**RANCANG BANGUN SISTEM INFORMASI PKL DENGAN PENGUJIAN WHITE BOX TESTING DAN SYSTEM USABILITY SCALE (STUDI KASUS: PROGRAM STUDI INFORMATIKA UPN VETERAN JAWA TIMUR)**

**SKRIPSI**



**Disusun Oleh :**

**Abdullah Al Fatih**

**21081010089**

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN” JAWA TIMUR**

**SURABAYA**

**2025**

# **BAB I**

# **PENDAHULUAN**

## **1.1 Latar Belakang**

Perkembangan teknologi, khususnya internet telah mengalami transformasi yang signifikan sejak awal kemunculannya hingga saat ini. Saat ini, internet sudah sangat umum digunakan di berbagai bidang kehidupan dan memberikan manfaat seperti akses informasi yang cepat [1]. Dengan berkembangnya internet, sistem informasi juga mengalami kemajuan pesat. Sistem informasi adalah sistem yang dapat didefinisikan dengan mengumpulkan, memproses, menyimpan, menganalisis, menyebarkan informasi untuk tujuan tertentu [2]. Sistem informasi mengacu pada kombinasi dari teknologi, orang, dan proses yang digunakan untuk mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi. Penerapan sistem informasi berguna untuk banyak hal seperti efisiensi operasional, pengintegrasian informasi, pemantauan kinerja, peningkatan pengambilan keputusan, dan lain sebagainya. Kampus UPN Veteran Jawa Timur memanfaatkan penerapan sistem informasi untuk pengintegrasian informasi dan efisiensi operasional praktek kerja lapangan.

Praktek Kerja Lapangan (PKL) adalah salah satu cara untuk mengintegrasikan secara sistematis dan sinkron antara program akademik dengan program penguasaan keahlian yang diperoleh melalui kegiatan kerja langsung di tempat kerja untuk mencapai tingkat keahlian tertentu [3]. PKL juga menjadi wadah bagi mahasiswa untuk mengenal tantangan di lingkungan industri atau organisasi, serta membantu mereka mempersiapkan diri untuk menghadapi tuntutan profesional setelah lulus. Dalam pelaksanaannya, mahasiswa ditempatkan di berbagai perusahaan atau institusi yang relevan dengan bidang studi mereka. PKL biasanya berlangsung selama beberapa bulan, tergantung pada kebijakan universitas. Selain itu, mahasiswa diwajibkan menyusun laporan yang berisi evaluasi pengalaman mereka selama menjalani PKL. Pengelolaan data terkait PKL seperti penempatan, bimbingan, hingga evaluasi mahasiswa dapat menjadi tantangan tersendiri, dan di sinilah sistem informasi PKL memiliki peran penting untuk mengotomatisasi dan memudahkan seluruh proses tersebut. Sistem informasi PKL dirancang untuk membantu mempermudah berbagai proses administratif dan operasional kegiatan yang melibatkan beberapa pihak, seperti mahasiswa, dosen pembimbing, dan dosen penguji. Dengan adanya sistem ini, pengelolaan data terkait penempatan mahasiswa dapat dilakukan dengan lebih efisien, mulai dari pendaftaran, pemilihan lokasi PKL, persyaratan berkas penting hingga evaluasi kegiatan.

Penelitian sebelumnya oleh S. Thya Safitri dan Didi Supriyadi berjudul "Rancang Bangun Sistem Informasi Praktek Kerja Lapangan Berbasis Web dengan Metode Waterfall" [4] memiliki beberapa keterbatasan. Pertama, tidak dilakukan pengujian mendalam pada fitur-fitur utamanya. Kedua, tidak ada pengujian langsung kepada pengguna sistem. Ketiga, tidak dilakukan survei kegunaan untuk mengukur kualitas sistem. Untuk mengatasi kesenjangan ini, penelitian ini akan menggunakan metode White Box Testing dan System Usability Scale. Tujuannya adalah untuk memeriksa kecacatan dan mengukur kualitas sistem informasi PKL secara lebih komprehensif.

White Box Testing—juga dikenal sebagai "glass box testing"—adalah metode pengujian perangkat lunak yang berfokus pada alur kode program. Dalam metode ini, masukan dan keluaran diuji berdasarkan spesifikasi yang dibutuhkan [5]. White Box Testing terdiri dari beberapa teknik pengujian, antara lain control flow testing, path testing atau basic path testing, dan data flow testing. Dalam penelitian ini, penulis memilih menggunakan path testing. Metode ini menguji seluruh kemungkinan jalur dalam kode program, memastikan setiap jalur dilalui minimal satu kali. Path testing sangat efektif untuk mengevaluasi program-program yang kompleks [6].

Kegunaan adalah aspek penting dari keberhasilan sebuah program komputer. Kemudahan penggunaan merupakan aspek fundamental yang harus dicapai, mengacu pada bagaimana seseorang dapat menggunakan program tersebut dengan mudah, terutama bagi mereka yang pertama kali menggunakan program tersebut. System Usability Scale atau SUS adalah teknik pengujian kegunaan suatu program komputer atau aplikasi [7]. System Usability Scale memiliki kelebihan dibandingkan metode lain, yaitu jumlah pertanyaan sedikit—hanya berjumlah 10—dan skor nilai yang mudah dipahami—0 hingga 100 dimana nilai lebih tinggi mengindikasikan kegunaan yang lebih baik [8].

Fokus penelitian ini adalah perancangan dan pembangunan sistem informasi Praktik Kerja Lapangan (PKL) dengan tujuan untuk mengotomatisasi dan mempermudah proses administrasi serta pengelolaan data PKL di perguruan tinggi. Sistem ini diharapkan dapat mengintegrasikan berbagai fungsi penting secara efisien, mulai dari pendaftaran mahasiswa, pemilihan lokasi PKL, pengelolaan berkas penting, hingga evaluasi kinerja mahasiswa. Selain itu, penelitian ini juga menekankan pengujian perangkat lunak dan survei kegunaan, dalam rangka menguji kegunaan dan kualitas dari sistem informasi. Penelitian ini diharapkan dapat bermanfaat sebagai acuan dalam perancangan dan pembangunan perangkat lunak.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang yang telah diuraikan sebelumnya, berikut merupakan rumusan masalah yang hendak penulis selesaikan:

1. Bagaimana merancang dan membangun sistem informasi yang dapat mengintegrasikan proses pendaftaran, penempatan, monitoring, dan evaluasi PKL secara efektif dan efisien?
2. Bagaimana memastikan sistem informasi PKL yang dikembangkan dapat diakses dengan mudah dan memiliki nilai guna oleh semua pihak yang terlibat, seperti mahasiswa, dosen pembimbing, dan dosen penguji?
3. Bagaimana implementasi White Box Testing dan System Usability Scale untuk memastikan keandalan dan kelayakan sistem informasi PKL yang dibangun?

## **1.3 Fokus Penelitian**

Fokus penelitian ini lebih diarahkan pada perancangan dan pembangunan sistem informasi. Penulis akan lebih mengeksplorasi praktek-praktek terbaik dalam merancang dan membangun sistem informasi PKL. Pengujian White Box Testing hanya terbatas pada modul-modul penting, seperti autentikasi, pendaftaran, evaluasi kinerja mahasiswa, dan lain sebagainya. Namun evaluasi keseluruhan sistem akan tetap dilakukan menggunakan metode System Usability Scale.

## **1.4 Tujuan Penelitian**

Penulis melakukan perancangan dan pembuatan Sistem Informasi PKL bertujuan untuk:

1. Merancang dan membangun sistem informasi Praktik Kerja Lapangan (PKL) terintegrasi untuk mempermudah proses pendaftaran, penempatan, pemantauan, dan evaluasi mahasiswa.
2. Menguji keandalan dan memeriksa kecacatan sistem menggunakan metode White Box Testing.
3. Mengukur tingkat kegunaan sistem informasi PKL yang dikembangkan dengan metode System Usability Scale (SUS).
4. Menyediakan akses mudah bagi mahasiswa, dosen pembimbing, dan dosen penguji untuk memantau dan mengelola data Praktik Kerja Lapangan secara efektif dan efisien.

## **1.5 Manfaat Penelitian**

* Bagi Penulis: Penelitian ini dapat memperluas wawasan serta memperdalam kemampuan di bidang pengembangan perangkat lunak.
* Bagi pembaca: Penelitian ini dapat mengenalkan dan menambah ketertarikan pembaca dengan dunia pengembangan perangkat lunak.
* Bagi Pihak Kampus: Penelitian ini dapat mengoptimalkan operasional kegiatan Praktik Kerja Lapangan (PKL) dan meningkatkan produktivitas secara keseluruhan.

**BAB II**

**TINJAUAN PUSTAKA**

**2.1 Penelitian Terdahulu**

Penelitian terdahulu memiliki peran penting sebagai data pendukung dalam sebuah penelitian. Aspek krusial dari penelitian terdahulu adalah keterkaitan atau relevansinya dengan permasalahan yang diangkat dalam penelitian ini. Berikut adalah beberapa penelitian terdahulu yang penulis gunakan sebagai rujukan:

1. Dalam penelitian oleh S. Thya Safitri dan Didi Supriyadi pada tahun 2015 berjudul “Rancang Bangun Sistem Informasi Praktek Kerja Lapangan Berbasis Web dengan Metode Waterfall”, sistem informasi PKL telah dikembangkan berfokus untuk menyelesaikan permasalahan administrasi PKL Perguruan Tinggi ST3 Telkom Purwokerto yang masih serba manual. Metode administrasi tradisional ini menyebabkan pencatatan dan penyimpanan data yang kurang terstruktur dan kurang rapi, pencarian data yang membutuhkan waktu lama, adanya redundansi data, dan lain-lain. Namun kekurangan dari penelitian ini antara lain tidak dilakukan pengujian mendalam pada fitur-fitur utamanya, tidak ada pengujian langsung kepada pengguna sistem dan tidak dilakukan survei kegunaan untuk mengukur kualitas sistem.
2. Penelitian yang dilakukan oleh Mohammad Zaenuddin H, Sri Endang Anjarwani, I Wayan Agus Arimbawa pada tahun 2017 yang berjudul “Rancang Bangun Sistem Informasi Praktik Kerja Lapangan Pada Program Studi Teknik Informatika Universitas Mataram Menggunakan Extreme Programming”, telah dikembangkan sistem informasi PKL secara terstruktur dan menjawab permasalahan administrasi manual kegiatan praktek kerja lapangan di Program Studi Teknik Informatika Universitas Mataram. Sistem informasi PKL tersebut berbasis web dan memiliki fitur seperti pengajuan kegiatan PKL, cetak berkas-berkas resmi, penentuan dosen pembimbing, dan masih banyak lagi. Sistem informasi ini juga terdiri dari 5 orang pemegang hak akses, diantaranya mahasiswa, staf program studi, ketua program studi, dosen, dan perusahaan. Namun metode pengujian User Acceptance Test (UAT) yang digunakan dalam penelitian ini tidak dijabarkan dengan lebih detail, sehingga menimbulkan kesulitan dalam memahami proses pengujiannya.
3. Penelitian "Pengembangan Sistem Informasi Praktik Kerja Lapangan (PKL) Siswa Berbasis Website Menggunakan Metode Extreme Programming (Studi Kasus: SMK Negeri 1 Sumenep)" yang dilakukan oleh M. Ro’if, Tri Afirianto, Satrio Hadi Wijoyo pada tahun 2024 menghasilkan sistem informasi PKL secara terstruktur dan komprehensif dengan metode Extreme Programming. Penelitian ini terdiri dari beberapa tahapan, yaitu identifikasi masalah, studi literatur, eksplorasi informasi, perencanaan, pengembangan, serta kesimpulan dan saran. Pengembangan sistem informasi terbagi menjadi 3 iterasi, yaitu iterasi 1 guru pembimbing, iterasi 2 panitia PKL, dan iterasi 3 kakomli dan siswa. Setiap iterasi melibatkan pengujian unit testing dan acceptance testing, diikuti dengan System Usability Scale untuk mengukur kualitas sistem informasi secara keseluruhan.
4. Penelitian “Penerapan Whitebox Testing pada Pengujian Sistem Menggunakan Teknik Basis Path” yang dilakukan oleh Helmi Rafli Raychan Zen dan Ilyas Nuryasin pada tahun 2024 menguji sistem informasi operasional bisnis laundry. Pengujian dilakukan secara sistematis dimulai dari tahap pembuatan flowchart, pembuatan flowgraph, menghitung cyclomatic complexity, penentuan independent path, pembuatan test case, dan pengujian test case.

Dari beberapa penelitian rancang bangun sistem yang ada, hanya penelitian ketiga yang menekankan pengujian sistem dan pengukuran kualitas sistem informasi secara komprehensif. Padahal pengujian sistem dan survei kegunaan pada sistem informasi merupakan aspek yang penting. Karena jika tidak dilakukan kedua hal tersebut, akan sulit menilai kualitas sistem dari sudut pandang pengguna. Untuk mengatasi hal tersebut, pada penelitian ini penulis akan menggunakan metode white box testing untuk menguji keandalan dan metode system usability scale untuk mengukur kualitas sistem informasi.

**2.2 Dasar Teori**

**2.2.1 Sistem Informasi**

Menurut Sutabri (2012), sistem informasi adalah sebuah sistem dalam suatu organisasi yang memadukan kebutuhan pengolahan transaksi harian dengan fungsi manajerial dan kegiatan strategis. Sistem ini mendukung operasi organisasi dan menyediakan laporan-laporan yang diperlukan kepada pihak luar tertentu. Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (building blocks). Komponen-komponen ini meliputi blok masukan, model, keluaran, teknologi, basis data, dan kendali. Sebagai suatu sistem, keenam blok tersebut berinteraksi satu sama lain, membentuk kesatuan yang berfungsi untuk mencapai tujuan tertentu.

a. Blok masukan (input block)

Masukan mewakili data yang masuk ke dalam sistem informasi. Masukan ini mencakup metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

b. Blok Model (model block)

Blok model merupakan gabungan prosedur, logika, dan model matematika yang mengolah data input serta data tersimpan dalam basis data dengan metode tertentu untuk menghasilkan keluaran yang diinginkan.

c. Blok Keluaran (output block)

Produk dari sistem informasi berupa keluaran yang terdiri dari informasi berkualitas dan dokumentasi. Keluaran ini bermanfaat bagi semua tingkatan manajemen dan pengguna sistem.

d. Blok teknologi (technology block)

Teknologi merupakan "kotak peralatan" dalam sistem informasi. Teknologi digunakan untuk menerima masukan, menjalankan model, menyimpan dan mengakses data, mengolah keluaran, serta membantu kendali sistem secara keseluruhan. Teknologi terdiri dari tiga bagian utama: teknisi (brainware), perangkat lunak (software), dan perangkat keras (hardware).

e. Blok basis data (database block)

Basis data adalah kumpulan data yang saling terkait, tersimpan di perangkat keras komputer dan dimanipulasi menggunakan perangkat lunak. Penyimpanan data dalam basis data penting untuk penyediaan informasi selanjutnya. Data perlu diorganisasi dengan baik agar menghasilkan informasi berkualitas dan mengoptimalkan kapasitas penyimpanan. Untuk mengakses atau memanipulasi basis data, digunakan perangkat lunak khusus yang disebut DBMS (Database Management System).

f. Blok kendali (control block)

Sistem informasi dapat terancam oleh berbagai faktor, mulai dari bencana alam, kebakaran, suhu ekstrem, air, debu, ketidakefisienan, hingga sabotase. Oleh karena itu, perlu dirancang dan diterapkan beberapa mekanisme pengendalian. Tujuannya adalah untuk mencegah kerusakan sistem atau memastikan penanganan yang cepat jika kerusakan telah terjadi [9].

**2.2.2 Website**

Website atau situs adalah kumpulan halaman yang menampilkan informasi dalam bentuk teks, gambar (diam atau bergerak), animasi, suara, atau kombinasi semuanya. Halaman-halaman ini bisa bersifat statis atau dinamis dan membentuk satu rangkaian yang saling terkait. Setiap halaman dihubungkan melalui jaringan halaman lain. Hubungan antar halaman web disebut "Hyperlink," sedangkan teks yang menjadi media penghubung disebut "Hypertext." Untuk membangun sebuah website, ada beberapa hal yang perlu dipersiapkan, yaitu Nama Domain (Domain name/URL—Uniform Resource Locator), Rumah Website (Website Hosting), dan Content Management System (CMS) [10].

Pengelompokan jenis web umumnya didasarkan pada fungsi, sifat atau gaya, dan bahasa pemrograman yang digunakan. Berikut adalah jenis-jenis web berdasarkan sifat atau gayanya:

1. Website Dinamis: Jenis website ini menyajikan konten yang selalu berubah-ubah. Bahasa pemrograman yang digunakan antara lain PHP, ASP, .NET, dan memanfaatkan database seperti MySQL atau MS SQL. Contoh website dinamis: www.artikel-it.com, www.detik.com, www.technomobile.co.cc, www.polinpdg.ac.id, dan lain-lain.

2. Website Statis: Website jenis ini memiliki konten yang jarang diubah. Bahasa pemrograman yang digunakan adalah HTML dan belum memanfaatkan database. Contoh website statis antara lain web profil organisasi [10].

**2.2.3 Praktek Kerja Lapangan**

Praktek Kerja Lapangan (PKL) adalah metode untuk mengintegrasikan program akademik dengan penguasaan keahlian secara sistematis dan sinkron. Keahlian ini diperoleh melalui pengalaman kerja langsung di tempat kerja untuk mencapai tingkat kompetensi tertentu. Tujuan pelaksanaan PKL meliputi: menambah pengetahuan dan wawasan keilmuan mahasiswa melalui pengalaman langsung di dunia kerja, mengaplikasikan keterampilan dan keahlian sesuai bidang ilmu yang dipelajari secara langsung sehingga mahasiswa dapat memahami ruang lingkup bidang kerja sesuai kompetensinya, membentuk sikap dan mental mahasiswa agar mampu dan berani menghadapi tantangan dunia kerja yang penuh persaingan. Selain itu, PKL juga memiliki berbagai manfaat lainnya [3].

**2.2.4 Next.js**

Next.js adalah kerangka kerja React ringan yang digunakan untuk mengembangkan aplikasi statis dan dirender oleh server. Kerangka kerja ini menggunakan struktur direktori sebagai metode perutean untuk halaman web, dengan "app" sebagai halaman default. Melalui penggunaan direktori halaman, Next.js menyediakan perutean otomatis, sementara sisi server menangani pengambilan data untuk setiap permintaan [11]. Next.js, yang dikembangkan oleh Vercel, dirancang untuk menyederhanakan pembuatan aplikasi web yang dirender di server, dibuat secara statis, dan hibrida. Fitur-fitur utamanya meliputi pemisahan kode otomatis, render sisi server, pembuatan situs statis, dan router API bawaan. Secara kolektif, fitur-fitur ini meningkatkan kinerja dan SEO, memungkinkan pengembang untuk membuat aplikasi web yang efisien dan dapat diskalakan. Kerangka kerja ini mendukung baik rendering sisi server maupun pembuatan situs statis, memberikan fleksibilitas kepada pengembang untuk melakukan pra-rendering halaman baik pada waktu pembuatan atau waktu permintaan. Fleksibilitas ini menjadikan Next.js cocok untuk berbagai kasus penggunaan, mulai dari aplikasi dinamis hingga situs web statis [12].

**2.2.5 White Box Testing**

Menurut Verma dkk. (2017), White Box Testing adalah teknik pengujian perangkat lunak yang berfokus pada struktur internal aplikasi. Metode ini menyelidiki logika internal, struktur kode, dan alur kontrol aplikasi. Teknik ini juga dikenal dengan nama lain seperti Clear Box Testing, Open Box Testing, Glass Box Testing, Transparent Box Testing, Code-Based Testing, atau Structural Testing. Dalam white box testing, penguji harus memiliki pengetahuan mendalam tentang bahasa pemrograman (source code) karena berkaitan dengan kerja internal aplikasi. Metode ini memerlukan informasi yang detail tentang aplikasi yang diuji.

White Box Testing adalah metode pengujian yang diterapkan langsung pada kode sumber perangkat lunak, memeriksa semua jalur kode. Untuk melakukan teknik pengujian ini, penguji harus memiliki kemampuan pemrograman yang kuat. White Box Testing membutuhkan sumber daya manusia yang sangat terampil dan memiliki pengetahuan implementasi yang baik. White Box Testing terdiri dari beberapa teknik:

1. Control Flow Testing: Teknik pengujian struktural ini memeriksa aliran kontrol program, termasuk jalur sederhana dan rumit. Pengujian ini diterapkan pada semua perangkat lunak untuk menguji aliran kontrol. Ini adalah teknik mendasar yang berlaku pada program kecil dan subbagian dari program besar.
2. Path Testing/Basic Path Testing: Pengujian jalur ini menguji semua jalur program yang mungkin. Ini adalah teknik yang komprehensif, memastikan bahwa semua jalur dilalui setidaknya satu kali. Teknik cakupan jalur lebih baik daripada cakupan cabang dan cocok untuk menguji program yang kompleks.
3. Data Flow Testing: Pengujian aliran data berfokus pada titik-titik di mana variabel menerima nilai dan di mana nilai tersebut digunakan. Teknik ini dapat mengidentifikasi beberapa masalah seperti:
4. Variabel yang dideklarasikan tetapi tidak pernah digunakan dalam program.
5. Variabel yang digunakan dalam program tetapi tidak dideklarasikan.
6. Deklarasi ganda variabel sebelum digunakan [6].

**2.2.6 Flowchart**

Flowchart atau diagram alir adalah alat visual yang menggunakan simbol-simbol untuk mewakili komponen fisik dan/atau logika dari sistem informasi. Secara umum, diagram alir didefinisikan oleh standar ANSI/ISO sebagai representasi grafis dari definisi, analisis, atau metode penyelesaian masalah, di mana simbol digunakan untuk mewakili operasi, data, aliran, peralatan, dan lain-lain [13]. Berikut merupakan simbol-simbol flowchart yang umum digunakan [14]:

|  |  |
| --- | --- |
| **Simbol** | **Keterangan** |
|  | Terminal: Menunjukkan awal atau akhir dari diagram alir. |
|  | Input/Output: Menunjukkan operasi input atau output. |
|  | Process: Mewakili tahapan/langkah dalam suatu proses. |
|  | Predefined Process: Menunjukkan proses yang didefinisikan di tempat lain. |
|  | Comment: Menunjukkan informasi tambahan tentang tahapan dalam suatu proses. |
|  | Flow Line: Menunjukkan arah proses. |
|  | Document Input/Output: Menunjukkan operasi input dokumen atau output dokumen. |
|  | Decision: Mewakili proses penentuan yang mempengaruhi langkah berikutnya dalam suatu proses. |
|  | On-page Connector: Digunakan untuk menghubungkan bagian flowchart yang berada di halaman yang sama. |
|  | Off-page Connector: Digunakan untuk menghubungkan bagian flowchart yang berada di halaman lain. |

Tabel 2.1 simbol-simbol flowchart

**2.2.7 Flowgraph**

Flowgraph menggambarkan jalannya logika dalam program. Flowgraph dapat dibuat dari kode program atau diagram alir sistem. Flowgraph menggunakan simbol lingkaran (node) dan panah (edge). Node mewakili pernyataan prosedural, sedangkan edge mewakili alur logika program [15].

**2.2.8 Cyclomatic Complexion**

Cyclomatic Complexity adalah pengukuran kuantitatif untuk mengetahui kompleksitas logika dalam program. Dalam pengujian whitebox menggunakan metode Basis Path, nilai Cyclomatic Complexity menentukan jumlah jalur independen dalam basis set program. Jumlah jalur independen ini menentukan jumlah kasus uji minimal yang harus dilakukan untuk memastikan semua persyaratan pada jalur independen telah dieksekusi setidaknya satu kali [15]. Rumus untuk menghitung Cyclomatic Complexity adalah sebagai berikut:

V(G) = E - N + 2 atau V (G) = P + 1

dimana:

V(G) = Cyclomatic Complexity

E = jumlah edge pada flowgraph

N = jumlah node pada flowgraph

P = jumlah predicate node pada flowgraph

**2.2.9 Unified Modelling Language**

Menurut Jacobson dkk. (2021), Unified Modeling Language (UML) adalah bahasa pemodelan visual yang digunakan secara umum untuk menentukan, menvisualisasikan, membangun, dan mendokumentasikan komponen sistem perangkat lunak. Bahasa ini digunakan untuk memahami, merancang, melacak, mengonfigurasi, memelihara, dan mengendalikan informasi tentang sistem tersebut. UML dimaksudkan untuk digunakan dalam semua metode pengembangan, tahapan siklus hidup, domain aplikasi, dan media. Bahasa pemodelan ini bertujuan untuk menyatukan pengalaman masa lalu tentang teknik pemodelan dan menggabungkan praktek terbaik perangkat lunak saat ini ke dalam pendekatan standar. UML mencakup konsep semantik, notasi, dan pedoman. Ia memiliki bagian statis, dinamis, lingkungan, dan organisasi.

UML menggunakan informasi tentang struktur statis dan perilaku dinamis dari sebuah sistem. Sistem dimodelkan sebagai kumpulan objek yang saling berinteraksi untuk memberikan manfaat bagi pengguna. Struktur statis menjelaskan jenis-jenis objek penting dan hubungan di antara mereka. Perilaku dinamis menjelaskan sejarah objek dan komunikasi di antara mereka untuk mencapai tujuan. Memodelkan sistem dari beberapa sudut pandang yang saling terkait memungkinkan pemahaman yang lebih baik untuk berbagai tujuan [16].

**2.2.10 Use Case Diagram**

Use Case Diagram menangkap perilaku sistem, subsistem, atau kelas dari sudut pandang pengguna eksternal. Use Case Diagram termasuk ke dalam Unified Modelling Language (UML). Diagram ini membagi fungsionalitas sistem menjadi transaksi yang bermakna bagi aktor/pengguna ideal sebuah sistem. Bagian-bagian fungsionalitas interaktif ini disebut use case. Setiap use case menggambarkan interaksi dengan aktor sebagai rangkaian pesan antara sistem dan satu atau lebih aktor. Istilah "aktor" mencakup manusia, sistem lain, dan proses komputer [16].

**2.2.11 Activity Diagram**

Activity Graph adalah bentuk khusus dari mesin status yang digunakan untuk memodelkan komputasi dan alur kerja. Aktivitas-aktivitas dalam graf ini mewakili status-status pelaksanaan komputasi, bukan status-status objek biasa. Activity Graph mengasumsikan komputasi berjalan tanpa interupsi eksternal. Sebuah Activity Graph berisi status aktivitas yang mewakili eksekusi pernyataan dalam prosedur atau kinerja aktivitas dalam alur kerja. Transisi penyelesaian dijalankan ketika aktivitas sebelumnya selesai. Activity Diagram merupakan diagram yang menggambarkan Activity Graph. Activity Diagram termasuk ke dalam Unified Modelling Language (UML). Activity Diagram dapat mencakup cabang, serta percabangan kontrol ke dalam Concurrent Thread yang mewakili aktivitas yang dapat dilakukan secara bersamaan oleh objek atau orang yang berbeda. Activity Diagram memungkinkan kontrol konkuren selain kontrol sekuensial [16].

**2.2.12 System Usability Scale**

System Usability Scale (SUS) adalah alat penilaian kegunaan yang dikembangkan oleh John Brooke untuk memungkinkan praktisi menilai kegunaan produk atau layanan dengan cepat dan mudah. Instrumen SUS terdiri dari 10 pernyataan yang dinilai berdasarkan skala 5 poin tingkat persetujuan. Skor akhir SUS berkisar antara 0 hingga 100, dengan skor lebih tinggi menunjukkan kegunaan yang lebih baik. Karena pernyataan-pernyataannya bergantian antara positif dan negatif, diperlukan kehati-hatian dalam menilai survei. Meskipun ada alternatif lain, SUS memiliki beberapa kelebihan yang membuatnya menjadi pilihan yang baik, yaitu tidak bergantung pada jenis teknologi tertentu sehingga dapat digunakan untuk menilai berbagai macam antarmuka komputer, relatif cepat dan mudah digunakan, memberikan skor tunggal yang mudah dipahami oleh berbagai pihak, serta tidak memiliki hak milik sehingga hemat biaya [8].

**Daftar Pustaka**

[1] A. G. Gani, “SEJARAH dan PERKEMBANGAN INTERNET DI INDONESIA.”

[2] E. Mufida, E. Rahmawati, dan H. Hertiana, “Rancang Bangun Sistem Informasi Inventory pada Salon Kecantikan,” *Jurnal Mantik Penusa*, vol. 3, no. 3, hlm. 99–102, Des 2019.

[3] M. Arifin, “Analisa dan Perancangan Sistem Informasi Praktek Kerja Lapangan pada Instansi/Perusahaan,” vol. 5, Apr 2014.

[4] S. T. Safitri dan D. Supriyadi, “Rancang Bangun Sistem Informasi Praktek Kerja Lapangan Berbasis Web dengan Metode Waterfall,” *JURNAL INFOTEL - Informatika Telekomunikasi Elektronika*, vol. 7, no. 1, hlm. 69, Mei 2015, doi: 10.20895/infotel.v7i1.32.

[5] C. T. Pratala, E. M. Asyer, I. Prayudi, dan A. Saifudin, “Pengujian White Box pada Aplikasi Cash Flow Berbasis Android Menggunakan Teknik Basis Path,” *Jurnal Informatika Universitas Pamulang*, vol. 5, no. 2, hlm. 111, Jun 2020, doi: 10.32493/informatika.v5i2.4713.

[6] A. Verma, A. Khatana, dan S. Chaudhary, “A Comparative Study of Black Box Testing and White Box Testing,” *International Journal of Computer Sciences and Engineering*, vol. 5, no. 12, hlm. 301–304, Des 2017, doi: 10.26438/ijcse/v5i12.301304.

[7] A. Sidik, S. Sn, M. Ds, U. Islam, K. Muhammad, dan A. Al-Banjari, “Penggunaan System Usability Scale (SUS) Sebagai Evaluasi Website Berita Mobile.” [Daring]. Tersedia pada: http://m.detik.com

[8] A. Bangor, P. T. Kortum, dan J. T. Miller, “An empirical evaluation of the system usability scale,” *Int J Hum Comput Interact*, vol. 24, no. 6, hlm. 574–594, Agu 2008, doi: 10.1080/10447310802205776.

[9] T. Sutabri, *Konsep sistem informasi*. Penerbit Andi, 2012.

[10] R. Hidayat, *Cara praktis membangun website gratis*. Elex Media Komputindo, 2010.

[11] H. A Jartarghar, G. Rao Salanke, A. K. A.R, S. G.S, dan S. Dalali, “React Apps with Server-Side Rendering: Next.js,” *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, vol. 14, no. 4, hlm. 25–29, Des 2022, doi: 10.54554/jtec.2022.14.04.005.

[12] E. Kroon Celander dan A. Möllestål, “A Comparative Analysis of Next.js, SvelteKit, and Astro for E-commerce Web Development,” 2024.

[13] K. B. Lloyd dan J. Solak, “Flowcharting Techniques,” dalam *Encyclopedia of Information Systems*, Elsevier, 2003, hlm. 331–344. doi: 10.1016/B0-12-227240-4/00071-X.

[14] A. B. Chaudhuri, *Flowchart and algorithm basics: The art of programming*. Mercury Learning and Information, 2020.

[15] C. P. C. Munaiseche dan G. C. Rorimpandey, “Penerapan Metode Basis Path Analysis dalam Pengujian White Box Sistem Pakar,” *Prosiding SISFOTEK*, vol. 5, no. 1, hlm. 124–128, 2021.

[16] L. Jacobson dan J. R. G. Booch, “The unified modeling language reference manual,” 2021.