

NextJob : Sistem Informasi Pencarian Pekerjaan Berbasis Android Menggunakan Metode Agile

Muh Raenaldy^{1*}, Fitria Nur Dina Salam², Muh Yasin Kadir³, Reyna Buryani Al'safinat.R⁴, Muh Fikra Junian Riswansyah⁵

^{1,2,3,4,5}Universitas Negeri Makassar, Indonesia

¹muhraenaldy25@email.com, ²fitrislm6@email.com, ³mhmmadyasin05@email.com, ⁴reynaburyani@email.com,

⁵fikrajr328@email.com

ITEJ Journal

Article History:

Submitted: 27-12-2024

Accepted: 27-02-2025

Published: 13-03-2025

Keywords:

Android Apps; Agile Methods;

Job Search; Software

Development.

*Corresponding Author:

Muh Raenaldy

ABSTRAK

Sistem Informasi Lowongan Pekerjaan Sekitar Kota Makassar adalah platform digital yang menyediakan informasi tentang pekerjaan yang tersedia di sekitar kota Makassar, termasuk deskripsi pekerjaan, persyaratan, dan cara mengajukan lamaran. Pencarian kerja adalah proses penting bagi individu yang mencari peluang kerja untuk mencocokkan keterampilan dan minat mereka dengan posisi pekerjaan yang sesuai. Pada penelitian ini, metode yang digunakan adalah model pengembangan perangkat lunak *Agile*. Berbeda dengan pendekatan tradisional, metode ini memprioritaskan kemampuan beradaptasi dan keterlibatan pelanggan, kontras dengan pendekatan tradisional, yang mengarah pada pengiriman perangkat lunak yang lebih cepat dan lebih akurat. Penelitian ini berhasil mengembangkan aplikasi NextJob yang berfungsi sebagai platform pencarian kerja bagi masyarakat Kota Makassar. Menggunakan metodologi *Agile*, aplikasi ini mampu menyediakan solusi yang adaptif dan tepat guna dalam menghubungkan pencari kerja dengan peluang pekerjaan yang sesuai. Implementasi dan pengujian aplikasi menunjukkan bahwa NextJob efektif dalam meningkatkan proses pencarian kerja, yang pada gilirannya diharapkan dapat membantu mengurangi tingkat pengangguran di daerah tersebut.

PENDAHULUAN

Sistem Informasi Lowongan Pekerjaan Sekitar Kota Makassar adalah platform digital yang menyediakan informasi tentang pekerjaan yang tersedia di sekitar kota Makassar, termasuk deskripsi pekerjaan, persyaratan, dan cara mengajukan lamaran. Pencarian kerja adalah proses penting bagi individu yang mencari peluang kerja untuk mencocokkan keterampilan dan minat mereka dengan posisi pekerjaan yang sesuai. Di pasar kerja yang kompetitif saat ini, pencari kerja sering menghadapi tantangan dalam mengidentifikasi lowongan pekerjaan yang relevan dan mengamankan pekerjaan yang selaras dengan tujuan karir mereka [1]. Menurut data yang dirilis oleh Dinas Ketenagakerjaan Kota Makassar, pada tahun 2023, tingkat pengangguran terbuka di Kota Makassar mencapai 10,60 persen. Angka tersebut menunjukkan penurunan dari tahun sebelumnya, di mana pada tahun 2022 mencapai 11,68 persen, dan lebih rendah dari tahun 2021 yang mencatatkan 13,18 persen. Ini menandakan adanya peningkatan kondisi ketenagakerjaan di Kota Makassar dalam beberapa tahun terakhir. Dengan menyediakan platform Sistem Informasi Lowongan Pekerjaan di sekitar Kota Makassar merupakan sebuah solusi yang dibutuhkan saat ini. Sistem Android dirancang untuk mengatasi masalah belajar tanpa tujuan di antara mahasiswa sarjana dan kesenjangan antara keterampilan lulusan dan harapan pemberi kerja khususnya di kota Makassar [2].

Dalam penelitian Prabowo (2022) tidak membahas potensi bias atau keterbatasan dalam Tes Penerimaan Pengguna yang dilakukan, yang dapat memengaruhi penilaian keseluruhan kesesuaian aplikasi dengan kebutuhan dan preferensi pengguna dalam mencari pekerjaan. Sistem Android dirancang untuk mengatasi masalah belajar tanpa tujuan di antara mahasiswa sarjana dan kesenjangan antara keterampilan lulusan dan harapan pemberi kerja. Dengan memanfaatkan rekomendasi pekerjaan yang dipersonalisasi berdasarkan preferensi karier [3], portal kerja memainkan peran penting dalam menghubungkan pencari kerja dengan peluang yang sesuai. Platform ini mengurangi gesekan pencarian kerja di pasar tenaga kerja formal dan informal, sehingga memberikan manfaat bagi lulusan dengan meningkatkan hasil kerja [4]. Selain itu, sistem papan kerja online khusus perguruan tinggi menawarkan pengalaman pencarian kerja yang ditargetkan bagi siswa, membantu transisi ke dunia kerja dengan mencocokkan mereka dengan daftar pekerjaan yang relevan [5]. Evolusi portal kerja elektronik telah merevolusi proses rekrutmen, menyediakan database karyawan potensial untuk perekrut dan daftar perusahaan untuk pencari kerja [5].

Filosofi Pemilihan Brand “NextJob” untuk Aplikasi ini berfokus pada tujuan utama yang ingin dicapai, yaitu menyediakan platform yang membantu pencari kerja untuk menemukan “Pekerjaan” yang sesuai dengan kualifikasi dan preferensi mereka. Kata “NEXT” mengandung makna progresif dan prospektif, yang memberikan kesan bahwa aplikasi ini membuka pintu menuju peluang Karir baru yang lebih baik. Sedangkan “JOB” jelas Menunjukkan fokus utama aplikasi pada menyediakan informasi dan akses ke lowongan pekerjaan.

Pemilihan brand "NextJob" diharapkan dapat merefleksikan semangat dan janji aplikasi ini kepada pengguna membantu mereka menemukan "pekerjaan" yang lebih baik, lebih sesuai, dan lebih prospektif bagi karir mereka. Nama ini juga mudah diingat dan memiliki makna yang jelas sehingga dapat dengan mudah dipahami oleh target pengguna aplikasi ini, yaitu pencari kerja di sekitar Kota Makassar.

METODE

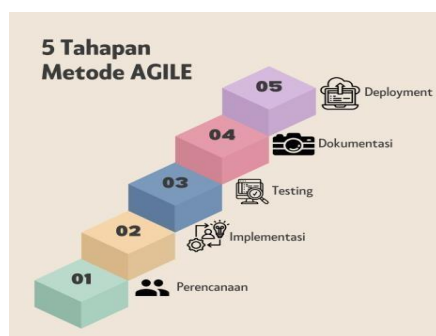
Pada penelitian ini, metode yang digunakan adalah model pengembangan perangkat lunak *Agile*. Berbeda dengan pendekatan tradisional, metode ini memprioritaskan kemampuan beradaptasi dan keterlibatan pelanggan, kontras dengan pendekatan tradisional, yang mengarah pada pengiriman perangkat lunak yang lebih cepat dan lebih akurat [6]. Pendekatan tangkas memungkinkan tim untuk mengirimkan perangkat lunak lebih cepat dan dengan tingkat akurasi yang lebih tinggi melalui iterasi yang cepat dan umpan balik yang berkelanjutan dari pelanggan. Hal ini memungkinkan tim Anda dengan cepat merespons perubahan permintaan pasar dan meningkatkan kepuasan pelanggan serta produktivitas secara keseluruhan [7].

Metodologi *Agile* termasuk Scrum. Metodologi scrum merupakan metodologi pengembangan perangkat lunak yang responsif berdasarkan metodologi *agile* yang dapat memberikan nilai dan manfaat terbaik terhadap perangkat lunak yang dikembangkan. Metode scrum telah menjadi standar industri, hal ini memungkinkan perusahaan untuk merespons dengan cepat terhadap perubahan permintaan pasar dan meningkatkan kepuasan pelanggan dan produktivitas keseluruhan [8]. Scrum menyederhanakan manajemen proyek dengan membagi pekerjaan menjadi unit-unit kecil, jangka pendek, dan dapat dikelola yang disebut sprint. Setiap sprint biasanya berlangsung 2 hingga 4 minggu. Selama periode ini, tim berfokus pada penyelesaian tugas tertentu dan mengadakan pertemuan harian untuk memantau kemajuan. Melalui proses ini, perusahaan dapat dengan cepat merespons perubahan permintaan pasar dan meningkatkan kepuasan pelanggan. Penelitian menunjukkan bahwa penggunaan Scrum meningkatkan produktivitas tim secara keseluruhan dan kualitas perangkat lunak yang dihasilkan [7].

Beragamnya metodologi tangkas yang tersedia menimbulkan tantangan dalam memilih metode yang paling tepat, meningkatkan kebutuhan akan analisis kinerja dan berpotensi memanfaatkan kecerdasan buatan untuk mengoptimalkan kualitas perangkat lunak [9]. Berbagai metodologi *agile* yang tersedia saat ini menawarkan pendekatan perkakas yang berbeda untuk pengembangan perangkat lunak, namun meskipun *agile* memiliki banyak manfaat, keragaman metodologi yang ada membuat sulit untuk beradaptasi dengan kebutuhan spesifik suatu proyek atau organisasi metode untuk Karena Scrum, Kanban, dan Extreme Programming (XP) menghadirkan tantangan unik dalam memilih pendekatan terbaik untuk setiap proyek. Oleh karena itu, analisis kinerja dan evaluasi menyeluruh terhadap persyaratan proyek sangatlah penting. Beberapa penelitian terbaru juga menyarankan penggunaan kecerdasan buatan untuk mengoptimalkan proses pengembangan perangkat lunak dalam kerangka tangkas. AI memungkinkan Anda memprediksi dan mengatasi potensi hambatan serta membuat keputusan berdasarkan data yang lebih tepat waktu [7].

Secara keseluruhan, metodologi tangkas memberikan pendekatan dinamis dan kolaboratif untuk pengembangan perangkat lunak. Dengan menekankan iterasi yang cepat, keterlibatan pelanggan, dan manajemen proyek yang efisien, *Agile* memungkinkan tim pengembangan menjadi lebih fleksibel dan responsif terhadap perubahan kebutuhan. Hal ini tidak hanya meningkatkan efisiensi operasional, namun juga memastikan bahwa produk akhir lebih memenuhi harapan dan kebutuhan pengguna. Metodologi tangkas juga mendukung peningkatan kualitas produk melalui praktik-praktik seperti pengujian berkelanjutan, integrasi berkelanjutan, dan retrospektif. Pengujian berkelanjutan memungkinkan identifikasi dini terhadap masalah, sehingga bisa segera diperbaiki sebelum menjadi lebih kompleks. Integrasi berkelanjutan membantu memastikan bahwa semua bagian sistem bekerja dengan baik bersama-sama, mencegah konflik yang bisa muncul di kemudian hari.

Selain itu, sesi retrospektif memberikan kesempatan bagi tim untuk mengevaluasi proses yang telah berjalan, mengidentifikasi area yang perlu diperbaiki, dan mengimplementasikan perubahan yang dapat meningkatkan efektivitas dan efisiensi kerja tim. Dengan semua elemen ini, metodologi tangkas tidak hanya meningkatkan responsivitas terhadap perubahan, tetapi juga secara proaktif meningkatkan kualitas produk dan proses pengembangan itu sendiri. Metode ini mengakomodasi umpan balik terus-menerus dari pengguna dan pemangku kepentingan, memastikan bahwa produk selalu relevan dan bermanfaat, yang pada akhirnya meningkatkan kepuasan pelanggan dan daya saing di pasar.



Gambar 1. Tahapan Metode Agile

2.1 Tahap Perencanaan

Tahap Perencanaan merupakan tahap pertama yang merancang secara detail setiap tindakan, tujuan, serta berbagai aktivitas-aktivitas yang dibutuhkan untuk pengembangan sistem informasi [10]. Pada tahap ini, developer akan mendefinisikan cara kerja dari suatu sistem informasi atau aplikasi, serta menentukan tampilan desain yang akan digunakan. Selain itu, perencanaan sistem yang akan dikembangkan disusun melalui pengumpulan data dari pengguna melalui wawancara langsung untuk memahami kebutuhan mereka. Setelah itu, desain sistem secara menyeluruh dilakukan dengan menggunakan alat UML dan antarmuka pengguna oleh pengembang *system* [8].

2.2 Tahap Implementasi

Tahap Implementasi adalah tahapan dimana sistem akan diterapkan berdasarkan hasil dari analisis dan desain dari sistem yang sebelumnya telah dirancang. Selain itu pada tahap ini juga perancangan code dari program yang dibuat harus sesuai dengan desain yang telah dirancang sebelumnya menggunakan Bahasa pemrograman [10]. Selama tahap implementasi, rencana yang sudah disusun sebelumnya akan diterapkan untuk menciptakan sistem. Proses pengembangan sistem ini menggunakan bahasa pemrograman Java dengan database Firebase [6].

2.3 Tahap Testing

Tahap Testing atau tahap pengujian merupakan tahap dimana aplikasi yang dirancang akan diuji untuk mengevaluasi apakah sistem dapat berfungsi dengan semestinya [10]. Pada tahapan ini, dilakukan pengujian terhadap perangkat lunak yang telah dikembangkan pada tahap implementasi. Pengujian aplikasi bertujuan untuk mengevaluasi sejauh mana fungsionalitas program telah tercapai dengan baik. Hal ini mencakup pengecekan terhadap kesesuaian alur aplikasi dengan proses bisnis yang telah ditetapkan, serta identifikasi kemungkinan kesalahan atau bug dalam pengembangan aplikasi tersebut [9].

2.4 Tahap Dokumentasi

Tahap Dokumentasi merupakan tahapan dimana tim developer akan menyusun berbagai macam dokumen yang diperlukan untuk menjelaskan berbagai macam fungsi dan modul yang ada pada aplikasi yang telah dirancang. Hal ini bertujuan agar bahan developing dapat diberikan kepada tim pengembang yang akan mengembangkan aplikasi tersebut di masa depan, dan juga dapat memudahkan tim developer untuk mengelola dan memelihara sistem informasi tersebut [10]. Pada fase ini, modul dan fungsi yang ada dalam sistem informasi didokumentasikan sebagai catatan selama proses pengembangan dan untuk memudahkan tim dalam langkah-langkah pengembangan berikutnya [8].

2.5 Tahap Deployment

Tahap Deployment adalah tahapan dimana hasil akhir dari pengembangan sistem informasi akan diunggah ke layanan penyedia aplikasi mobile (dalam hal ini PlayStore/AppStore) agar dapat diakses dan digunakan oleh pengguna melalui koneksi internet. Hal ini bertujuan agar sistem informasi dapat bekerja dengan semestinya dan dapat sesuai dengan apa yang diharapkan oleh user [10]. Pada tahap ini, sistem yang telah dikembangkan disiapkan untuk digunakan oleh pengguna akhir, termasuk tim pelaksana. Hal ini bertujuan untuk memastikan bahwa sistem tersebut dapat digunakan secara efektif oleh para masyarakat yang menggunakan aplikasi ini [6].

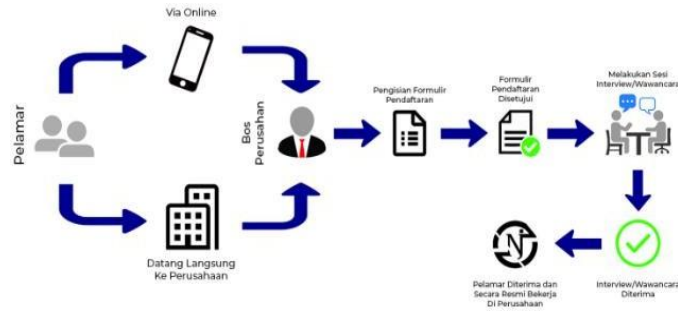
HASIL DAN PEMBAHASAN

3.1 Tahap Perencanaan

Pada tahapan ini, peneliti akan mewawancarai user terkait kebutuhan fitur sistem yang akan dikembangkan sesuai dengan kebutuhan pasar. Tahap perencanaan dalam rekayasa perangkat lunak adalah fase pertama yang sangat krusial dan menentukan arah keseluruhan proyek. Keberhasilan atau kegagalan proyek sering kali ditentukan oleh seberapa baik fase perencanaan ini dilaksanakan. Fase ini mencakup beberapa langkah penting, mulai dari memahami kebutuhan pengguna dan bisnis hingga menyusun desain sistem yang komprehensif. Secara keseluruhan, tahap perencanaan dalam rekayasa perangkat lunak adalah proses yang kompleks dan membutuhkan perhatian yang mendalam terhadap detail. Dengan melakukan perencanaan yang baik, tim pengembang dapat memastikan bahwa proyek berjalan lancar dan memenuhi kebutuhan pengguna dan bisnis dengan efektif.

Perencanaan yang matang dan terstruktur akan membantu mengurangi risiko dan memastikan bahwa setiap langkah dalam proses pengembangan perangkat lunak dapat dilaksanakan dengan efisien dan tepat waktu. Secara keseluruhan, tahap perencanaan dalam rekayasa perangkat lunak membutuhkan dedikasi yang tinggi terhadap detail dan pemahaman yang mendalam tentang kebutuhan pengguna dan bisnis. Dengan melakukan perencanaan yang baik dan terstruktur, tim pengembang dapat meningkatkan peluang keberhasilan proyek dan memastikan bahwa sistem yang dikembangkan dapat memberikan nilai tambah yang signifikan bagi pengguna dan bisnis yang bersangkutan. Kemudian dalam perencanaan pengembangannya, prosedur yang akan dilakukan user setelah menggunakan sistem NextJob nantinya dapat

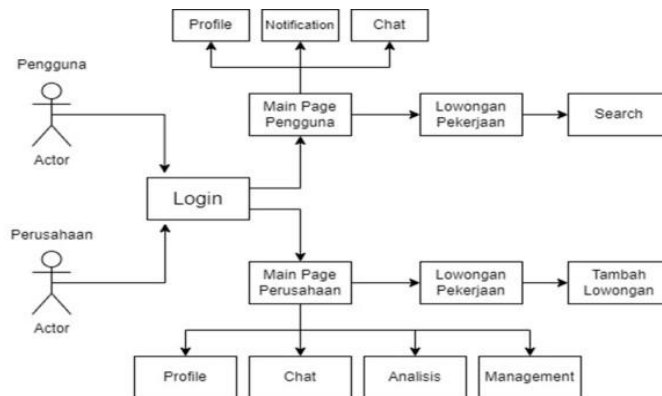
dilihat pada gambar 2.



Gambar 2. Perancangan Prosedur Penggunaan NextJob

a. Use Case Diagram

Usecase Diagram dalam sistem pencarian kerja NextJob menggambarkan interaksi antara pengguna dan sistem, serta fungsi dan peran yang berbeda dalam sistem[11]. Ini membantu mengidentifikasi tindakan pengguna dan interaksinya selama pencarian pekerjaan. Diagram ini juga menentukan perilaku dan fungsionalitas sistem tanpa memerlukan detail implementasi[12]. Dengan menggunakan analisis dan desain berorientasi objek seperti UML, sistem ketenagakerjaan dan rekrutmen dapat dikembangkan secara efektif, mencakup diagram seperti kasus penggunaan, kelas manajemen, pencarian kerja, rekrutmen, urutan registrasi, dan aktivitas modifikasi[13][14]. Pendekatan ini memfasilitasi pemahaman dan perancangan sistem serta interaksi pengguna seperti yang ditunjukkan pada gambar 3.



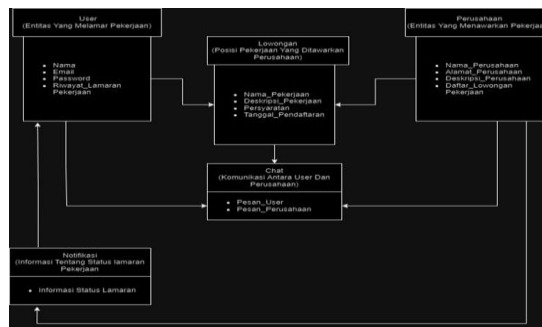
Gambar 3. Desain Use Case Diagram Aplikasi NextJob

b. Class Diagram

Class Diagram adalah jenis diagram *UML* yang menampilkan struktur statis suatu sistem dengan menunjukkan kelas-kelas dalam sistem, beserta atribut, metode, dan hubungan antar kelas. Diagram ini menggunakan kotak dengan tiga bagian untuk menggambarkan nama kelas, atribut, dan metode. Hubungan antar kelas, seperti asosiasi, pewarisan, dan *agregasi*, digambarkan dengan berbagai jenis garis [15]. Class diagram memiliki 3 relasi yang digunakan, yaitu :

Association, sebuah relasi yang memperlihatkan adanya suatu interaksi diantara class. Kedua generalization, sebuah relasi diantara class yang bersifat dari khusus ke umum. Dan yang ketiga constraint, sebuah relasi yang ada pada sistem untuk memberikan sebuah batasan pada sistem hingga ditemukannya aspek yang tidak fungsional [16].

Interaksi antar kelas dalam *Class* diagram menunjukkan bagaimana pengguna dan perusahaan dapat menggunakan aplikasi untuk mencari pekerjaan, melamar pekerjaan, dan berkomunikasi satu sama lain, dapat dilihat pada gambar 4.



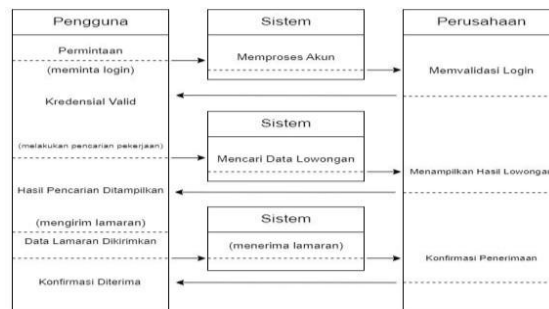
Gambar 4. Class Diagram NextJob dalam Firebase Database

c. Sequence Diagram

Sequence Diagram adalah representasi perilaku UML yang menggambarkan bagaimana objek berinteraksi dan bertukar pesan dari waktu ke waktu. Sequence Diagram menunjukkan bagaimana pesan dikirim antara objek atau instansi lain untuk melakukan tugas. Mereka digunakan selama fase desain detail, di mana komunikasi antarpemroses yang tepat harus dibangun sesuai dengan protokol formal. Sequence Diagram Memainkan peran penting dalam membantu pengguna untuk memahami operasi sistem dan memvisualisasikan interaksi antara objek sistem [17].

Selain itu, diagram ini menjembatani model kasus pengguna UML, dan kelas objek yang ditentukan dalam model struktural. Pada tahapan ini, tim pengembangan akan menggambarkan interaksi antar-objek dalam sistem berdasarkan urutan waktu dalam bentuk *sequence* diagram [18]. Ini menunjukkan bagaimana objek-objek saling bertukar pesan untuk melakukan suatu fungsi seperti yang terlihat pada gambar 5 [17][19]. Sequence diagram ini sangat penting untuk mengidentifikasi alur kerja dan memastikan bahwa semua interaksi berjalan sesuai dengan kebutuhan yang telah diidentifikasi. Lebih lanjut, sequence diagram membantu dalam menemukan potensi bottleneck dan masalah dalam alur komunikasi antar-objek yang mungkin tidak terlihat dalam model statis. Selain itu, diagram ini berfungsi sebagai panduan bagi pengembang selama fase implementasi, memastikan bahwa semua elemen sistem berinteraksi secara harmonis dan efisien.

Dengan demikian, sequence diagram tidak hanya mendokumentasikan arsitektur dinamis dari sistem tetapi juga memfasilitasi komunikasi yang lebih baik di antara anggota tim pengembang, serta antara tim pengembang dan pemangku kepentingan.

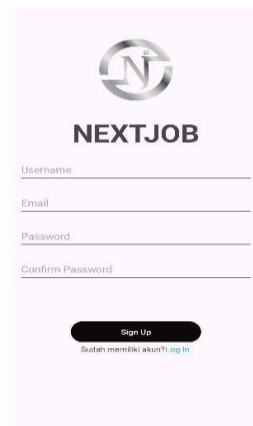


Gambar 5. Sequence Diagram Sistem Informasi NextJob

3.2 Tahap Implementasi

a. Register

Fitur SignUp bagi pengguna pada aplikasi NextJob memungkinkan pengguna untuk membuat akun terlebih dahulu jika belum memiliki akun. Pada bagian ini, pengguna akan mendaftarkan Nama, Email, serta membuat *Password* sebagai syarat pembuatan akun. Proses pendaftaran dibuat agar mudah dan ramah pengguna, sehingga memudahkan pengguna untuk membuat akun baru tanpa hambatan. Dengan menawarkan fitur *signup* yang efisien dan aman, NextJob bertujuan untuk memberikan pengalaman pengguna yang luar biasa dan memastikan bahwa data pengguna dikelola dan dilindungi secara efektif. Fitur ini berfungsi sebagai langkah awal yang penting dalam menghubungkan pencari kerja dengan peluang kerja yang sesuai, membantu mereka dalam perjalanan pencarian kerja yang lebih efisien dan terorganisir. *Source code* dari fitur registrasi ini dapat diakses melalui link GitHub berikut : <https://github.com/dinaaaach/NextJob>. Tampilannya *user interface*-nya dapat dilihat pada gambar 6.

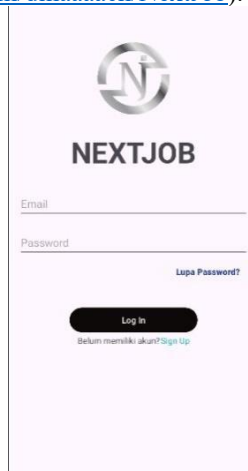


The screenshot shows the registration page of the NextJob application. At the top is the NextJob logo, which consists of a stylized 'N' and 'J' inside a circle. Below the logo, the text 'NEXTJOB' is displayed. The form contains four input fields: 'Username', 'Email', 'Password', and 'Confirm Password'. Below these fields is a black button labeled 'Sign Up'. Underneath the button, there is a link that says 'Sudah memiliki akun? Log In'.

Gambar 6. Tampilan UI Fitur Registrasi

b. Login

Fitur Login bagi pengguna pada aplikasi NextJob mengharuskan pengguna untuk masuk ke akun yang telah dibuat oleh pengguna sebelumnya pada menu SignUp seperti yang ditunjukkan pada gambar 7. Dengan fungsionalitas *Login* yang aman dan efisien, NextJob berdedikasi untuk memberikan pengalaman pengguna yang luar biasa sekaligus menjaga privasi informasi pribadi pengguna. Fungsionalitas ini sangat penting untuk menawarkan pengguna akses tanpa batas dan terlindungi ke berbagai layanan dan sumber daya aplikasi, memfasilitasi upaya pencarian kerja mereka dengan efisiensi dan organisasi yang lebih baik. Adapun link program dari tampilan ini, dapat diakses melalui link GitHub (<https://github.com/dinaaaach/NextJob>).

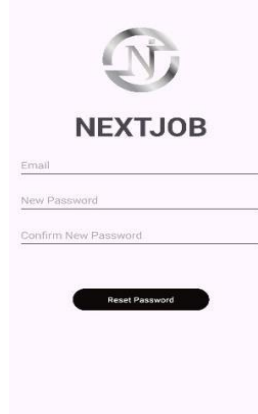


The screenshot shows the login page of the NextJob application. At the top is the NextJob logo, which consists of a stylized 'N' and 'J' inside a circle. Below the logo, the text 'NEXTJOB' is displayed. The form contains two input fields: 'Email' and 'Password'. Below the 'Password' field is a link that says 'Lupa Password?'. Below these fields is a black button labeled 'Log In'. Underneath the button, there is a link that says 'Belum memiliki akun? Sign Up'.

Gambar 7. Tampilan UI Fitur Login

c. Lupa Password

Fitur Reset Password adalah fitur yang berfungsi untuk mengganti/mereset kata sandi jika suatu ketika pengguna lupa kata sandi yang sebelumnya sudah dibuat. Adapun kode programnya dapat dilihat melalui link GitHub berikut : <https://github.com/dinaaaach/NextJob> .



The screenshot shows the reset password page of the NextJob application. At the top is the NextJob logo, which consists of a stylized 'N' and 'J' inside a circle. Below the logo, the text 'NEXTJOB' is displayed. The form contains three input fields: 'Email', 'New Password', and 'Confirm New Password'. Below these fields is a black button labeled 'Reset Password'.

Gambar 8. Tampilan UI Fitur Lupa Password

3.3 Tahap Testing

White Box testing atau pengujian *structural* merupakan metode pengujian perangkat lunak yang menganalisis struktur internal dan kode aplikasi [20]. Metode ini memungkinkan penguji untuk merancang kasus uji yang memastikan semua jalur kode dijalankan. Adapun beberapa kelebihan dan kekurangan dalam pengujian dengan metode *hite Box* :

Kelebihan:

- Metode White Box mampu mengidentifikasi kesalahan pada kode dengan menghapus baris yang tidak perlu.
- Cakupan pengujian aplikasi lebih maksimal saat melakukan uji coba skenario.

Kekurangan:

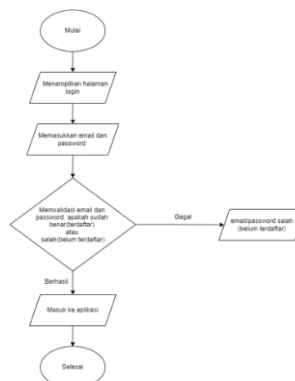
- Pengujian dengan metode White Box membutuhkan biaya yang tinggi karena membutuhkan penguji yang berpengalaman.
- Beberapa alur program mungkin tidak dapat diuji karena tidak memungkinkan untuk memeriksa setiap baris kode untuk menemukan kesalahan.

Dalam penelitian mengenai Tesis Pemantauan Sistem Informasi Program Studi Informatika, pengujian ini digunakan untuk mendeteksi kesalahan dalam sistem melalui analisis kode dan struktur internal. Hasil pengujian mengkategorikan fungsi sistem berdasarkan tingkat risiko, dari rendah hingga tinggi, tergantung pada kompleksitasnya. Pengujian ini mencakup 77,5% dari 285 jalur independen yang ditargetkan, menunjukkan evaluasi menyeluruh terhadap fungsi sistem berdasarkan struktur kodenya[21].

a. Fitur Login

Flowchart :

Flowchart adalah bagan (chart) yang mengarahkan alir (flow) atau yang biasa disebut dengan diagram alir secara logika dalam prosedur atau program sistem. Diagram alir adalah representasi visual dari langkah-langkah dan urutan prosedur dalam sebuah aplikasi. Diagram ini membantu analisis dan pemrogram dengan memberikan gambaran umum tentang aplikasi yang sedang dikembangkan dan memecahnya menjadi bagian-bagian yang lebih kecil dan lebih mudah dikelola, sehingga memudahkan analisis. Penyederhanaan ini membantu dalam pemecahan masalah. Diagram alir digunakan dalam desain aplikasi untuk menggambarkan logika aplikasi [22]. Selain sebagai alat untuk merancang dan mengembangkan aplikasi, flowchart juga dapat digunakan sebagai alat pembelajaran. Para mahasiswa atau pemula dalam dunia pemrograman dapat menggunakan flowchart untuk memahami konsep dasar dalam algoritma dan logika pemrograman.



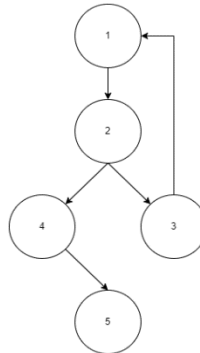
Gambar 9. Flowchart Login

Flowgraph :

Flowgraph dalam *white box* adalah representasi grafis dari aliran control dalam program perangkat lunak, yang membantu memahami struktur internal dan desain untuk tujuan pengujian [23]. *Flowgraph* ini menangkap informasi aliran data dan memfasilitasi pembuatan kasus uji dengan menganalisis jalur yang ada dalam kode. Ini terdiri dari serangkaian simpul yang terhubung oleh segmen garis berarah (cabang) yang memodelkan ketergantungan variabel keluaran terhadap variabel masukan. Penggunaan flowgraph memungkinkan pengidentifikasian jalur eksekusi kritis dalam perangkat lunak, serta area yang rentan terhadap kesalahan atau kerentanan.

Dengan memvisualisasikan aliran control, pengembang dapat lebih mudah mendeteksi anomali atau bug yang mungkin tersembunyi dalam logika program. Selain itu, flowgraph juga dapat digunakan untuk mengoptimalkan kode, dengan cara mengidentifikasi dan menghilangkan jalur yang tidak efisien atau redundan. Implementasi flowgraph dalam pengujian white box memastikan bahwa setiap cabang dan simpul diuji secara menyeluruh, meningkatkan cakupan pengujian dan keandalan

perangkat lunak secara keseluruhan.



Gambar 10. Flowgraph Login

Menghitung Cyclomatic Complexity :

Menghitung Kompleksitas Siklomatik dalam Pengujian *Whitebox* melibatkan pengukuran jumlah jalur independen linier melalui kode sumber program [24]. Metrik ini sangat penting dalam pengembangan perangkat lunak, khususnya dalam pengujian *Whitebox*, di mana fokusnya adalah pada struktur kode [25]. Kompleksitas dihitung menggunakan grafik aliran control program, di mana node mewakili kelompok perintah dan tepi menghubungkan perintah yang dapat dijalankan secara berurutan [26]. Oleh karena itu, kompleksitas siklomatis dan penilaian jumlah rute independen linier melalui kode sumber program menjadi tolok ukur penting dalam pengujian kotak putih. Metrik ini membantu pengembang dalam memahami, mengevaluasi, dan meneliti arsitektur kode sumber program dengan efisiensi yang lebih besar, sehingga dapat menunjukkan dan memperbaiki masalah potensial dengan kecepatan dan akurasi yang lebih baik. Dengan mempertimbangkan kompleksitas siklomatis, pengembang dapat meningkatkan kualitas dan ketergantungan perangkat lunak yang mereka buat, sehingga mengurangi kemungkinan gangguan atau malfungsi dalam program.

Menentukan jumlah Node dan Edge dalam flowgraph login

Node = 5

Edge = 5

Jumlah Region = $E - N + 2 = n$ (Nilai Rangkaian Path)
 $= 5 - 5 + 2 = 2$ Path

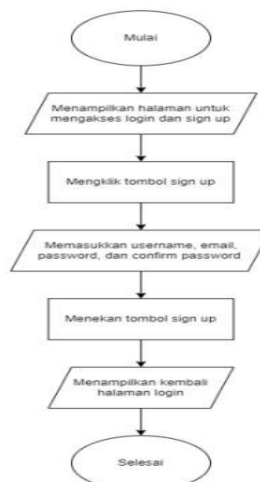
Maka diperoleh jalur pengujian serangkaian path sebagai berikut :

Path 1: 1-2-4-5 (Pass)

Path 2: 1-2-3 (Fail)

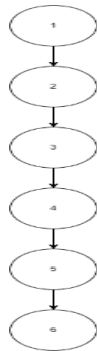
b. Fitur Register

Flowchart :



Gambar 11. Flowchart Register

Flowgraph :



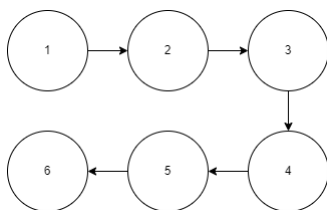
- Menghitung Cyclomatic Complexity
 Node = 6
 Edge = 5
 Jumlah Region = $E - N + 2$
 $= 5 - 6 + 2 = 1$ Path
- Rangkaian jalur path yang diperoleh
 Path 1: 1-2-3-4-5-6 (Pass)

Gambar 12. Flowgraph Register
 c. **Fitur Lupa Password**
Flowchart :



Gambar 13. Flowchart Lupa Password

Flowgraph :



- Menghitung Cyclomatic Complexity
 Node = 6
 Edge = 5
 Jumlah Region = $E - N + 2$
 $= 5 - 6 + 2 = 1$ Path
- Rangkaian jalur path yang diperoleh
 Path 1: 1-2-3-4-5-6 (Pass)

Gambar 14. Flowgraph Lupa Password

a. Membuat Kasus Uji

Tabel 1

Kasus Uji

Kode Fitur	Nama Fitur	Keluaran Sasaran	Hasil Tes
TC 1	Registrasi	User berhasil mendaftarkan akun dan data tersimpan ke dalam database	Passed
TC 2	Login	User berhasil login dan diarahkan ke halaman home page	Passed
TC 3	Lupa Password	Password berhasil diubah	Failed

Berdasarkan Tabel 1 Kasus Uji, analisis menunjukkan bahwa fitur registrasi (TC 1) berhasil dengan baik, di mana pengguna dapat mendaftarkan akun dan data tersimpan ke dalam database sesuai harapan. Fitur login (TC 2) juga berfungsi dengan benar, memungkinkan pengguna berhasil login dan diarahkan ke halaman home page.

Namun, fitur lupa *password* (TC 3) gagal dalam pengujian, mengindikasikan adanya masalah dalam proses pengubahan *password* yang memerlukan perhatian lebih lanjut untuk diperbaiki. Secara keseluruhan, dua dari tiga fitur yang diuji berhasil melewati pengujian, sementara fokus perbaikan perlu diarahkan pada fitur lupa *password* untuk memastikan semua fungsi penting beroperasi dengan baik dan meningkatkan pengalaman pengguna. Tindakan tambahan perlu dilakukan untuk mengidentifikasi alasan yang mendasari kerusakan dan menerapkan perbaikan yang sesuai. Hal ini dapat mencakup peninjauan kembali logika dan implementasi kode yang terkait dengan fitur tersebut, serta melakukan pengujian tambahan untuk memverifikasi fungsionalitas yang diantisipasi. Oleh karena itu, upaya yang bertujuan untuk meningkatkan keandalan dan ketahanan fungsionalitas lupa kata sandi akan meningkatkan perjalanan pengguna secara menyeluruh dan menjamin pengoperasian yang mulus dari semua aspek aplikasi.

3.4 Tahap Dokumentasi

Dokumentasi adalah bagian integral dari setiap proyek pengembangan perangkat lunak, yang mencakup semua informasi penting tentang sistem. Dokumentasi dibagi menjadi dua kategori utama: Dokumentasi pengguna berfokus pada cara menggunakan perangkat lunak dan mencakup panduan pengguna, FAQ, dan tutorial. Dalam aplikasi Sistem Informasi Pencarian Kerja NextJob, dokumentasi yang digunakan dibuat dalam bentuk dokumen Spesifikasi Kebutuhan Perangkat Lunak (SRS). Model *Agile* seperti Scrum dan Kanban menekankan komunikasi verbal dalam tim pengembangan, yang dapat menyebabkan kurangnya dokumentasi formal[27]. Namun, bagi tim Site Reliability Engineering (SRE), dokumentasi yang jelas sangat penting untuk keberhasilan operasional dalam sistem terdistribusi berskala besar [28]. Ketika menerapkan metodologi *Agile*, penting untuk mempertimbangkan jenis dokumen yang diperlukan, seperti arah proyek, peran, templat, dan pengumpulan informasi, untuk memastikan keberhasilan proyek dan keselarasan dengan model proses *Agile* yang dipilih. Selain itu, penelitian ini menunjukkan adanya korelasi antara jumlah dokumentasi dan kepuasan dalam pencarian informasi pada proyek *Agile* yang menekankan pentingnya praktik dokumentasi yang memadai untuk efisiensi pengambilan informasi.

3.5 Tahap Deployment

Tahap deployment adalah fase di mana perangkat lunak yang telah dikembangkan dan diuji siap untuk diimplementasikan di lingkungan produksi. Sistem Pengiriman dan Umpan Balik untuk sistem informasi pencarian kerja NextJob dapat dirancang dengan menggabungkan berbagai elemen dari konteks yang disediakan. Sistem ini dapat mencakup perangkat pengiriman dan penyebaran dengan mekanisme pelumasan katup untuk memastikan penyebaran yang efisien, perangkat pencarian penyebaran yang memanfaatkan informasi kinerja untuk mengoptimalkan konsumsi daya, serta sistem pengiriman implan dengan mekanisme penguncian untuk penyebaran yang akurat [29]. Selain itu, kode umpan balik dapat dihasilkan berdasarkan kinerja penerapan dan disampaikan melalui saluran terpisah untuk pembaruan status [30]. Dengan mengintegrasikan komponen-komponen ini, sistem NextJob dapat meningkatkan akurasi rekomendasi pekerjaan dengan secara efektif memprediksi aplikasi yang akan diajukan pencari kerja berikutnya, dengan mempertimbangkan preferensi pekerjaan, konten tekstual, lokasi geografis, dan mekanisme perhatian yang dipersonalisasi. Proses ini memastikan bahwa sistem tetap relevan dan berfungsi dengan baik dalam jangka Panjang.

KESIMPULAN DAN SARAN

Pada hasil penelitian yang dituangkan ke dalam artikel ini menyajikan pengembangan aplikasi NextJob, sebuah platform pencarian kerja berbasis Android yang dirancang untuk membantu masyarakat Kota Makassar dalam mencari pekerjaan. Melalui penggunaan metodologi *Agile*, aplikasi ini berhasil menciptakan solusi yang adaptif dan tepat guna, menghubungkan pencari kerja dengan peluang kerja yang sesuai dengan keterampilan dan preferensi mereka. *Agile*, yang menekankan iterasi cepat dan keterlibatan pelanggan, memungkinkan tim pengembang untuk merespons perubahan kebutuhan dengan cepat dan efektif. Implementasi NextJob menunjukkan bahwa aplikasi ini dapat meningkatkan proses pencarian kerja dan diharapkan dapat mengurangi tingkat pengangguran di daerah tersebut.

NextJob dirancang untuk mengatasi masalah ketidaksesuaian antara keterampilan lulusan dan kebutuhan pasar kerja. Platform ini menawarkan informasi detail tentang lowongan kerja, termasuk deskripsi pekerjaan, persyaratan, dan cara melamar. Dengan pendekatan yang berfokus pada adaptabilitas dan keterlibatan pengguna, aplikasi ini mampu memberikan rekomendasi pekerjaan yang dipersonalisasi berdasarkan preferensi karir pengguna. Keberhasilan aplikasi ini mencerminkan potensi besar dari penerapan teknologi dan metodologi *Agile* dalam mengatasi tantangan ketenagakerjaan. Implementasi dan pengujian aplikasi menunjukkan bahwa NextJob efektif dalam meningkatkan efisiensi dan akurasi pencarian kerja, yang pada akhirnya diharapkan dapat mengurangi tingkat pengangguran di Kota Makassar. Hal ini menunjukkan bahwa pendekatan *Agile* tidak hanya meningkatkan efisiensi operasional, tetapi juga memastikan bahwa produk akhir lebih sesuai dengan kebutuhan pengguna.

Dengan nama "NextJob", aplikasi ini mencerminkan semangat progresif dan prospektif, memberikan kesan bahwa aplikasi ini membuka pintu menuju peluang karir yang lebih baik. Nama ini dipilih untuk mencerminkan tujuan utama aplikasi, yaitu membantu pengguna menemukan pekerjaan yang lebih baik dan sesuai dengan kualifikasi mereka. Secara keseluruhan, artikel ini menunjukkan bahwa pengembangan aplikasi berbasis *Agile* dapat menjadi solusi efektif dalam mengatasi masalah ketenagakerjaan. NextJob sebagai platform pencarian kerja berbasis Android berhasil menunjukkan potensi besar teknologi dalam membantu masyarakat mengatasi tantangan mencari pekerjaan yang sesuai, meningkatkan proses pencarian kerja, dan diharapkan dapat berkontribusi dalam mengurangi tingkat pengangguran di Makassar.

UCAPAN TERIMA KASIH

Puji dan syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, saya dapat menyelesaikan artikel ini. Penulisan artikel ini dilakukan dalam rangka memenuhi tugas akhir mata kuliah Analisis Desain

Sistem Informasi prodi Pendidikan Teknik Informatika dan Komputer, Fakultas Teknik, Universitas Negeri Makassar. Kami ingin mengucapkan terima kasih yang sebesar-besarnya kepada semua yang telah membaca dan mendukung artikel kami yang berjudul ***“NextJob: Sistem Informasi Pencarian Pekerjaan Berbasis Android Menggunakan Metode Agile”***. Tulisan ini sungguh bermakna bagi kami, dan melihat tanggapan positif dari pembaca membuat kami merasa sangat bersemangat.

Tak lupa pula untuk mengucapkan terima kasih kepada PTIK B 22 atas dukungan, kerja sama, dan semangat kalian selama ini. Kehadiran kalian membuat perjalanan ini menjadi lebih menyenangkan dan penuh makna. Terutama kepada anggota kelompok 9, terima kasih atas kerja keras, dedikasi, dan komitmen kalian dalam proyek ini. Kolaborasi kita telah menghasilkan sesuatu yang luar biasa. Bangga bisa bekerja bersama kalian.

Kami berharap bahwa artikel ini dapat memberikan manfaat dan inspirasi bagi semua yang membacanya. Terima kasih atas waktu dan perhatiannya. Saya sangat menghargai setiap komentar dan feedback yang diberikan untuk memperbaiki dan meningkatkan kualitas tulisan saya di masa mendatang. Terima kasih sekali lagi untuk semua dukungan dan apresiasi yang telah diberikan. Semoga kita semua terus berbagi pengetahuan dan pengalaman untuk menciptakan lingkungan yang lebih baik.

REFERENSI

- [1] M. Ammar, D. Firdaus., and P. Agung, “Aplikasi Pencari Tempat Magang Berbasis Android Menggunakan Metode Agile Scrum,” *J. Inform. UPGRIS*, vol. 8, no. 1, doi: 10.26877/jiu.v8i1.12029.
- [2] B. Salathiel and S. Suresh, “Job Search Sustem in Android Environment,” *Appl. Intell. Agents*, vol. 2, no. 3, pp. 1–17, doi: 10.5121/IJIST.2012.2301.
- [3] van Patrick, J. Esch., B. Stewart, and F. Joseph, “Marketing AI recruitment: The next phase in job application and selection,” *Comput. Human Behav.*, vol. 90, pp. 215–222, doi: 10.1016/J.CHB.2018.09.009.
- [4] S. Neha and C. Indu, “Agile Methodologies : A Performance Analysis to Enhance Software Quality.” pp. 181–188. doi: 10.1201/9781003079996-12.
- [5] F. Reza and R. Joko, “Implementation of Pre-Marriage and Parenting Education Information System with Android-Based Agile Method,” *Int. J. Educ. Sci. Technol. Eng.*, vol. 5, no. 1, pp. 43–49, doi: 10.36079/lamintang.ijeste-0501.380.
- [6] D. L. Alexandes, R. Aditio, and Y. Jumaryadi, “Implementasi Metode Agile dalam Pengembangan Sistem E-document,” *J. Inf. Syst. Res.*, vol. 4, no. 1, pp. 318–329, 2022, doi: 10.47065/josh.v4i1.2349.
- [7] H. R. Suharno, N. Gunantara, and M. Sudarma, “Analisis Penerapan Metode Scrum Pada Sistem Informasi Manajemen Proyek Dalam Industri & Organisasi Digital,” *Maj. Ilm. Teknol. Elektro*, vol. 19, no. 2, p. 203, 2020, doi: 10.24843/mite.2020.v19i02.p12.
- [8] K. Anwar, L. D. Kurniawan, M. I. Rahman, and N. Ani, “Aplikasi Marketplace Penyewaan Lapangan Olahraga Dari Berbagai Cabang Dengan Metode Agile Development,” *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 9, no. 2, pp. 264–274, 2020, doi: 10.32736/sisfokom.v9i2.905.
- [9] I. Rabbani, E. Krisnanik, and S. Kom, “E – Commerce Perlengkapan Haji Dan Umroh Berbasis Web Menggunakan Metode Agile Software Development,” *Pros. Semin. Nas. Mhs. Bid. Ilmu Komput. dan Apl.*, vol. 1, no. 2, pp. 432–443, 2020.
- [10] V. B. Gulo, A. Triayudi, and A. Iskandar, “Sistem Informasi Aplikasi Pemesanan Makanan Restoran Berbasis Web Menggunakan Metode Agile Development,” *J. Ris. Komputer*, vol. 10, no. 1, pp. 2407–389, 2023, doi: 10.30865/jurikom.v10i1.5633.
- [11] I. Rafiq, A. Mahmood, S. Razzaq, S. H. M. Jafri, and I. Aziz, “IoT applications and challenges in smart cities and services,” *J. Eng.*, vol. 2023, no. 4, pp. 1–25, 2023, doi: 10.1049/tje2.12262.
- [12] V. Sandeep, A. Sisay, M. Chala., and Fathi, “Automated use case diagram generation from textual user requirement documents.” pp. 1–4. doi: 10.1109/CCECE.2017.7946792.
- [13] T. Hong-jie, “Analysis and Design of Employment and Recruitment System based on UML.” pp. 399–403. doi: 10.2991/JIMEC-16.2016.71.
- [14] S. Martina, S. Marion, H. Christian, and K. Gerti, “The Use Case Diagram.” pp. 23–47. doi: 10.1007/978-3-319-12742-2_3.
- [15] M. Erwan, G. Christophe, L. Pascale, and Gall, “A structural operational semantics for interactions with a look at loops,” *Form. Lang. Autom. Theory*.
- [16] T. Arianti, A. Fa’izi, S. Adam, and M. Wulandari, “Perancangan Sistem Informasi Perpustakaan Menggunakan Diagram Uml (Unified Modelling Language),” *J. Ilm. Komput. Tera[an dan Inf.*, vol. 1, no. 1, pp. 19–25, 2022, [Online]. Available: <https://journal.polita.ac.id/index.php/politati/article/view/110/88>
- [17] A.-F. Sabah, *UML Sequence Diagram: An Alternative Model*. arXiv: Software Engineering.
- [18] A. Mohammad, M. Haris, M. Sajjad, and N. Mahmood, “Improving the Security of UML Sequence Diagram Using Genetic Algorithm,” *IEEE Access*, vol. 8, pp. 62738–62761, doi: 10.1109/ACCESS.2020.2981742.
- [19] O. Toshitaka, O. Takayuki, and O. Atsushi, “Supporting Change Management of Sequence Diagrams.” pp. 35–46. doi: 10.1007/978-3-030-53949-8_4.
- [20] A. Andriyadi, D. Yuliawati, S. Saleh, and B. Bachry, “Implementing White Box Testing for Evaluating the Inner

- Logic Code of the Research, Staffs, and Library Information System of Institute of Informatics and Business Darmajaya,” ... *Inf. ...*, no. December, pp. 178–182, 2020.
- [21] S. Emi and Eriana, “Pengujian sistem informasi aplikasi perpustakaan berbasis web dengan white box testing,” vol. 15, no. 2, pp. 28–33.
 - [22] M. F. Londjo, “Implementasi White Box Testing Dengan Teknik Basis Path Pada Pengujian Form Login,” *J. Siliwaangi*, vol. 7, no. 2, pp. 35–40, 2021.
 - [23] W. Ginanjar and Sasmito, “White Box Testing with Basis Path Technique in the Demography Administration Website.” doi: 10.1109/ICSECC51444.2020.9557428.
 - [24] M. Omar, A. Sallabi., B. R., and H. Mohamed, “A Cyclomatic Complexity Visual Tool For Simple Source Code Coverage,” vol. 7, no. 5, pp. 136–141. doi: 10.23956/IJARCSSE/SV7I5/0172.
 - [25] R. Chandrakant and Gujar, “Use and Analysis on Cyclomatic Complexity in Software Development,” *Int. J. Comput. Appl. Technol. Res.*, vol. 8, no. 5, pp. 153–156, doi: 10.7753/IJCATR0805.1002.
 - [26] S. Ronny, A. Ronggo, A. Fawwaz, and Akbar, “Pengujian white box pada sistem informasi monitoring skripsi program studi informatika,” vol. 1, no. 2, pp. 539–547.
 - [27] N. Shylaja and R. Vivek, “Why SRE documents matter,” *Commun. ACM*, vol. 16, no. 4, doi: 10.1145/3267122.
 - [28] S. Muneeb, sman U, and Waheed, “Documentation in Agile Development A Comparative Analysis.” doi: 10.1109/INMIC.2018.8595625.
 - [29] E. Jonathan, R. Wilson., S. J., S. John., and P. R., “Implant Delivery and Deployment System and Method.”
 - [30] D. David, J. Grewe., P. D., H. Kenneth, and B. David, “Lubrication Apparatus for a Delivery and Deployment Device.”