

Rancang Bangun Website Penyedia Jasa Tenaga Ahli Menggunakan Metode Agile

1st Muhammad Arif
Fakultas Informatika

Telkom University Purwokerto
Purwokerto, Jawa Tengah, Indonesia

muhammadarif@student.telkomuniversity.ac.id

2nd Muhammad Lulu Latif Usman
Fakultas Informatika

Telkom University Purwokerto
Purwokerto, Jawa Tengah, Indonesia

muhlulu@telkomuniversity.ac.id

3rd Gunawan Wibisono
Fakultas Informatika

Telkom University Purwokerto
Purwokerto, Jawa Tengah, Indonesia

gunawanw@telkomuniversity.ac.id

Abstrak—Secara kodrati, manusia tidak dapat menjalani hidupnya secara individual tanpa menjalin hubungan sosial, termasuk dalam memenuhi kebutuhan akan tenaga kerja terampil. Pekerja lepas seperti tukang bangunan, montir, dan penyedia jasa lainnya memainkan peran vital dalam menunjang aktivitas ekonomi masyarakat. Di Kabupaten Banyumas, yang dihuni oleh 1.828.573 jiwa, terdapat 97.210 individu yang bekerja secara mandiri sebagai tenaga ahli. Jumlah ini mencerminkan potensi besar sekaligus tantangan dalam mengelola informasi dan layanan terkait. Fakta di lapangan menunjukkan masih adanya hambatan dalam menemukan penyedia jasa yang tepat karena tidak tersedianya sistem informasi yang terintegrasi dan mudah diakses. Minimnya keterhubungan antara masyarakat dan penyedia jasa menyebabkan terhambatnya distribusi pekerjaan yang seharusnya bisa diselesaikan secara efisien. Penelitian ini bertujuan untuk merancang dan membangun sistem informasi berbasis website guna menjembatani kebutuhan antara masyarakat dan tenaga ahli. Sistem ini mengusung pendekatan Agile development, yang memungkinkan pengembangan berjalan secara bertahap, responsif terhadap perubahan, dan melibatkan kolaborasi aktif dari pengguna. Pengujian fungsional dilakukan menggunakan metode black-box testing untuk memastikan bahwa fitur yang dikembangkan sesuai dengan tujuan dan kebutuhan pengguna. Hasil dari pengembangan ini diharapkan tidak hanya menjadi sarana pencarian jasa yang efisien, tetapi juga mendorong profesionalisme tenaga ahli lokal, memperkuat jaringan kerja, dan menghadirkan ekosistem layanan digital yang adaptif serta berkelanjutan.

Kata kunci: *Agile, Black-box testing, Manajemen pelayanan, Tenaga Kerja, Website*

I. PENDAHULUAN

Berdasarkan data dari Portal Satu Data Kabupaten Banyumas (oleh Dimas Satria), jumlah penduduk di wilayah tersebut pada tahun 2023 mencapai 1.828.573 jiwa, terdiri atas 919.879 laki-laki dan 908.694 perempuan[1]. Di sisi lain, merujuk pada hasil Survei Angkatan Kerja Nasional (Sakernas) edisi Agustus 2023, sebanyak 866.024 penduduk berusia 15 tahun ke atas tercatat aktif bekerja dalam sepekan terakhir. Dari jumlah tersebut, 111.698 individu bekerja secara mandiri

dalam kategori pekerja bebas maupun tenaga profesional tidak tetap. Data ini bersumber dari Badan Pusat Statistik Kabupaten Banyumas dan bagian dari inisiatif Satu Data Indonesia, sehingga dapat dianggap valid dan akurat untuk mendukung analisis serta perencanaan kebutuhan tenaga kerja.[2] Jumlah penduduk yang besar disertai banyaknya tenaga kerja lepas mencerminkan dinamika ekonomi daerah yang signifikan, permintaan akan jasa tenaga ahli di berbagai bidang diperkirakan akan terus meningkat, sehingga informasi mengenai pekerja lepas tenaga ahli menjadi semakin penting. Namun, karena keterbatasan mengenai informasi yang diberikan, masyarakat sering kesulitan dalam mencari layanan. Selain itu, pekerja jasa tenaga ahli juga memerlukan manajemen pelayanan yang baik untuk memberikan kepuasan optimal kepada pelanggan. Penyedia jasa tenaga ahli di Banyumas sebelumnya cenderung mengandalkan jaringan lokal atau rekomendasi untuk memperoleh pelanggan[3]. Tidak adanya platform digital menyebabkan keterbatasan dalam menjangkau calon pelanggan di luar lingkup komunitas mereka. Saat ini, meskipun teknologi informasi berkembang pesat, masih banyak penyedia jasa tenaga ahli yang belum memanfaatkan website atau platform digital untuk memperkenalkan layanan mereka[4]. Hal ini membuat pelanggan sering kesulitan dalam menemukan informasi yang akurat dan terpercaya mengenai layanan yang mereka butuhkan. Selain itu, pekerja lepas mengalami kesulitan dalam manajemen sistem pelayanan, seperti jadwal kerja, permintaan pelanggan, dan kepuasan layanan. Apabila isu ini belum ditindaklanjuti secara optimal, dikhawatirkan akan menimbulkan dampak yang lebih kompleks di kemudian hari. Semakin banyak pelanggan yang kesulitan menemukan penyedia jasa tenaga ahli yang mereka butuhkan. Selain itu, penyedia jasa juga akan kesulitan berkembang karena tidak memiliki media yang efektif untuk mempromosikan diri dan mengelola manajemen pelayanan[5]. Oleh karena itu, dibutuhkan sebuah sistem berbasis teknologi informasi, seperti website, yang dapat menjembatani komunikasi antara penyedia jasa tenaga ahli dengan pelanggan serta membantu pengelolaan pelayanan secara lebih efisien[6]. Teknologi informasi memberikan peran

penting dalam perkembangan komputer, salah satu perkembangan teknologi informasi yaitu website[7]. Website merupakan sebuah media Informasi yang saling berelasi dan dapat diakses melalui internet sebagai tempat untuk menjembatani antara pekerja jasa tenaga keahlian dengan pelanggan. Metode yang digunakan dalam pembuatan website memiliki peran penting dalam keberhasilannya, beragam metode dalam perancangan sistem antara lain yaitu Waterfall, Prototype, dan Agile[8]. Metode yang paling sering digunakan dalam perancangan sistem yaitu metode Waterfall merupakan pendekatan pengembangan perangkat lunak yang bersifat sistematis dan linier, di mana setiap tahapan harus diselesaikan terlebih dahulu sebelum melanjutkan ke tahap berikutnya. Pendekatan ini dianalogikan seperti aliran air terjun, yang mengalir secara bertahap dari satu tingkat ke tingkat berikutnya tanpa kembali ke tahap sebelumnya. satu arah yang dimulai dari perencanaan lalu tahap-tahap selanjutnya, salah satu kekurangannya adalah waktu yang diperlukan untuk mengembangkan sistem cukup lama, sehingga biaya yang dibutuhkan juga meningkat[8]. Metode prototyping, metode prototyping adalah teknik pengumpulan informasi dari pengguna dengan cepat dan berfokus pada aspek tampilan perangkat yang akan dilihat oleh pelanggan atau pengguna yang akan mengevaluasi untuk mengidentifikasi dan menyempurnakan kebutuhan pengembangan perangkat lunak[9]. Metode ini memiliki kekurangan, salah satunya adalah biaya dan sumber daya yang tinggi, sehingga kurang relevan untuk digunakan pada penelitian dengan anggaran terbatas. Metode Agile, Agile adalah salah satu metodologi pengembangan perangkat lunak yang efisien dan fleksibel. Dalam merancang sistem untuk penelitian ini, peneliti menggunakan metode Agile sebagai dasar metodologi[10]. Metode Agile ini diterapkan karena penelitian ini memiliki siklus perancangan yang singkat. Meskipun demikian, saat pengembangan akan ada tahap-tahap selanjutnya untuk mengembangkannya berdasarkan kebutuhan pengguna. Berdasarkan uraian yang telah disampaikan, maka diusulkan penelitian dengan judul “RANCANG BANGUN WEBSITE PENYEDIA JASA TENAGA AHLI MENGGUNAKAN METODE AGILE”.Diharapkan penelitian ini dapat memberikan solusi dan inovasi positif terhadap permasalahan yang ada.

II. KAJIAN TEORI

A. Rancang bangun

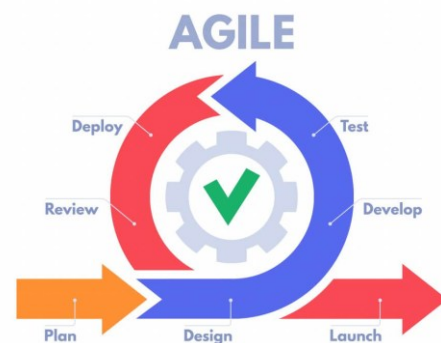
Rancang bangun adalah proses yang menentukan aktivitas informasi dan tugas-tugas khusus yang dibutuhkan untuk pengembangan

teknologi. Proses ini mencakup beberapa tahap penting, yaitu analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan produk teknologi tersebut. Dengan kata lain, rancang bangun melibatkan semua langkah yang diperlukan untuk mengidentifikasi kebutuhan, merancang solusi yang tepat, mengembangkan dan menguji teknologi, serta memastikan teknologi tersebut berfungsi dengan baik dan terus diperbarui sesuai kebutuhan.Website[11].

B. Pekerja Tenaga Ahli

Pekerja Tenaga Ahli atau Jasa tenaga keahlian adalah layanan pekerja dengan keahlian khusus untuk menyelesaikan suatu pekerjaan. Pekerja ini dibagi menjadi dua kategori, yaitu berdasarkan hasil dan berdasarkan waktu[4]. Pekerja tenaga keahlian berdasarkan hasil adalah mereka yang menawarkan keahlian tertentu dan menyelesaikan proyek yang nantinya akan diberikan imbalan yang ditentukan berdasarkan hasil kerja mereka[12].

C. Agile



GAMBAR 1
(METODE AGI)

Agile merupakan pendekatan dalam pengembangan perangkat lunak yang menitikberatkan pada pencapaian hasil secara bertahap melalui kolaborasi intensif, evaluasi berkelanjutan, dan penyesuaian yang adaptif. Alih-alih mengandalkan perencanaan jangka panjang yang kaku, metode ini menekankan fleksibilitas dan respons cepat terhadap perubahan kebutuhan. Prosesnya berpusat pada pembuatan produk minimum yang layak (MVP), yang secara konsisten disempurnakan melalui umpan balik langsung dari pengguna maupun tim pengembang. Dengan mendorong keterlibatan aktif dari seluruh anggota tim dan siklus iteratif yang cepat, Agile menciptakan ekosistem kerja yang dinamis dan efisien untuk mencapai tujuan proyek secara berkesinambungan.

1. Planing

Pada tahap awal planning (perencanaan) ini, dilakukan pengumpulan data dan penerimaan usulan

kebutuhan dari berbagai aktivitas sistem. Tujuannya adalah untuk mendapatkan gambaran yang jelas mengenai fitur utama, fungsi, dan anggaran yang diinginkan [13].

2. Design

Design merupakan bagian dari proses yang dilakukan oleh peneliti untuk menentukan citra tampilan dalam perancangan agar aplikasi terlihat lebih berwarna dan berkarakter. Pada tahap ini, yang terpenting adalah pemilihan warna dan karakter pada website [13].

3. Develop and Testing

Pada tahap pengembangan (develop) ini mengarah pada serangkaian aktivitas yang dilakukan untuk menerjemahkan desain menjadi produk atau sistem yang berfungsi. Ini melibatkan implementasi konsep-konsep desain ke dalam kode atau bentuk yang dapat dijalankan dan digunakan. Pada tahap testing (pengujian) dilakukan dengan metode pengujian *Blackbox* testing untuk memastikan produk atau sistem yang dikembangkan berfungsi dengan spesifikasi yang telah ditetapkan dan standar kualitas yang diinginkan [13].

4. Deploy and Review

Pada tahap deploy dan review ini, produk atau sistem yang sudah melalui tahap pengembangan dan pengujian sudah siap untuk dirilis atau diterapkan secara luas. tahap ini melibatkan proses pengimplementasian produk atau sistem ke lingkungan produksi atau pengguna, dan evaluasi terakhir untuk peluncuran secara resmi [13].

D. Html (*Hyper text Markup Language*)

Hypertext Markup Language (HTML) adalah bahasa standar untuk membuat tampilan halaman web. HTML digunakan untuk menyusun dan menata tampilan dengan membuat heading, daftar, tabel, dan elemen lainnya. Penulisan HTML melibatkan penggunaan tag, seperti font, untuk mendefinisikan struktur dan format konten halaman web. UML (Unified Modeling Language) – Parafrasa[14].

E. Css (*Cascading Style Sheets*)

Cascading Style Sheets (CSS) digunakan untuk mengatur tampilan web. CSS diterapkan dalam HTML untuk mengubah tampilan seperti warna Background, font, dan elemen lainnya. Meskipun CSS merupakan file terpisah, CSS tetap terhubung dengan HTML karena memberikan warna dan karakter pada halaman web, sehingga membuatnya terlihat lebih menarik [15].

F. Tailwind css

Tailwind CSS merupakan framework CSS berbasis utility-first yang memudahkan pengembangan antarmuka web yang responsif dan modern[7]. Framework ini memungkinkan pengembang untuk memanfaatkan kelas-kelas utility secara langsung dalam markup HTML, sehingga menghasilkan desain yang konsisten, fleksibel, dan mudah dikustomisasi[3]. Dengan pendekatan tersebut, Tailwind CSS mendukung pengembangan UI/UX yang lebih cepat dan efisien dibandingkan dengan framework berbasis komponen seperti Bootstrap[16].

G. MySql

Dengan perkembangannya saat ini, MySQL dapat dikenal sebagai SQL, Singkatan dari SQL itu sendiri Structured Query Language yang merupakan bahasa khusus untuk mengelola sistem basis data MySQL adalah sistem manajemen basis data relasional yang dikenal menggunakan bahasa sederhana untuk mengakses dan mengelola basis data. Pendekatan ini membuat MySQL lebih mudah dipahami dan digunakan oleh pengembang perangkat lunak, keunggulan lain dari MySQL adalah bersifat opensource. artinya komunitas pengembang dapat dengan bebas mengakses dan memodifikasi kode sumber. Keterbukaan ini memudahkan penggunaan MySQL untuk berbagai keperluan, mulai dari proyek kecil hingga sistem besar[17].

H. Mysql workbeans

MySQL Workbench MySQL Workbench adalah perangkat lunak grafis yang dirancang untuk membantu pengguna dalam merancang skema database, memodelkan struktur data, mengelola koneksi, serta memelihara sistem basis data MySQL secara lebih efisien. Aplikasi ini mendukung berbagai fitur penting seperti perancangan visual database (ERD), eksekusi query SQL, administrasi server, hingga migrasi data dari sistem lain. berbagai sistem operasi seperti Windows, Linux, dan macOS. Dengan kemampuannya sebagai alat visualisasi basis data, MySQL Workbench memudahkan pengguna dalam membuat model basis data (ERD), menjalankan query SQL, serta mengelola koneksi ke server MySQL. Selain itu, MySQL Workbench menyediakan fitur yang lengkap untuk perancangan skema database, pengujian, dan pemeliharaan basis data, sehingga sangat membantu pengembang dalam membangun dan mengelola aplikasi yang berbasis data.[17]

I. Prisma

Prisma adalah ORM modern untuk Node.js dan TypeScript yang mempermudah pengelolaan

database seperti PostgreSQL, MySQL, SQLite, SQL Server, dan MongoDB. Prisma memiliki tiga komponen utama, yaitu Prisma Client untuk query database, Prisma Migrate untuk migrasi skema, dan Prisma Studio sebagai GUI pengelola data. File `schema.prisma` adalah konfigurasi utama yang memuat data model (tabel dan field), relasi antar tabel, konfigurasi database, dan pengaturan generator Prisma Client. Struktur file ini mencakup generator (menentukan engine), datasource (koneksi database), model (representasi tabel), field (kolom tabel dengan tipe data), atribut field (seperti `@id`, `@default`, `@unique`), dan relasi antar tabel (`@relation`)[18]. Prisma mendukung relasi one-to-one, one-to-many, dan many-to-many dengan sintaks yang sederhana dan jelas. Kelebihannya meliputi query database yang mudah tanpa SQL mentah, migrasi skema otomatis, dukungan TypeScript dengan tipe data otomatis, dan pengelolaan relasi yang praktis[19].

J. Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah sebuah standar pemodelan visual yang dirancang untuk mempermudah proses perancangan sistem berbasis objek. Dengan menyediakan notasi grafis yang seragam, UML memungkinkan para pengembang untuk menyampaikan ide dan struktur sistem secara lebih jelas dan terstruktur kepada berbagai pihak yang terlibat. UML sendiri merupakan hasil penyatuan dari tiga pendekatan pemodelan terkemuka, yaitu metode Booch, Object Modeling Technique (OMT), dan Object-Oriented Software Engineering (OOSE). Metode Booch, yang dikembangkan oleh Grady Booch, menekankan pendekatan desain berorientasi objek dengan empat siklus utama: pengenalan kelas dan objek, penjabaran makna hubungan antar elemen, serta pendalaman ke dalam aspek detail sistem secara bertahap dan berulang[21].

1. Use Case Diagram

Use Case Diagram adalah representasi fungsional dari sebuah sistem yang disusun berdasarkan perspektif pengguna. Diagram ini menggambarkan bagaimana aktor (user atau entitas lain) berinteraksi dengan sistem melalui berbagai skenario penggunaan. Setiap skenario merepresentasikan rangkaian langkah atau kejadian yang menunjukkan bagaimana suatu tujuan tertentu dapat dicapai melalui sistem, mulai dari aksi pengguna hingga respons sistem yang dihasilkan. antara aktor dan sistem yang terjadi dalam suatu konteks tertentu. Skenario ini disusun secara sistematis untuk meminimalkan ketergantungan pada individu, perangkat keras, maupun urutan waktu, sehingga dapat menggambarkan fungsi sistem secara umum dan konsisten[22].

Masalah umum dalam pembuatan *Use Case* diagram adalah terlalu memfokuskan pada interaksi antara pengguna (*user*) dan sistem tanpa mempertimbangkan konteks proses bisnis secara menyeluruh. Sering kali, perubahan pada proses bisnis justru menjadi solusi terbaik untuk menyelesaikan permasalahan sistem. Oleh karena itu, penting untuk membedakan antara *business Use Case* dan *system use case*. Pada Gambar 2.2, digunakan pendekatan level dalam *Use Case* dengan menggunakan istilah *sea level* dan *fish level*. Pendekatan ini bertujuan untuk memberikan perspektif yang berbeda dalam memahami skenario sistem, mulai dari gambaran umum hingga detail interaksi [6]. Model ini nantinya akan digunakan oleh penulis sebagai dasar dalam merancang *Use Case* diagram yang sesuai dengan kebutuhan sistem.

2. Activity Diagram

Activity Diagram merupakan salah satu bagian penting dari *Unified Modeling Language (UML)* yang digunakan untuk menggambarkan aspek dinamis dari suatu sistem. Diagram ini menampilkan logika prosedural dari suatu proses bisnis atau alur kerja (*workflow*) yang terjadi di dalam sistem. Tujuan utama dari pembuatan *Activity Diagram* adalah untuk menangkap perilaku dinamis dari sistem dengan cara menunjukkan aliran aktivitas atau aliran kendali dari satu aktivitas ke aktivitas lainnya. Dengan demikian, diagram ini sangat berguna dalam memodelkan urutan kegiatan, proses paralel, serta keputusan yang terjadi dalam suatu sistem perangkat lunak. Adapun Simbologi yang sering digunakan pada saat pembuatan *activity diagram*[8].

3. Class Diagram

Class merupakan kumpulan objek yang memiliki atribut dan operasi (*operation*) yang sama. Diagram ini menghubungkan struktur yang di dalamnya mencakup atribut data dari masing-masing kelas serta operasi atau fungsi yang dapat dilakukan oleh kelas tersebut. *Class* diagram digunakan untuk menggambarkan struktur statis dari sistem dengan menampilkan kelas-kelas yang ada, atribut-atributnya, metode atau operasinya, serta hubungan antar kelas tersebut. Dalam penyusunannya, digunakan simbologi tertentu yang mengacu pada referensi dari literatur buku dan sumber daring sebagai standar dalam pemodelan *Class*[22].

K. Blackbox

Blackbox merupakan metode pengujian aplikasi yang berfokus spesifikasi tanpa harus memperhatikan detail dari aplikasi yang diuji, *Blackbox* testing berfungsi untuk memeriksa hasil input dan output yang dilakukan di akhir

perancangan sistem. Tujuan dalam melakukan pengujian adalah apakah perangkat lunak yang dirancang dapat berfungsi dengan baik atau tidak. Untuk perhitungan persentase *blackbox* digunakan menghitung tingkat keberhasilan selama pengujian fungsionalitas aplikasi. Proses perhitungan persentase ini dapat dilakukan menggunakan rumus berikut[23].

III. METODE

Metode penelitian ini mengandalkan pendekatan *blackbox* testing sebagai strategi validasi fungsionalitas aplikasi, yang didahului oleh serangkaian tahap awal seperti perancangan skenario pengujian, penentuan parameter input, serta simulasi perilaku sistem tanpa mengakses struktur internalnya.[23].

a. Identifikasi Masalah

Untuk langkah awal penilitan ini adalah identifikasi bagaimana penyedia jasa menawarkan jasa nya sebelum ada ide platform web penyedia jasa online yang dimana masih cukup kuni *dor to dor* dalam menawarkan layanan. Penelitian ini bertujuan untuk merancang sebuah aplikasi dengan metode agile dengan melakukan sesi wawancara dan reservasi dengan penyedia jasa layanan langsung untuk mengumpulkan data dan informasi yang nantinya informasi tersebut dapat membantu peneliti untuk merancang aplikasi[24]

b. Pengumpulan Data

Proses pengumpulan data bertujuan untuk mendapatkan berbagai data, teknik pengumpulan data di penelitian ini ada tiga antara lain

1. Studi Literatur

Studi literatur dilakukan untuk mengumpulkan data dan informasi yang relevan dengan topik penelitian. Informasi tersebut diperoleh dari berbagai sumber seperti buku, tesis, makalah penelitian, artikel ilmiah, internet, serta referensi terpercaya lainnya.[24]

2. Wawancara

Pada tahap wawancara, peneliti menggunakan pedoman wawancara terstruktur untuk memperoleh informasi yang dibutuhkan. Wawancara ini dilakukan bersama

Bapak Tarno, salah satu penyedia jasa di bidang layanan serba bisa, guna menggali data secara langsung dari sumber yang relevan.

c. Merancang Aplikasi

Setelah data terkumpul, langkah selanjutnya adalah merancang aplikasi yang telah diidentifikasi, yaitu sebuah platform yang ditujukan untuk melayani para penyedia jasa tenaga ahli[24].

1. Planing

Pada tahap ini, saya membuat perencanaan awal untuk menggambarkan interaksi utama antara aktor dan sistem pada platform penyedia jasa. Perencanaan ini mencakup fitur-fitur dasar yang akan dibangun, seperti proses registrasi dan login bagi pengguna. Setelah berhasil login, pengguna dapat memilih untuk menjadi pencari jasa atau penyedia jasa (*worker*)[13]. Pencari jasa memiliki akses untuk mencari layanan dan melakukan pemesanan, sementara penyedia jasa dapat menawarkan layanan kepada pengguna lain. Selain itu, terdapat peran admin yang bertugas memantau aktivitas pengguna serta mengelola akun dengan menambah atau menghapus *user* [8]. Perencanaan ini menjadi fondasi untuk pengembangan sistem lebih lanjut, termasuk dalam penentuan alur kerja aplikasi, kebutuhan teknis, dan struktur basis data yang diperlukan.

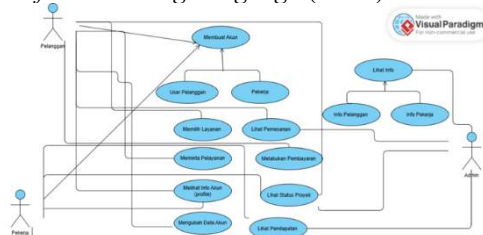
2. Design

Selanjutnya, desain low fidelity merupakan gambaran kasar dari tampilan antarmuka aplikasi ini yang bertujuan untuk memvisualisasikan alur navigasi dan susunan komponen utama tanpa menampilkan detail visual secara lengkap [25]. Desain ini digunakan sebagai acuan awal dalam pengembangan sebelum masuk ke tahap desain yang lebih rinci.



GAMBAR 2
(LOW FIDELITY)

d. *Unified Modeling Language* (UML)



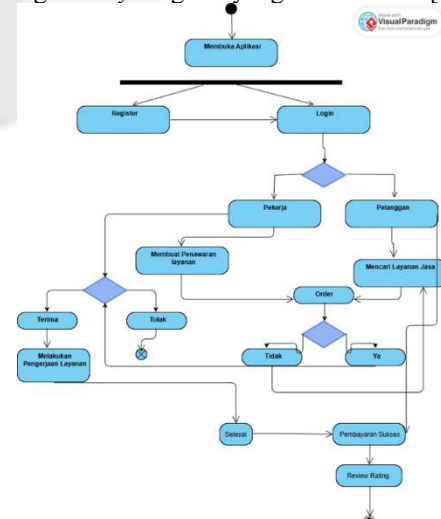
GAMBAR 3
(UML)

Use Case Diagram pada Gambar interaksi antara aktor dengan sistem, yang bertujuan untuk menunjukkan kebutuhan sistem sebagai dasar dalam perancangan perangkat lunak. Skenario ini membantu dalam membatasi serangkaian persyaratan yang dapat divalidasi saat proses pengembangan berlangsung.

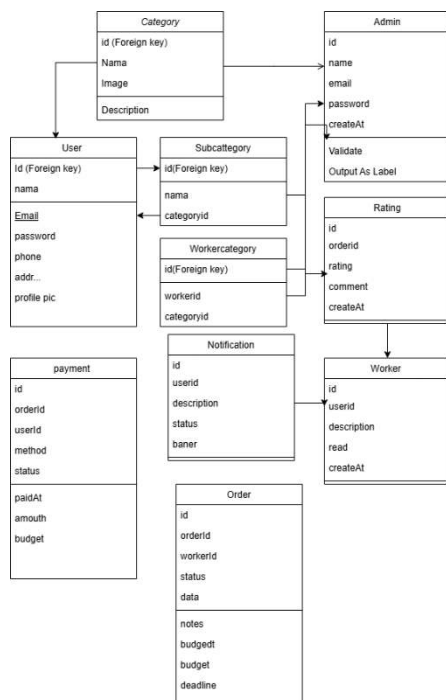
Melalui pendekatan berbasis cerita, *Use Case* memberikan gambaran bagaimana sistem digunakan, serta mengarahkan developer untuk memahami fungsionalitas dan fitur-fitur yang dibutuhkan oleh perangkat lunak.

e. *Activity* Diagram

Activity Diagram pada Gambar 2.4 di atas merupakan diagram aktivitas sederhana yang menggambarkan alur bagaimana pengguna berinteraksi dengan aplikasi TenangAja.com. Sesuai dengan dasar teori, simbologi dalam *activity diagram* digunakan untuk menunjukkan bagaimana aktivitas berlangsung secara berurutan. Melalui diagram ini, dapat terlihat gambaran umum proses kerja aplikasi dari awal hingga akhir, sehingga memudahkan dalam memahami alur sistem yang berjalan. Pada Gambar 3.7 terdapat sepuluh *activity diagram* yang dimana masing-masing merepresentasikan alur aktivitas dari sistem berdasarkan *Use Case* yang telah ditentukan sebelumnya. Diagram-diagram ini digunakan untuk menggambarkan proses bisnis serta logika alur kerja dari interaksi antara pengguna dan sistem secara lebih rinci. Setiap *activity diagram* menunjukkan urutan aktivitas yang menggambarkan bagaimana sistem merespons tindakan pengguna sesuai dengan skenario yang ada dalam *Use Case diagram*. Dengan demikian, *activity diagram* berfungsi sebagai visualisasi yang memperjelas dinamika perilaku sistem dalam menjalankan setiap fungsinya. Berikut adalah uraian dari masing-masing *activity diagram* yang telah disusun[6].



GAMBAR 4
(ACTIVITY DIAGRAM)

f. *Class Diagram*

GAMBAR 5
(CLASS DIAGRAM)

Menggambarkan struktur basis data yang digunakan dalam sistem, mencakup beberapa basis data yang berfungsi untuk menampung data seperti pekerja, pelanggan, dan entitas lainnya yang terlibat. Diagram berfungsi sebagai peta rinci dari sebuah konteks diagram yang lebih besar, memberikan pemahaman yang lebih mendalam terhadap bagian-bagian sistem secara spesifik.

IV. HASIL DAN PEMBAHASAN

A. Skenario Percobaan 1

Tahap skenario percobaan ini merupakan proses penyusunan alur uji coba yang dirancang berdasarkan kerangka rencana yang telah ditetapkan sebelumnya, sebagai panduan dalam pelaksanaan eksperimen yang terstruktur.

1. *Planing* (Skenario Percobaan 1)

Skenario ini adalah untuk membuat tampilan aplikasi yang mudah digunakan baik oleh tukang (penyedia jasa) maupun pelanggan (pencari jasa). Pada tahap perencanaan ini, dilakukan identifikasi awal terhadap kebutuhan sistem, peran pengguna, dan fitur utama yang akan dikembangkan. Perencanaan ini mencakup proses registrasi dan login bagi pengguna,

pembagian peran sebagai pencari jasa maupun penyedia jasa, serta fungsi-fungsi seperti mencari layanan, memesan jasa, menawarkan layanan, hingga pengelolaan akun oleh admin. Perencanaan ini bertujuan untuk memastikan bahwa jalannya interaksi antara pengguna dan sistem berlangsung secara optimal. Dengan demikian, hasil perencanaan tersebut menjadi acuan penting dalam proses perancangan antarmuka maupun pengembangan fungsi-fungsi sistem yang sesuai dengan kebutuhan pengguna. Desain antarmuka yang ramah pengguna dan efisien dalam mendukung proses transaksi jasa secara digital.

2. *Design* (Skenario Percobaan 1)

Pada skenario ini dilakukan proses perancangan tampilan antarmuka aplikasi yang mencakup pembuatan low fidelity design. *Low fidelity* design digunakan untuk menggambarkan alur dan struktur dasar aplikasi secara sederhana. Desain ini berfungsi sebagai acuan awal dalam proses pengembangan agar tampilan aplikasi sesuai dengan kebutuhan pengguna dan mudah digunakan baik oleh penyedia jasa maupun pencari jasa.

3. *Development* (Skenario Percobaan 1)

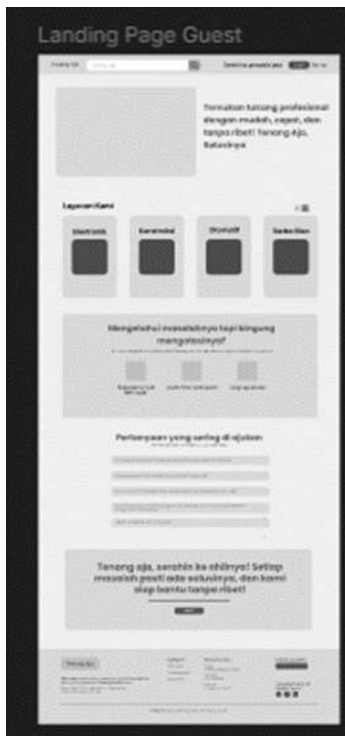
Pada skenario ini, dilakukan proses untuk pengembangan aplikasi menggunakan Next.js dan Tailwind CSS dengan tujuan merealisasikan tampilan antarmuka yang telah dirancang pada tahap sebelumnya, yaitu tahap desain *low fidelity*. Tahap perancangan awal ini mencakup pembuatan tampilan mulai dari halaman login, register, hingga dashboard.

B. Hasil Percobaan

1. Hasil Percobaan 1

Pada skenario ini dilakukan proses perencanaan, perancangan, hingga pengembangan awal aplikasi layanan digital yang ditujukan untuk memudahkan interaksi antara penyedia jasa (tukang) dan pencari jasa (pelanggan). Tahap perencanaan dimulai dengan mengidentifikasi kebutuhan sistem, peran pengguna, serta fitur-fitur utama seperti registrasi, login, pencarian layanan, pemesanan jasa, penawaran layanan, dan pengelolaan akun oleh admin.

Perencanaan ini bertujuan untuk menyusun alur sistem yang efisien dan *user-friendly*. Setelah tahap perencanaan, dilakukan perancangan tampilan antarmuka dalam bentuk *low fidelity design* untuk menggambarkan struktur dasar dan alur navigasi aplikasi secara sederhana.



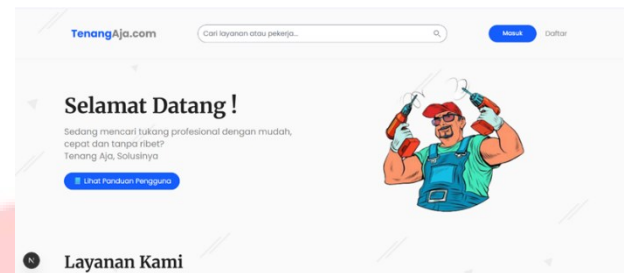
GAMBAR 6
(HASIL PERCOBAAN 1)

Ditampilkan hasil percobaan pertama berupa desain *low fidelity*. Desain ini berfungsi untuk memberikan gambaran awal secara kasar mengenai struktur dan alur aplikasi, di mana tampilannya masih sederhana dan belum mencapai bentuk akhir yang sempurna

2. Hasil Percobaan 2

Pada percobaan tahap 2 ini, dilakukan perencanaan lanjutan untuk menyusun alur yang dibutuhkan oleh sistem serta menjelaskan bagaimana sistem akan berjalan hingga ke tahap pengujian. Perencanaan ini bertujuan untuk memberikan gambaran menyeluruh mengenai alur kerja aplikasi, mulai dari proses awal hingga keterkaitan antar fitur, sehingga proses pengembangan dapat berjalan lebih terstruktur dan pengujian dilakukan secara lebih efektif. Selain itu, dilakukan pula perancangan tampilan antarmuka aplikasi yang mencakup pembuatan *low fidelity design* untuk

menggambarkan alur dasar dan struktur aplikasi secara sederhana sebagai acuan awal pengembangan. Selanjutnya, dibuat *high fidelity design* dengan tampilan visual yang lebih mendetail dan menyerupai bentuk akhir aplikasi, meliputi elemen UI seperti warna, ikon, dan tipografi agar tim pengembang dan stakeholder dapat memahami tampilan akhir aplikasi secara lebih nyata.

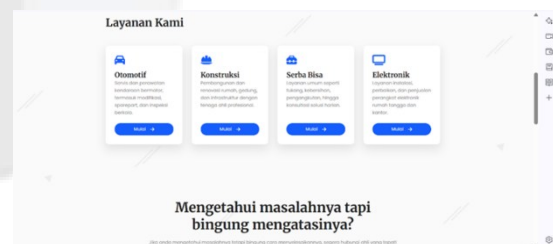


GAMBAR 7
(HASIL PERCOBAAN 2)



GAMBAR 8
(HASIL PERCOBAAN 2)

Pada Gambar 7 dan 8 ditunjukkan hasil percobaan kedua dari perancangan tahap 2, yang telah menambahkan fitur-fitur inti aplikasi seperti layanan, serta fitur tambahan yang di bungkus hamburger bar seperti pergantian antara pengguna (*user*) dan penyedia jasa (*worker*), termasuk tampilan profil pengguna (*user profile*).



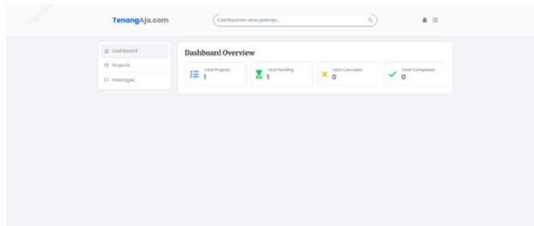
GAMBAR 9
(HASIL PERCOBAAN 2)

Pada Gambar 9 ditampilkan hasil percobaan kedua pada perancangan tahap 2, di mana fitur layanan sudah dapat diakses secara aktif. Selain itu, telah diterapkan logika dasar sistem yang menggambarkan bagaimana alur aplikasi berjalan. Tampilan antarmuka juga mulai diperindah dengan penambahan CSS dan pengaturan gaya visual, sehingga desain terlihat lebih menarik, interaktif, dan mendekati tampilan akhir aplikasi yang sebenarnya.

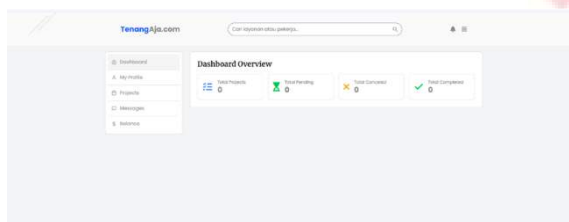


GAMBAR 10
(HASIL PERCOBAAN 2)

Gambar 10 ditampilkan penambahan tampilan antarmuka yang sudah dilengkapi dengan tombol "Mulai" sebagai fitur untuk melanjutkan untuk memulai aplikasi dalam penggunaan.

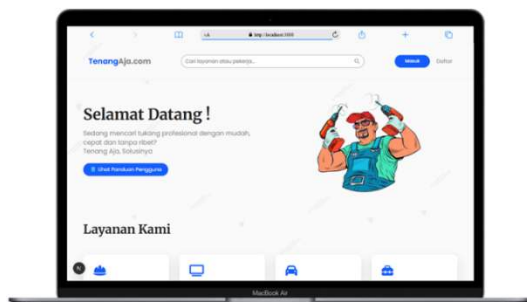


GAMBAR 11
(HASIL PERCOBAAN 2)



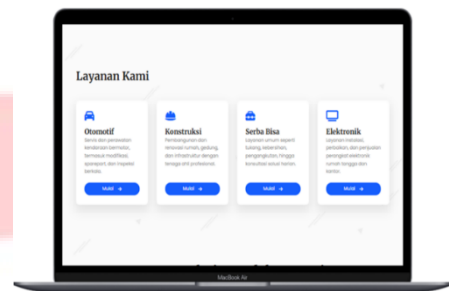
GAMBAR 12
(HASIL PERCOBAAN 2)

Gambar 11 & 12 menunjukkan hasil percobaan kedua dalam merancang tampilan dashboard untuk penyedia jasa. Pada tampilan ini, terdapat beberapa fitur utama seperti My Profile untuk melihat dan mengatur data pribadi, Project untuk mengelola daftar pekerjaan yang sedang berlangsung maupun yang telah selesai, Pesan untuk berkomunikasi langsung dengan pengguna, serta Balance yang menampilkan informasi saldo atau penghasilan penyedia jasa.



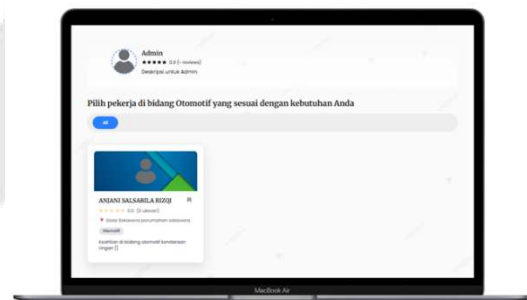
GAMBAR 14
(HASIL PERCOBAAN 2)

Pada gambar hasil akhir perancangan tahap 1 bagian dashboard merupakan tampilan awal sistem yang berperan sebagai titik awal interaksi antara pengguna dan aplikasi. Tampilan ini mencakup elemen-elemen penting seperti logo aplikasi, tombol login dan register, fitur pencarian, serta bagian hero yang berfungsi sebagai penyambutan awal bagi pengguna. Desain ini dirancang untuk memberikan kesan pertama yang informatif dan menarik, sekaligus mempermudah navigasi ke fitur utama aplikasi..



GAMBAR 15
(HASIL PERCOBAAN 2)

Pada Gambar Hasil Akhir Perancangan 2, ditampilkan bagian fitur layanan yang mencakup kategori seperti Otomotif, Konstruksi, Serba Bisa, dan Elektronik. Pada tampilan ini, pengguna yang berperan sebagai pencari jasa dapat mulai menjelajahi dan memilih penyedia jasa sesuai dengan kategori layanan yang dibutuhkan.



GAMBAR 16
(HASIL PERCOBAAN 2)

Ada Gambar Hasil Akhir Perancangan 2, fitur layanan pada bagian *Otomotif* menampilkan daftar penyedia jasa yang menawarkan layanan di bidang tersebut, sehingga pengguna dapat dengan mudah memilih jasa sesuai kebutuhan mereka.

3. Analisis Percobaan

a. Analisis Percobaan 1

Setelah aplikasi selesai diproduksi, langkah selanjutnya adalah melakukan tahap uji coba. Dalam proses ini, peneliti melakukan pengujian langsung bersama salah satu penyedia jasa, yaitu Mas Echo. Berdasarkan hasil interaksi tersebut, Mas Echo memberikan masukan agar aplikasi dilengkapi dengan fitur pembayaran digital, seperti metode transfer, untuk mempermudah proses transaksi antara pengguna dan penyedia jasa.

Pengujian *Blackbox*

Pada tahap pengujian, dilakukan uji coba terhadap salah satu penyedia jasa di bidang layanan serba bisa, yaitu Mas Toni Aprianto, melalui platform Google Meet pada tanggal 26 April 2025. Metode pengujian diterapkan pada setiap tahapan pengembangan sistem dengan tujuan untuk mengevaluasi sejauh mana aplikasi dapat digunakan secara efektif oleh pengguna.

V. KESIMPULAN

Berdasarkan hasil eksplorasi dan refleksi pengembangan, aplikasi *Tukang* berhasil menjadi solusi platform pertemuan antara penyedia jasa dan pelanggan, dengan tampilan antarmuka responsif berbasis Next.js dan Tailwind CSS. Fitur utama seperti registrasi, pencarian tukang, dan sistem booking telah berjalan lancar, didukung backend MySQL dan Prisma ORM yang aman dan terstruktur. Meskipun API sudah cukup stabil, diperlukan peningkatan pada aspek keamanan dan performa, terutama saat trafik tinggi. Aplikasi ini mampu memenuhi kebutuhan dasar kedua pihak dan diharapkan mempermudah pengelolaan jasa secara online secara lebih profesional, meskipun beberapa penyempurnaan seperti autentikasi dan otorisasi masih perlu dilakukan.

REFERENSI

- [1] J. Astuti and E. Fatma, "Evaluasi Pemilihan Penyedia Jasa Kurir Berdasarkan Metode Analytical Hierarchy Process (Ahp)," *J. Manaj. Ind. Dan Logistik*, vol. 1, no. 1, p. 28, 2017, doi: 10.30988/jmil.v1i1.5.
- [2] J. Maylia Suhendro, M. Sudarma, and D. Care Khrisne, "Rancang Bangun Aplikasi Seluler Penyedia Jasa Perawatan Dan Kecantikan Menggunakan Framework Flutter," *J. SPEKTRUM*, vol. 8, no. 2, p. 68, 2021, doi: 10.24843/spektrum.2021.v08.i02.p9.
- [3] Janiver, W. Janis, D. J. Mamahit, B. A. Sugiarso, and A. M. Rumagit, "Rancang Bangun Aplikasi Online SistemPemesanan Jasa Tukang Bangunan BerbasisLokasi," *J. Tek. Inform.*, vol. 15, no. 1, p. 1, 2020.
- [4] A. Zikrillah and D. Gusman, "Sistem Informasi Pencarian Jasa Tukang Berbasis Web (Analysis)," *J. Pustaka Cendekia Inform.*, vol. 1, no. 1, pp. 38–44, 2023, doi: 10.70292/pctif.v1i1.11.
- [5] S. Susliansyah, H. Sumarno, H. Priyono, and L. Maulida, "Implementasi Rancangan Aplikasi Pemesanan Jasa Tukang Bangunan Berbasis Android dengan Metode Waterfall," *REMIK Ris. dan E-Jurnal Manaj. Inform. Komput.*, vol. 8, no. 3, pp. 801–809, 2024.
- [6] D. Pranata, H. Hamdani, and D. M. Khairina, "Rancang Bangun Website Jurnal Ilmiah Bidang Komputer (Studi Kasus : Program Studi Ilmu Komputer Universitas Mulawarman)," *Inform. Mulawarman J. Ilm. Ilmu Komput.*, vol. 10, no. 2, p. 25, 2015, doi: 10.30872/jim.v10i2.187.
- [7] A. Voutama, "Sistem Antrian Cucian Mobil Berbasis Website Menggunakan Konsep CRM dan Penerapan UML," *Komputika J. Sist. Komput.*, vol. 11, no. 1, pp. 102–111, 2022, doi: 10.34010/komputika.v11i1.4677.
- [8] S. Suhari, A. Faqih, and F. M. Basysyar, "Sistem Informasi Kepegawaian Menggunakan Metode Agile Development di CV. Angkasa Raya," *J. Teknol. dan Inf.*, vol. 12, no. 1, pp. 30–45, 2022, doi: 10.34010/jati.v12i1.6622.
- [9] T. A. Pertiwi, N. Try Luchia, P. Sinta, A. Dahlia, I. Rachmat Fachrezi, and R. Aprinastya, "Perancangan Dan Implementasi Sistem Informasi Absensi Berbasis Web Menggunakan Metode Agile Software Development," *J. Test. dan Implementasi Sist. Inf.*, vol. 1, no. 1, pp. 53–66, 2023, doi: 10.55583/jtisi.v1i1.325.
- [10] A. A. Arsyad, M. Mashud, and A. Sumardin, "Implementasi Metode Agile Scrum Pada Sistem Informasi Akuntansi CV Tritama Inti Persada," *J. Ilm. Sist. Inf. Akunt.*, vol. 2, no. 2, pp. 82–87, 2022, doi: 10.33365/jimasia.v2i2.2241.
- [11] R. Indah Melyani, R. Rosita, and S. Aji, "Pengembangan Sistem Informasi Penggajian Berbasis Web Menggunakan Framework Laravel dengan Metode Agile Software Development," *J. Sist. Inf. Akunt.*,

- vol. 3, no. 1, pp. 31–36, 2023, doi: 10.31294/jasika.v3i01.2195.
- [12] S. P. Collins *et al.*, “No Title 済無No Title No Title No Title,” vol. 30, no. 3, pp. 505–520, 2021.
- [13] A. B. Prahastyo, A. Triayudi, and B. Rahman, “E-Commerce Produk Hasil Pertanian Berbasis Web dengan Metode Agile Software Development,” *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 3, no. 6, pp. 1334–1339, 2023, doi: 10.30865/klik.v3i6.911.
- [14] H. Handayani, A. M. Ayulya, K. U. Faizah, D. Wulan, and M. F. Rozan, “Perancangan Sistem Informasi Inventory Barang Berbasis Web Menggunakan Metode Agile Software Development,” *J. Test. dan Implementasi Sist. Inf.*, vol. 1, no. 1, pp. 29–40, 2023, doi: 10.55583/jtisi.v1i1.324.
- [15] A. D. Putra and A. D. Putra, “Rancang Bangun Aplikasi E-Commerce Untuk Usaha Penjualan Helm,” *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 1, no. 1, pp. 17–24, 2020, doi: 10.33365/jatika.v1i1.145.
- [16] D. Debiyanti, S. Sutrisna, B. Budrio, A. K. Kamal, and Y. Yulianti, “Pengujian Black Box pada Perangkat Lunak Sistem Penilaian Mahasiswa Menggunakan Teknik Boundary Value Analysis,” *J. Inform. Univ. Pamulang*, vol. 5, no. 2, p. 162, 2020, doi: 10.32493/informatika.v5i2.5446.
- [17] H. Yuliansyah, “Perancangan Replikasi Basis Data Mysql Dengan Mekanisme Pengamanan Menggunakan Ssl Encryption,” *J. Inform.*, vol. 8, no. 1, pp. 826–836, 2014, [Online]. Available: <http://journal.uad.ac.id/index.php/JIFO/article/view/2081>
- [18] A. D. Ramadhan and Y. Prayudi, “Implementasi Object-Relational Mapping (Orm) Prisma Dalam Perancangan Restful Api Untuk Web Sda Division Di Pt Telkom Indonesia Tbk,” *Technol. J. Ilm.*, vol. 16, no. 2, p. 256, 2025, doi: 10.31602/tji.v16i2.17880.
- [19] R. M. Rizky and N. Santoso, “Pengembangan Sistem Informasi Manajemen Katering (Studi Kasus: Catering Jasmine Koperasi Wanita Patra Kota Dumai),” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 6, pp. 2835–2842, 2023, [Online]. Available: [https://j-ptiik.ub.ac.id/index.php/j-](https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/12862)
- ptiik/article/view/12862
- [20] M. T. Prihandoyo, “Unified Modeling Language (UML) Model Untuk Pengembangan Sistem Informasi Akademik Berbasis Web,” *J. Inform. J. Pengemb. IT*, vol. 3, no. 1, pp. 126–129, 2018, doi: 10.30591/jpit.v3i1.765.
- [21] A. Maulana, H. Hasan, D. Silvi Masfufah, D. Hidayat, L. Nidasari, and W. Amjad Humaid, “Proyek Pengembangan Aplikasi Pengelolaan Kasir Dan Transaksi,” *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 9, no. 3, pp. 4572–4577, 2025, doi: 10.36040/jati.v9i3.13714.
- [22] V. Febrian, M. R. Ramadhan, M. Faisal, and A. Saifudin, “Pengujian pada Aplikasi Penggajian Pegawai dengan menggunakan Metode Blackbox,” vol. 5, no. 1, pp. 61–66, 2020.
- [23] T. S. Jaya, “Pengujian Aplikasi dengan Metode Blackbox Testing Boundary Value Analysis (Studi Kasus: Kantor Digital Politeknik Negeri Lampung),” *J. Inform. J. Pengemb. IT*, vol. 3, no. 1, pp. 45–48, 2018, doi: 10.30591/jpit.v3i1.647.
- [24] F. Abdussalaam and A. Ramdani, “Perancangan Sistem Informasi Manajemen Praktek Kerja Lapangan Berbasis Web Menggunakan Metode Agile,” *INFOKOM (Informatika & Komputer)*, vol. 10, no. 2, pp. 33–43, 2023, doi: 10.56689/infokom.v10i2.950.
- [25] I. P. Andhikha, “Implementasi dan Analisis Kinerja MySQL Cluster Menggunakan Metode Load Balancing IMPLEMENTASI DAN ANALISIS KINERJA MySQL CLUSTER MENGGUAKAN METODE LOAD BALANCING,” 2020, [Online]. Available: www.dev.mysql.com.