Proses pengembangan *software* lebih penting dibandingkan dengan dokumentasi

Pada masa lalu, tim pengembang menghabiskan banyak waktu untuk membangun dokumentasi yang mendalam. Pada *agile*, dokumentasi ini dapat berupa kisah pengguna untuk membantu tim pengembang mengembangkan fitur. Dokumentasi ini penting, tetapi perangkat lunak berjalan dengan baik.

3] Kolaborasi dengan klien lebih penting dibandingkan dengan kontrak

Kontrak penting, akan tetapi produk pengembang mungkin berbeda dari rancangan awal yang telah disepakati. Dengan adanya kerja sama dengan *client*, tim pengembang dapat menyesuaikan mereka sesuai dengan keinginan *client* dan apa yang ditulis dalam kontrak.

4] Menanggapi perubahan lebih penting daripada mengikuti rencana

Setiap tim pengembang harus siap dengan perubahan karena *client* akan terus meminta perubahan dan revisi sampai *client* merasa puas dengan produk yang dibuat.

Metode *agile* ini dapat dirasakan oleh semua orang yang terlibat dalam prosesnya, termasuk pengembang, klien, dan juga vendor.

Peningkatan produktivitas adalah keuntungan bagi *developer*, karena setiap tim memiliki kemampuan untuk menyelesaikan semua tugas tanpa perlu menunggu tim lain untuk menyelesaikannya. Tidak hanya itu, *developer* memiliki banyak waktu untuk berkembang seperti yang mereka inginkan.

Vendor dapat menghemat uang dan berkonsentrasi pada pengembangan fitur dan peningkatan efisiensi dengan mengetahui apakah vendor memenuhi permintaan pengembangan. (Setiawan, 2021)



2.2.6 Blackbox Testing

Black box testing atau dapat disebut juga *Behavioral Testing* merupakan pengujian yang dilakukan di akhir pembuatan perangkat lunak untuk mengamati hasil input dan output tanpa mengetahui struktur kodenya. (Setiawan, 2021)

Ada beberapa teknik yang biasanya digunakan untuk menguji perangkat lunak. Berikut ini adalah teknik-tekniknya:

- **All pair testing**

Teknik *all pair testing* ini dikenal juga dengan *pairwise testing*. Pengujian ini digunakan untuk menguji semua kemungkinan kombinasi dari seluruh pasangan berdasarkan input parameternya.

- **Boundary value analysis**

Teknik ini berfokus pada pencarian error dari luar atau sisi dalam perangkat lunak.

- **Cause-effect graph**

Berikutnya adalah teknik *cause-effect graph*. Teknik pengujian ini menggunakan grafik sebagai patokannya. Grafik ini menggambarkan relasi antara efek dan penyebab dari *error*.

- **Equivalence partitioning**

Teknik ini bekerja dengan cara membagi data *input* dari beberapa perangkat lunak menjadi beberapa partisi data.

- **Fuzzing**

Fuzzing merupakan teknik pencarian bug dalam perangkat lunak dengan memasukkan data yang tidak sempurna.

- **Orthogonal array testing**

Selanjutnya adalah *orthogonal array testing*. Teknik ini digunakan jika input berukuran kecil, akan tetapi cukup berat jika digunakan dalam skala yang besar.

- **State transition**

Terakhir adalah *state transition*. Teknik ini berguna untuk melakukan pengujian terhadap mesin dan navigasi dari UI dalam bentuk grafik.



2.2.7 UML

Unified Modeling Language (UML) adalah himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (OOP), serta bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak. UML adalah metodologi pengembangan sistem berorientasi objek (OOP) dan sekelompok perangkat untuk mendukung pengembangan sistem tersebut. UML pertama kali diperkenalkan oleh Object Management Group pada tahun 1980-an dan sekarang banyak digunakan oleh para praktisi OOP.

UML juga merupakan dasar bagi perangkat desain berorientasi objek (OOP) dari IBM dan juga merupakan standar visualisasi. (Ramadhanti, 2023) Untuk mengetahui pendapat para ahli mengenai pengertian UML, berikut pemaparannya.

- **Menurut Booch**, UML merupakan bahasa standar yang digunakan dalam perancangan sebuah sistem.
- **Menurut Nugroho**, UML merupakan metode kolaborasi antara metode-metode booch, OMT (*Object Modeling Technique*), serta OOSE (*Object Oriented Software Engineering*) yang digunakan untuk menganalisa perancangan sistem.
- **Menurut Herlawati**, UML merupakan kumpulan diagram yang terdiri dari sembilan diagram atau delapan diagram.

Berikut ini adalah beberapa fungsi UML yang perlu diketahui. Diantaranya adalah:

1. Memodelkan Sistem Perangkat Lunak

Unified Modeling Language menggambarkan berbagai aspek sistem perangkat lunak, seperti struktur, perilaku, dan interaksi antara objek.

2. Meningkatkan Komunikasi dan Kolaborasi

Unified Modeling Language membantu pengembang berkomunikasi dengan baik dan menyederhanakan pemodelan sistem yang kompleks dengan memberikan notasi grafis yang dapat digunakan untuk menampilkan ide dan



konsep yang rumit secara visual.

3. Meningkatkan Kualitas Perangkat Lunak

Unified Modeling Language membantu pengembang perangkat lunak mengurangi kesalahan dan meningkatkan kualitas produk akhir. Dengan memodelkan sistem secara visual, pengembang dapat menemukan masalah dan kesalahan sebelum produk akhir dibuat.

4. Meningkatkan Efisiensi Pengembangan Perangkat Lunak

Unified Modeling Language memudahkan pengembang untuk mengembangkan perangkat lunak dengan cepat dan efisien karena meningkatkan pemahaman sistem secara keseluruhan dan mengurangi kompleksitas.

5. Memastikan Kepuasan Klien

Dalam pengembangan perangkat lunak, membantu memastikan bahwa produk akhir sesuai dengan kebutuhan dan persyaratan klien. Para pengembang dapat memperjelas dan menyederhanakan pemodelan sistem dengan menggunakan *Unified Modeling Language*. (Faulina, 2023)







Ada beberapa jenis dan juga bagian dari UML, di antaranya yaitu :






- **Use Case Diagram**

Untuk menggambarkan model proses bisnis dari sudut pandang pengguna sistem, diagram use case digunakan. Diagram ini terdiri dari *aktor* dan *use case*. *Actor* mewakili orang yang akan mengoperasikan sistem atau orang yang berinteraksi dengan sistem aplikasi yang dibuat, sedangkan *use case* mewakili operasi yang dilakukan oleh aktor.



Tabel 2. 2 Tabel Simbol-simbol Use Case Diagram

Simbol	Nama	Keterangan
	Actor	Mewakili peran orang, sistem yang lain, atau alat komunikasi dengan use case
	Use case	Abstraksi dan interaksi antara sistem dan aktor
	Association	Apa yang menghubungkan antara objek satu dengan objek yang lainnya
	Dependency	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (independent).
	generalization	Hubungan dimana objek anak (dependent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor).
	Include	Menunjukkan bahwa suatu use case






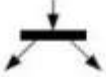
Simbol	Nama	Keterangan
		seluruhnya merupakan fungsionalitas dari use case lainnya
	Extend	Menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi terpenuhi
	Sistem	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	Collaboration	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
	Note	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.


- **Activity Diagram**

Activity Diagram dibuat untuk menjelaskan aktivitas komputer dan alur aktivitas organisasi. Ini adalah bentuk visual dari alir kerja yang berisi tindakan

dan aktivitas user dalam sebuah sistem aplikasi. Diagram ini menunjukkan alur sistem aplikasi secara garis besar.

Tabel 2. 3 Tabel Simbol-simbol Activity Diagram

Simbol	Nama	Deskripsi
	Status Awal	Status awal aktifitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
	Percabangan	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari 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
	Fork	Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel

Simbol	Nama	Deskripsi
	Join	Digunakan untuk menunjukkan kegiatan yang digabungkan


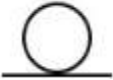
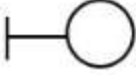



- **Sequence Diagram**

Diagram ini adalah kombinasi *diagram class* dan *object* yang menunjukkan model statis, tetapi ada juga yang dinamis.

Diagram ini menunjukkan proses operasi, pesan yang akan dikirim, dan kapan operasi dilakukan.

Diagram yang diatur berdasarkan waktu ini memiliki objek yang terkait dengan proses berlangsungnya operasi, yang diurutkan dari kiri ke kanan.

Tabel 2. 4 Tabel Simbol-simbol Sequence Diagram

Simbol	Nama	Keterangan
	Actor	Menggambarkan orang yang sedang berinteraksi dengan sistem
	Entity Class	Menggambarkan hubungan yang akan dilakukan
	Boundary Class	Menggambarkan sebuah gambaran dari foem
	Control Class	Menggambarkan penghubung antara boundary dengan table
	A focus of Control & A Life Line	Menggambarkan tempat mulai dan berakhirnya message
	A message	Menggambarkan pengiriman pesan



- **Class Diagram**

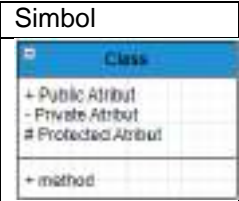
Class diagram atau diagram kelas adalah salah satu jenis diagram struktur pada UML yang menggambarkan dengan jelas struktur serta deskripsi *class*, atribut, metode, dan hubungan dari setiap objek. Ia bersifat statis, dalam artian diagram kelas bukan menjelaskan apa yang terjadi jika kelas-kelasnya berhubungan, melainkan menjelaskan hubungan apa yang terjadi. (Setiawan, Blog: Memahami Class Diagram Lebih Baik, 2021)

Diagram kelas ini melakukan banyak hal, tetapi fungsi utamanya adalah menunjukkan struktur sistem. Ada fungsi tambahan berikut:

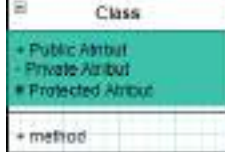
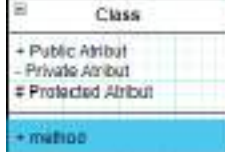
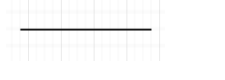
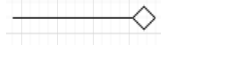
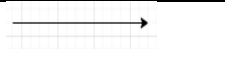
- ✓ Menunjukkan struktur sistem dengan jelas;
- ✓ Meningkatkan pemahaman tentang gambaran umum atau skema program;
- ✓ Dapat digunakan untuk analisis bisnis dan membuat model sistem dari sisi bisnis; dan
- ✓ Memberikan gambaran tentang sistem atau perangkat lunak dan hubungannya.

Setiap komponen berada di bagian yang berbeda. *Class* (*upper section*), atribut (*middle section*), dan metode (*bottom section*). Selain itu, masing-masing komponen dijelaskan di sini. *class diagram*:

Tabel 2. 5 Tabel Simbol-simbol dan Bagian Class Diagram

Simbol	Nama	Keterangan
	Class	Nama dari class tersebut



 <p>UML Class Diagram showing a class with the following elements:</p> <ul style="list-style-type: none"> Public Atribut Private Atribut Protected Atribut method 	Atribut	Atribut berisikan karakteristik objek dan berfungsi menjelaskan kualitas kelas. Atribut dituliskan pada bagian atau kolom kedua, dan hanya digunakan untuk mendeksripsikan <i>instance</i> dari suatu kelas.
 <p>UML Class Diagram showing a class with the following elements:</p> <ul style="list-style-type: none"> Public Atribut Private Atribut Protected Atribut method 	Metode	Metode atau operasi (<i>operation</i>) dituliskan pada bagian atau kolom ketiga. Metode menggambarkan perilaku kelas, seperti bagaimana interaksi kelas dengan data dan apa yang bisa dilakukan oleh kelas.
 <p>UML Association Diagram showing a simple line representing an association between two classes.</p>	Asosiasi	Asosiasi dapat diartikan sebagai hubungan antara dua <i>class</i> yang bersifat statis. Biasanya asosiasi menjelaskan <i>class</i> yang memiliki atribut tambahan seperti <i>class</i> lain.
 <p>UML Aggregation Diagram showing a line with an open diamond at the end, representing aggregation.</p>	Agregasi	Agregasi adalah hubungan antara dua <i>class</i> di mana salah satu <i>class</i> merupakan bagian dari <i>class</i> lain, tetapi dua <i>class</i> ini dapat berdiri masing-masing.
 <p>UML Inheritance Diagram showing a line with an open arrowhead at the end, representing inheritance.</p>	Pewarisan	Pewarisan atau <i>inheritance</i> dapat disebut juga <i>generalization</i> dalam <i>class diagram</i> adalah suatu kemampuan untuk mewarisi seluruh atribut dan metode

		dari class asalnya (<i>superclass</i>) ke class lain (<i>subclass</i>).
--	--	---

Selain diagram-diagram di atas masih ada beberapa lainnya, yaitu :

- Collaboration Diagram,
- Object Diagram,
- Component Diagram, dan
- Deployment Diagram. (Guntoro, 2023)

2.2.8 SMA Negeri Jogoroto

SMA Negeri Jogoroto merupakan salah satu SMA Negeri di Kabupaten Jombang, berlokasi di jalan Jogoroto No.75 B Jogoroto Kecamatan Jogoroto Kabupaten Jombang Jawa Timur. SMA Negeri Jogoroto terus berupaya meningkatkan kualitas pendidikan dengan visi yaitu

“Mewujudkan Peserta didik yang Bertaqwa, Berprestasi, Berinovasi, Berbudaya lingkungan, Berperan serta dalam kehidupan Global”

Sehingga ke depan diharapkan dapat memberikan sumbangsih pada penyediaan sumber daya manusia yang unggul, kompetitif, kreatif, inovatif, arif, dan berakhlakul karimah.

Sekolah ini akan terus mengembangkan program pembelajaran yang inovatif, Pembiasaan literasi, pendalaman materi AKM dan SNBT, mengembangkan minat dan bakat peserta didik, dan memperkuat kerja sama dengan lembaga pendidikan untuk memastikan lulusan SMAN Jogoroto memiliki daya saing yang tinggi di dunia pendidikan dan mempersiapkan masuk perguruan tinggi negeri dan swasta.

