

BAB II

TINJAUAN PUSTAKA

2.1 Teori Terkait

Sehubungan dengan pengembangan “Sistem Informasi Pengelolaan Laporan Evaluasi Diri Prodi”, terdapat beberapa teori relevan yang dapat menjadi landasan dalam pengembangan sistem ini. Pertama teori manajemen informasi penting untuk merancang sistem yang efisien dalam mengelola data penilaian, termasuk proses pengumpulan, penyimpanan, analisis, dan pelaporan data. Konsep manajemen informasi memberikan kerangka kerja untuk memahami cara terbaik mengintegrasikan dan menggunakan informasi sebagai bagian dari manajemen kursus.

Berdasarkan hasil penelitian yang dilakukan oleh Sri Hasta Mulyani, Ariyanto Nugroho, Marselina Endah Hiswati, dan Catur Paryadi (2019), melakukan Audit Mutu Internal (AMI) yang mencakup ranah akademik dan non-akademik merupakan langkah penting dalam implementasi Sistem Penjaminan Mutu Internal. Semua program studi termasuk dalam kategori akademik, sedangkan dekanat, lembaga, pusat, dan biro UNRIYO diklasifikasikan sebagai non-akademik. Badan Penjaminan Mutu UNRIYO saat ini tidak memiliki sistem informasi evaluasi mutu internal yang memadai untuk memenuhi kebutuhan organisasi, terutama yang berkaitan dengan dokumen

akreditasi. Penelitian ini mencoba untuk mengatasi kekurangan tersebut. Peramban Google Chrome akan digunakan untuk pengujian dan pengembangan sistem dengan menggunakan *framework CodeIgniter* [3].

Penelitian yang telah dilakukan oleh Sirait, Jumaria; Manurung, Suprpto; dan Simanjuntak, Harry Cristofel (2022) mengenai menyajikan wawasan yang lebih mendalam tentang efektivitas program pelatihan yang telah dijalankan serta mengevaluasi tingkat pencapaian program-program studi terhadap standar akreditasi yang telah ditetapkan. Dengan adanya informasi ini, diharapkan dapat ditemukan rekomendasi-rekomendasi atau langkah-langkah perbaikan yang diperlukan untuk meningkatkan mutu pendidikan di lingkungan PRODI [4].

Penelitian yang telah dilakukan oleh Istianah Muslim, Maksun Rois Adin Saf, Rika Perdana Sari, dan Silvana Rasio Henim (2021) meskipun pelaksanaannya terkadang terhambat oleh sumber daya yang terbatas dan proses yang rumit, Audit Mutu Internal (AMI) merupakan pemeriksaan tahunan yang penting untuk menjamin terpenuhinya kriteria Sistem Penjaminan Mutu Internal (SPMI) perguruan tinggi. Untuk menyiasati hal ini, sebuah sistem informasi online dibuat dengan menggunakan teknik prototipe yang mendigitalkan dan mengotomatisasi AMI, sehingga meningkatkan efektivitas dan efisiensinya. Sebuah studi kasus yang dilakukan di Politeknik Caltex

Riau (PCR) menunjukkan bagaimana sistem ini meningkatkan efektivitas dan efisiensi AMI dengan mengotomatisasi pelaporan, pencatatan hasil, dan melakukan tindak lanjut. Sistem ini juga meningkatkan kinerja SPMI, mencapai tingkat penggunaan 87% yang luar biasa [5].

Dalam penelitian yang dilakukan oleh Dias, M., Siswanto, L., Politeknik S, Pontianak N, dan Yani, J. (2023) yang berjudul 'Pengembangan Aplikasi Berbasis Web "SIMAMI" dengan Penambahan Fitur Laporan Evaluasi Diri Program Studi (LEDI PRO)', menekankan kesulitan dalam menerapkan prosedur penjaminan mutu internal di Politeknik Negeri Pontianak, terutama yang berkaitan dengan audit mutu internal, yang saat ini masih dilakukan secara manual. Oleh karena itu, diantisipasi bahwa (SIMAMI) akan tersedia untuk digunakan oleh banyak pemangku kepentingan, termasuk jurusan, program studi, administrasi institusi, dan auditor internal SPMI [6].

Menurut, Herpendi, Agustian Noor, Khairul Anwar Hafizd, dan Fathurrahmani (2024) Untuk mengatasi ketidakefisienan dalam proses audit menciptakan SI AMI (Sistem Informasi Audit Mutu Internal) di Politeknik Negeri Tanah Laut (Politala). Cara audit yang dilakukan saat ini adalah dengan menggunakan *Microsoft Office Word* untuk mengisi formulir dan berita acara, yang kemudian dicetak dan diarsipkan bersama dokumen fisik. Cara ini kurang efektif dan berbahaya untuk

penyimpanan data. Pemodelan prototipe, *CodeIgniter Framework*, dan *SQL Server* digunakan dalam pengembangan sistem yang baru. Dengan data yang secara otomatis dikumpulkan dan dihubungkan dari pengisian borang hingga berita acara AMI, SI AMI telah terbukti merampingkan proses pengarsipan dan pencarian dokumen untuk akreditasi, sehingga meningkatkan efektivitas dan efisiensi proses audit [7].

Menurut Dwi Rani Febriyanti di Universitas Budi Luhur (2020) mengalami beberapa kendala, diantaranya adalah kesimpulan hasil audit yang tidak konsisten dan keterlambatan laporan. Sebuah sistem informasi online yang memanfaatkan *PHP* dan *MySQL* melalui *XAMPP* dibuat untuk mengatasi hal ini dan meningkatkan ketepatan waktu dan efisiensi proses audit. Lembaga Penjaminan Mutu di Universitas Budi Luhur berkinerja lebih baik karena sistem ini memiliki checklist otomatis, pelaporan secara real-time, pencatatan hasil audit, dan persiapan resume. Program penjaminan mutu universitas telah meningkat dan laporan audit telah selesai tepat waktu berkat pemasangan teknologi ini [8].

Terakhir, analisis sistem dan teori desain penting dalam membentuk struktur dan fungsi sistem informasi. Dalam konteks proposal ini, teori ini diperlukan untuk mengidentifikasi kebutuhan pengguna, merancang database, dan mengembangkan antarmuka pengguna yang intuitif. Menerapkan sistem yang baik dan mematuhi

standar analisis dan desain sistem yang baik akan membantu memastikan keberhasilan pengelolaan laporan penilaian mandiri program internal.

2.2 Landasan Teori

2.2.1 Situs

Website adalah kumpulan dari halaman-halaman informatif dan menarik yang memuat berbagai elemen, seperti teks, gambar, video, suara, serta konten animasi. Umumnya, *website* terdiri dari beberapa halaman yang saling terkait di bawah satu alamat web atau subdomain, yang dapat diakses oleh publik melalui *World Wide Web (WWW)*. Komponen utama sebuah *website* adalah halaman-halaman *web* yang dapat dijangkau melalui *URL* dan biasanya disimpan di server. Halaman beranda, atau halaman utama, adalah halaman pertama yang dilihat pengunjung saat mengakses situs tersebut. Beberapa situs *web* juga menawarkan konten atau layanan berbayar, seperti langganan, berita, layanan email, atau akses informasi tertentu [9].

2.2.2 Visual Studio Code

Visual Studio Code adalah perangkat lunak editor kode yang gratis dan dikembangkan oleh *Microsoft*. Dapat digunakan di *Windows*, *Linux*, dan *MacOS*, perangkat lunak ini sangat cepat dan ringan. *Visual Studio Code* mendukung berbagai bahasa pemrograman, termasuk *JavaScript*,

TypeScript, dan *Node.js*. Selain itu, perangkat lunak ini memiliki ekosistem yang luas dengan banyak ekstensi yang dapat digunakan untuk bahasa dan lingkungan *runtime* lain seperti *PHP*, *Python*, *Java*, dan *.NET* [10].

2.2.3 Xampp & MySQL

XAMPP merupakan Software sumber terbuka yang berfungsi seperti web server lokal dan menyediakan berbagai program penting. Aplikasi ini kompatibel dengan berbagai sistem operasi, termasuk *Linux*, *Windows*, *MacOs*, dan *Solaris*. *XAMPP* terintegrasi dengan *Apache*, *MySQL*, dan *PHP*, menyediakan lingkungan server lengkap di dalam satu paket.

MySQL adalah salah satu aplikasi server database yang memanfaatkan bahasa pemrograman *SQL (Structure Query Language)*. Tujuannya adalah untuk mengatur data secara teratur dan sistematis. *MySQL* dapat dijalankan di *localhost* tanpa koneksi internet, memungkinkan developer dan programmer untuk mengembangkan aplikasi berbasis web di komputer [11].

2.2.4 Bootstrap

Bootstrap adalah kerangka *CSS* yang dirancang khusus untuk mengembangkan antarmuka depan situs web. *Bootstrap* juga dikenal di kalangan pengembang situs web sebagai salah satu kerangka kerja paling populer untuk pengembangan *CSS*, *HTML*, dan *JavaScript*. Fungsinya

untuk memungkinkan pengembangan website responsif. Artinya laman *web* dapat beradaptasi dengan berbagai ukuran layar di perangkat seperti ponsel, tablet, dan desktop [12].

2.2.5 Codeigniter 3

Menggunakan paradigma *MVC* (*Model, View, Controller*), *CodeIgniter* adalah sebuah *framework PHP open-source* yang memfasilitasi pengembangan halaman *web* dinamis dengan *PHP*. Dengan *CodeIgniter*, para pengembang *web* dapat membangun aplikasi dengan lebih cepat dan mudah dibandingkan dengan membangunnya dari awal. *CodeIgniter* pertama kali dirilis pada tanggal 28 Februari 2006. *Arsitektur Model-View-Controller (MVC) CodeIgniter* membagi fungsionalitas aplikasi dari tampilan dan kontroler, memungkinkan pengembang untuk mengatur kode mereka dengan cara yang lebih logis dan mudah dipahami. Karena perpustakaannya yang luas dengan fitur dan kemampuan yang telah dikonfigurasi sebelumnya, ia menawarkan fleksibilitas dan skalabilitas yang tinggi kepada pengembang aplikasi *web* sekaligus mempercepat proses pengembangan [13].

2.2.6 Hosting

Hosting, juga dikenal sebagai *web hosting*, adalah tempat di mana semua file dan data dari sebuah *website* disimpan, sehingga banyak orang dapat mengaksesnya melalui internet. File dan data dari sebuah *website* dapat mencakup gambar, video, email, script, aplikasi, dan database.

karena pembuatan situs *web* memerlukan *hosting*. Sebelum membuat *website*, harus menyewa hosting terlebih dahulu.

Sama seperti ketika membangun rumah, jika ingin membangun rumah yang besar maka perlu menambah luas lahan. Hal yang sama berlaku untuk situs *web*. Jika ingin membuat *website* dengan kapasitas dan performa tinggi, memerlukan hosting dengan *resource* yang besar. Dengan *hosting* yang tepat, *website* akan dapat diakses dengan lancar dan menangani trafik pengunjung dalam jumlah besar [14].

2.2.7 UML

Unified Modeling Language atau yang disingkat *UML*, Merupakan strategi dalam metode *RPL* (Rekayasa Peranagkat Lunak) yang berfungsi untuk menggambarkan cara kerja sistem, alur, kapasitas, tujuan dan komponen-komponen pengendalian sistem. Ada empat model *UML* yang paling sering digunakan yaitu *Usecase diagram*, *Class diagram*, *Sequence diagram*, dan *Activity diagram*.

1. *Use case Diagram*

Dalam pembuatan sistem monitoring, *Use Case Diagram* digunakan untuk memodelkan perilaku (*behavior*) dari sistem informasi. *Use Case Diagram* bersifat statis dan menampilkan interaksi antara aktor eksternal dan sistem.

2. *Class Diagram*

Class Diagram menggambarkan struktur sistem dengan menampilkan kelas-kelas dalam sistem, atributnya, metode yang

dimiliki, dan hubungan antar kelas. Diagram ini penting untuk memodelkan elemen statis dalam sistem.

3. *Sequence Diagram*

Sequence Diagram, atau Diagram Urutan, mendeskripsikan interaksi antar objek dengan menampilkan pesan yang dikirim dan diterima di antara objek tersebut dalam suatu urutan waktu tertentu.

4. *Activity diagram*

Activity Diagram, atau Diagram Aktivitas, merupakan diagram statis yang menggambarkan aliran aktivitas dalam suatu sistem bisnis atau proses. Diagram ini fokus pada alur kerja atau logika prosedural sistem.