

Pengembangan Alat Pembangkit Skenario Pengujian Dari BPMN

Fikri Achmad Fauzi^{1*}, Novi Setiani²

^{1,2}Informatika, Universitas Islam Indonesia, Daerah Istimewa Yogyakarta, Indonesia

Email Penulis Korespondensi: ¹21523012@students.uii.ac.id

Abstrak– Penelitian ini membahas tentang pengembangan alat pembangkit skenario pengujian otomatis dari model Business Process Model and Notation (BPMN) dengan mengintegrasikan algoritma Depth-First Search (DFS) dan Large Language Model (LLM). Pengembangan dilakukan menggunakan metode pengembangan perangkat lunak secara linier, yang meliputi tahapan analisis kebutuhan, perancangan alat arsitektur, perancangan antarmuka, dan implementasi sistem. Sistem yang dibangun mengintegrasikan algoritma *Depth-First Search* (DFS) untuk menelusuri seluruh jalur proses dari start event hingga end event, serta *Large Language Model* (LLM) untuk menghasilkan skenario pengujian dalam bentuk naratif yang mencakup deskripsi, langkah pengujian, data uji, dan hasil yang diharapkan. Hasil implementasi menunjukkan bahwa sistem mampu menelusuri jalur proses secara menyeluruh dan menghasilkan skenario pengujian yang terstruktur, relevan dengan konteks model BPMN, serta mudah dipahami oleh pengguna. Fitur tambahan seperti visualisasi jalur pada diagram BPMN, riwayat hasil pengujian, dan ekspor laporan ke format Excel maupun PDF untuk meningkatkan efisiensi validasi dan dokumentasi pengujian. Namun, sistem masih menghadapi keterbatasan berupa fenomena *path explosion* pada model dengan banyak percabangan, peningkatan waktu pemrosesan pada model kompleks, dan variasi redaksi keluaran dari LLM yang masih belum konsisten. Oleh sebab itu, penelitian lanjutan perlu dilakukan untuk mengoptimalkan mekanisme eksplorasi jalur serta standarisasi hasil keluaran agar sistem dapat bekerja lebih efisien pada berbagai model BPMN yang sedang diuji.

Kata Kunci: Business Process Model and Notation, Depth-First Search, Pengujian BPMN, Large Language Model, Otomatisasi Pengujian

Abstract– This study discusses the development of an automatic test scenario generation tool from Business Process Model and Notation (BPMN) models by integrating the Depth-First Search (DFS) algorithm and a Large Language Model (LLM). The development was carried out using a linear software development method, which includes the stages of requirements analysis, architectural design, interface design, and system implementation. The system integrates the DFS algorithm to explore all process paths from the start event to the end event, and utilizes the LLM to generate narrative test scenarios that include descriptions, test steps, test data, and expected results. The implementation results show that the system can comprehensively traverse process paths and produce structured test scenarios that are relevant to the BPMN model context and easily understood by users. Additional features such as path visualization on BPMN diagrams, test history management, and report export in Excel and PDF formats improve the efficiency of validation and test documentation. However, the system still faces several limitations, including the path explosion phenomenon in models with numerous branches, increased processing time in complex models, and inconsistencies in the narrative output generated by the LLM. Therefore, further research is needed to optimize the path exploration mechanism and standardize the generated outputs so that the system can operate more efficiently across various BPMN models under testing.

Keywords: Business Process Model and Notation (BPMN), Depth-First Search (DFS), BPMN Testing, Large Language Model (LLM), Test Automation

1. PENDAHULUAN

Business Process Model and Notation (BPMN) telah menjadi standar global untuk melakukan pemodelan proses bisnis karena kemampuannya yang dapat merepresentasikan alur kerja secara grafis yang mudah dipahami oleh pemangku kepentingan teknis maupun non-teknis[1]. Adopsi BPMN yang semakin meluas, didorong oleh pertumbuhan tahunan gabungan pasar manajemen proses bisnis (BPM) global sebesar 19,9%, menunjukkan minat yang tinggi dalam meningkatkan efisiensi dan efektivitas alur bisnis[2]. Selain itu, dalam beberapa tahun terakhir, permintaan terhadap otomatisasi proses bisnis meningkat secara signifikan. Sebanyak 86% pengambil keputusan di bidang TI dan bisnis menilai otomatisasi proses sebagai faktor penting dalam transformasi digital organisasi, sementara lebih dari 90% perusahaan melaporkan peningkatan efisiensi operasional berkat penerapan orkestrasi proses bisnis[3]. Penerapan BPMN mengalami peningkatan yang signifikan dalam pemodelan proses bisnis. Hal ini dikarenakan BPMN memiliki kapabilitas untuk mendukung efisiensi operasional dan komunikasi lintas tim[4]

Meskipun BPMN banyak digunakan, pengujian model yang dilakukan secara manual masih menghadapi berbagai kendala. Metode manual cenderung tidak efisien, memakan waktu, rentan terhadap kesalahan manusia, membutuhkan sumber daya yang besar, dan sulit diterapkan pada model yang kompleks[5]. Hasil penelitian lain juga menunjukkan bahwa ketiadaan otomatisasi dalam pengujian BPMN dapat menurunkan kualitas validasi hasil pengujian[6]. Oleh karena itu, otomatisasi pengujian dipandang sebagai solusi yang dapat meningkatkan efektivitas sekaligus menurunkan biaya pengujian. Berbagai upaya untuk melakukan otomatisasi pengujian pada model BPMN telah dilakukan. Kerangka kerja bPERFECT, misalnya yang mengkalifikasikan berbagai pendekatan pengujian BPMN berdasarkan metode seperti *Black Box*, *Integration Testing*, dan *Regression Testing*, namun sebagian besar penelitian tersebut masih berfokus pada transformasi model dan belum mengeksplorasi jalur proses BPMN secara menyeluruh[7]. Selain itu, terdapat pula

pengembangan kerangka kerja yang dapat secara langsung menurunkan kasus uji dari BPMN yang dapat dieksekusi, sehingga mengurangi beban pengembang dalam menyusun skenario pengujian secara manual[8]. Pendekatan lain juga mengusulkan pembangkitan skenario pengujian secara otomatis dengan memanfaatkan *Control Flow Graph* (CFG) untuk menelusuri jalur eksekusi dan menghasilkan kasus uji berdasarkan kriteria *branch coverage*. Metode ini terbukti efektif dalam meningkatkan cakupan pengujian dan efisiensi verifikasi, namun masih terbatas dalam menangani elemen – elemen seperti *inclusive gateway* dan *message flow*[9].

Untuk dapat membangkitkan skenario pengujian dari model BPMN, maka diperlukan mekanisme eksplorasi jalur eksekusi dari model yang dikembangkan. Algoritma *Depth-First Search* (DFS) dipilih karena dinilai efektif dalam pembuatan skenario pengujian karena mampu menelusuri jalur dari *start event* hingga *end event* secara menyeluruh, serta memastikan tidak ada jalur dari model BPMN yang terlewat[10]. Jalur yang diperoleh dari proses DFS selanjutnya diproses oleh *Large Language Models* (LLM) untuk membangkitkan skenario pengujian yang lebih naratif, mudah dipahami, serta dilengkapi dengan data masukan dan hasil yang diharapkan[11].

Selain itu, Penelitian yang lain juga memperkenalkan pemanfaatan *Large Language Model* (LLM) untuk membangkitkan model BPMN secara otomatis dari deskripsi teks alami. Sistem ini mampu memahami konteks aktivitas bisnis dan menghasilkan model BPMN yang sesuai, namun masih berfokus pada pembuatan model, bukan pada proses pengujian atau validasi otomatis[12]. Penelitian lain juga melakukan upaya peningkatan kualitas pengujian melalui pemanfaatan metamodels yang berfungsi untuk menjaga konsistensi transformasi dari BPMN ke skenario pengujian serta mendukung interoperabilitas antar sistem pengujian[13]. Di sisi lain, penerapan verifikasi formal BPMN juga diperkenalkan untuk mendeteksi kesalahan logika sebelum eksekusi pada model BPMN[14]. LLM memiliki potensi signifikan untuk meningkatkan pemodelan proses bisnis berbasis BPMN dengan menyederhanakan kompleksitas pemetaan proses serta menghasilkan dokumentasi yang lebih adaptif dan terotomatisasi[15].

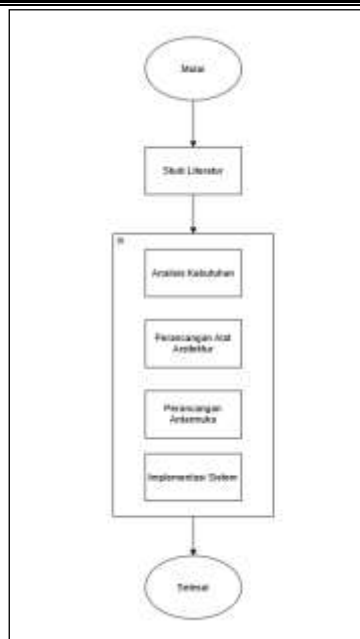
Berdasarkan berbagai kajian tersebut, dapat dilihat bahwa sebagian besar pendekatan pengujian otomatis berbasis BPMN masih berfokus pada transformasi model, verifikasi formal, atau pembuatan kasus uji dari struktur alur tanpa mempertimbangkan eksplorasi jalur eksekusi yang lebih mendalam. Pendekatan berbasis *Control Flow Graph* (CFG) misalnya, mampu menghasilkan skenario pengujian otomatis, namun belum secara eksplisit mengintegrasikan algoritma eksplorasi jalur yang adaptif seperti *Depth-First Search* (DFS). Selain itu, penelitian yang memanfaatkan *Large Language Model* (LLM) baru sebatas menghasilkan atau memahami model BPMN dari deskripsi teks alami, belum digunakan untuk membentuk skenario pengujian yang kontekstual dan dilengkapi dengan data uji maupun hasil yang diharapkan.

Dengan demikian, masih terdapat kesenjangan penelitian pada integrasi antara mekanisme eksplorasi jalur berbasis DFS dan kemampuan generatif LLM dalam membangun skenario pengujian otomatis yang komprehensif. Penelitian ini berupaya menutup kesenjangan tersebut melalui pengembangan alat pembangkit skenario pengujian dari BPMN yang dapat membaca model BPMN, mengekstraksi seluruh jalur eksekusi menggunakan DFS, serta memanfaatkan LLM untuk menghasilkan skenario pengujian yang lengkap, konsisten, dan mudah dipahami oleh pengguna dari berbagai peran teknis maupun non-teknis.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Penelitian ini menggunakan metode pengembangan perangkat lunak dengan pendekatan secara linier, karena sesuai untuk membangun alat yang tidak hanya memverifikasi model BPMN tetapi juga menghasilkan skenario pengujian yang terstruktur. Penelitian dilaksanakan pada periode Oktober 2024 - September 2025 secara daring, dengan kegiatan pengumpulan data lapangan melalui wawancara online menggunakan Zoom serta konstruksi perangkat lunak yang dilakukan di lingkungan pengembangan lokal dengan dukungan server basis data PostgreSQL. Target dari penelitian ini adalah menghasilkan alat pembangkit skenario pengujian dari BPMN yang dapat digunakan oleh pembuat BPMN maupun yang menguji model BPMN, sementara subjek wawancara terdiri dari tiga profesi yaitu System Analyst, Business Development Analyst Officer, dan Product Manager, yang dipilih untuk menggali kebutuhan nyata pengguna terhadap alat ini.



Gambar 1. Alur Pengembangan Alat BPMN

2.1.1 Analisis Kebutuhan

Analisis kebutuhan dilakukan untuk mengidentifikasi dan merumuskan kebutuhan fungsional dan non – fungsional dari alat yang dikembangkan. Tahap ini bertujuan memastikan sistem yang dibangun benar – benar menjawab permasalahan pengguna serta sejalan dengan tujuan penelitian. Proses analisis dilakukan melalui dua pendekatan, yaitu kajian literatur terhadap penelitian dan alat pengujian BPMN yang telah ada, serta wawancara dengan berbagai praktisi yang berperan dalam analisis proses bisnis dan pengujian perangkat lunak.

Dari hasil analisis fungsional, diperoleh bahwa alat yang dikembangkan harus mampu membaca dan memproses berkas BPMN secara otomatis, mengekstraksi elemen penting seperti *task*, *event*, *gateway*, dan *sequence flow*, serta menelusuri seluruh jalur proses menggunakan algoritma *Depth-First Search* (DFS) untuk menghasilkan skenario pengujian secara otomatis. Selain itu, sistem diharapkan dapat menampilkan hasil pengujian dalam bentuk langkah – langkah pengujian yang mudah dipahami, menyediakan data pengujian secara otomatis, dan menghasilkan laporan dalam format Excel maupun PDF untuk kepentingan dokumentasi.

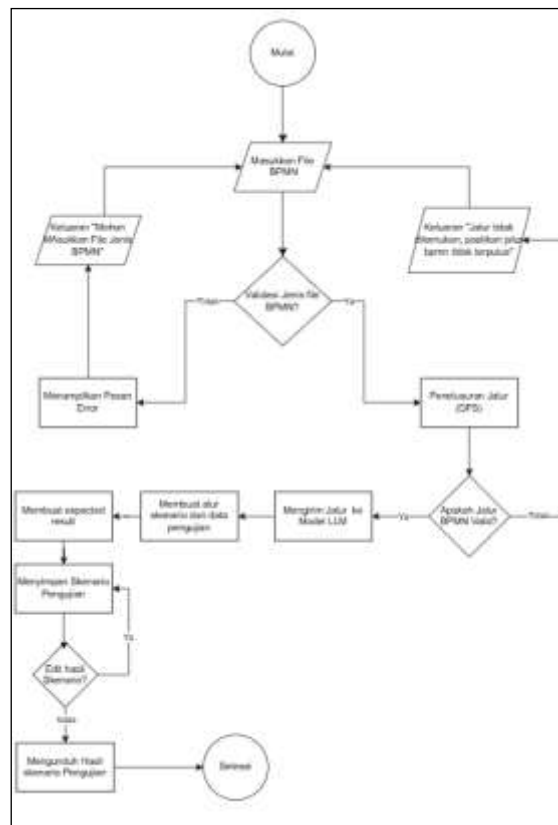
Sementara itu, dari sisi non-fungsional, sistem diharapkan memiliki fleksibilitas tinggi terhadap perubahan perubahan model bisnis yang sering terjadi, terutama ketika pengguna melakukan penyesuaian proses dalam setiap pertemuan atau iterasi pengembangan. Dengan demikian, sistem diharapkan mampu menyesuaikan perubahan model BPMN tanpa memerlukan konfigurasi ulang yang rumit. Selain itu, efisien dalam pengelolaan dokumentasi juga menjadi aspek yang penting. Sistem perlu mampu mempercepat proses pembuatan laporan hasil pengujian serta menyimpan hasil tersebut secara otomatis agar setiap perubahan dapat dilacak dengan mudah. Fitur riwayat pengujian menjadi kebutuhan yang dinilai penting untuk menjaga konsistensi data ketika terjadi pembaruan kebutuhan atau integrasi dengan sistem lain, sehingga mendukung efektivitas dan akurasi dalam proses pengujian berbasis BPMN.

2.1.2 Perancangan Alat Arsitektur

Berdasarkan hasil analisis kebutuhan, sistem ini dirancang untuk mendukung proses otomatisasi pengujian berbasis diagram BPMN. Arsitektur yang dikembangkan terdiri dari sejumlah komponen inti yang terintegrasi dan saling mendukung. Tahap awal dimulai dengan pemrosesan dan identifikasi diagram BPMN, di mana sistem membaca file BPMN yang diunggah pengguna, kemudian mengekstraksi elemen-elemen penting seperti *task*, *event*, dan *gateway*. Elemen tersebut diklasifikasikan sebagai dasar untuk analisis lebih lanjut. Selanjutnya dilakukan analisis jalur menggunakan algoritma *Depth-First Search* (DFS). Algoritma ini digunakan untuk menelusuri seluruh kemungkinan alur proses dari titik awal (*start event*) hingga titik akhir (*end event*). Dengan cara ini, sistem dapat mencakup seluruh kombinasi urutan proses dalam model BPMN dan menghasilkan skenario pengujian yang lebih representatif. Tahap berikutnya adalah pemanfaatan model *Large Language Model* (LLM). Jalur proses hasil analisis DFS, beserta konteks diagram BPMN, dikirimkan ke model LLM melalui integrasi API untuk menghasilkan deskripsi skenario pengujian yang mudah dipahami. Model ini juga membantu membangkitkan data uji untuk jalur yang dilalui oleh DFS, termasuk variabel masukan dan hasil yang diharapkan, yang kemudian disusun dalam format terstruktur sehingga dapat digunakan sebagai acuan validasi proses bisnis.

Agar data dapat diolah dan diakses dengan efisien, sistem menerapkan penyimpanan berbasis JSONB pada PostgreSQL. Seluruh hasil parsing, seperti elemen, jalur, maupun skenario pengujian, disimpan dalam format JSONB sehingga mendukung fleksibilitas akses, kemudahan integrasi antar modul, serta efisiensi kueri data. Untuk mendukung

komunikasi antara backend dan antarmuka pengguna, sistem menerapkan integrasi berbasis REST API. Setiap proses utama, mulai dari unggah file BPMN, pemrosesan, hingga penyajian skenario pengujian, diakses melalui endpoint API, sehingga pengelolaan data lebih terstruktur dan fleksibel. Selain itu, sistem juga menyediakan fitur visualisasi jalur uji dalam diagram BPMN. Jalur hasil analisis ditampilkan langsung dengan penyorotan (highlight) pada elemen terkait seperti *task*, *gateway*, dan *event* menggunakan warna berbeda. Visualisasi ini mempermudah pengguna dalam memahami alur proses dan memvalidasi skenario pengujian secara intuitif. Secara keseluruhan, rancangan arsitektur digambarkan melalui diagram alir yang menampilkan hubungan antar modul, mulai dari unggah file, analisis jalur, hingga penyusunan dan penyimpanan skenario. Pendekatan modular ini dibuat agar sistem tetap fleksibel, mudah dikembangkan, dan dapat diadaptasi pada kebutuhan di masa mendatang.



Gambar 2. Diagram Alir Proses Alat BPMN

2.1.3 Perancangan Antarmuka

Perancangan antarmuka pada alat ini mencakup beberapa komponen utama yang menggambarkan keseluruhan proses kerja sistem. Antarmuka dirancang sebagai aplikasi berbasis web yang memungkinkan pengguna melakukan seluruh tahapan, mulai dari mengunggah berkas BPMN hingga memperoleh skenario pengujian secara terstruktur. Rancangan antarmuka tersebut terdiri atas beberapa komponen utama sebagai berikut:

- Menu autentikasi pengguna, menu ini berfungsi untuk melakukan proses masuk dan pendaftaran akun, sehingga setiap aktivitas pengujian tercatat sesuai identitas pengguna.
- Menu unggah file BPMN, pada menu ini memungkinkan pengguna mengunggah berkas model BPMN yang kemudian dianalisis menggunakan algoritma *Depth-First Search (DFS)* untuk menelusuri seluruh jalur proses.
- Menu pembuatan skenario pengujian, pada bagian ini memanfaatkan hasil analisis jalur dari DFS yang nantinya akan dikirim ke model LLM agar menghasilkan skenario pengujian secara otomatis, yang mencakup deskripsi, langkah pengujian, data uji, dan hasil yang diharapkan.
- Menu hasil alur skenario pengujian, pada menu ini menampilkan hasil dari skenario pengujian dalam bentuk teks terstruktur serta visualisasi diagram BPMN yang menyoroti elemen jalur proses sesuai hasil analisis.
- Menu riwayat file BPMN, menyimpan riwayat aktivitas pengguna, termasuk berkas model yang telah diunggah, sehingga memudahkan pengguna melihat kembali hasil pengujian sebelumnya.
- Fitur ekspor hasil, pada bagian ini menyediakan opsi untuk mengunduh hasil pengujian dalam format PDF atau Excel sebagai dokumentasi dan validasi tambahan.

Seluruh komponen antarmuka terhubung dengan layanan *backend* melalui *REST API*, yang memastikan pertukaran data antara sisi pengguna dan server berjalan secara efisien dan terkoordinasi. Rancangan ini memungkinkan proses

analisis, pembangkitan skenario pengujian, dan visualisasi hasil dilakukan secara terpadu dalam satu lingkungan kerja berbasis web.

2.1.4 Implementasi Sistem

Implementasi sistem dilakukan berdasarkan rancangan arsitektur yang telah ditetapkan sebelumnya, dengan fokus pada pengembangan fitur – fitur utama yang mendukung proses otomatisasi pengujian berbasis BPMN. Proses pengembangan dilakukan secara linier, di mana setiap fungsionalitas diimplementasikan dalam modul – modul yang saling terhubung untuk memudahkan proses integrasi, pengujian, serta pengembangan lanjutan.

a. Pemrosesan Model BPMN

Sistem menerima masukan berupa berkas dengan ekstensi *.bpmn* yang diproses menggunakan pustaka *Camunda Model API*. Proses ini mengekstraksi elemen – elemen dalam model seperti *Start Event*, *Task*, *Exclusive Gateway*, *End Event*, dan *SequenceFlow*. Seluruh elemen yang terdeteksi disimpan dalam struktur data berbasis objek guna memudahkan analisis lebih lanjut pada tahap penelusuran jalur proses.

b. Implementasi Algoritma *Depth-First Search* (DFS)

Penelusuran jalur dilakukan menggunakan algoritma *Depth-First Search* (DFS) untuk mendeteksi seluruh kemungkinan jalur dari *Start Event* hingga *End Event* dalam model BPMN. Algoritma diimplementasikan secara rekursif dengan memanfaatkan struktur data *Linked List* sebagai penyimpanan jalur sementara, sehingga setiap kemungkinan jalur dapat ditelusuri secara sistematis tanpa kehilangan konteks urutan proses.

c. Integrasi dengan *Large Language Model*(LLM)

Hasil analisis jalur proses yang diperoleh dari algoritma DFS selanjutnya dikirim ke LLM untuk menghasilkan skenario pengujian secara otomatis. Model ini mengubah hasil penelusuran menjadi deksripsi pengujian, langkah pengujian, data uji, serta hasil yang diharapkan. Seluruh hasil keluaran tersebut disusun dalam format JSON agar mudah ditampilkan dan digunakan sebagai bagian dari dokumen pengujian.

d. Visualisasi dan Ekspor Hasil

Sistem menyediakan fitur visualisasi interaktif untuk menampilkan diagram BPMN beserta jalur proses yang terdeteksi. Elemen – elemen yang dilalui oleh algoritma DFS diberi penyorotan warna yang berbeda untuk memudahkan pengguna dalam proses validasi pengujian. Selain itu, sistem dilengkapi dengan fitur ekspor hasil pengujian ke dalam format PDF atau Excel, sehingga pengguna dapat menyimpan dan mendokumentasikan hasil analisis secara efisien.

2.2 Alat dan Bahan Penelitian

Penelitian ini menggunakan sejumlah alat dan bahan yang berfungsi mendukung proses pengembangan sistem. Alat untuk melakukan proses wawancara secara daring adalah dengan menggunakan *Zoom* sebagai media komunikasi utama, sedangkan *Google Docs* digunakan untuk mencatat, menyusun, dan menyimpan hasil wawancara. Bahan penelitian meliputi beberapa model BPMN dengan tingkat kompleksitas berbeda yang digunakan untuk menguji kemampuan alat dalam menelusuri jalur proses dan menghasilkan skenario pengujian.

Pada tahap pengembangan, sistem ini dibangun menggunakan pemrograman berbasis web yang mendukung proses pemodelan, analisis, dan penyimpanan data hasil pengujian BPMN. Pendekatan ini dipilih agar sistem mudah diakses dan dikembangkan lebih lanjut. Hasil wawancara dengan praktisi seperti System Analyst, Business Development Analyst Officer, dan Product Manager digunakan sebagai acuan dalam menentukan kebutuhan fungsional dan non-fungsional. Temuan dari wawancara tersebut juga menjadi dasar dalam menyesuaikan rancangan sistem dengan kebutuhan pengguna di lapangan serta memastikan hasil pengujian yang dihasilkan relevan dengan praktik industri.

3. HASIL DAN PEMBAHASAN

3.1 Hasil Penelitian Sebelumnya

Pendekatan verifikasi terhadap model BPMN telah banyak dikaji untuk meningkatkan keakuratan dan konsistensi proses bisnis. Salah satu pendekatan yang banyak digunakan adalah metode berbasis Control Flow Graph (CFG) dan Colored Petri Net (CPN) untuk melakukan analisis dan verifikasi otomatis[9]. Penelitian tersebut mampu mencapai branch coverage secara penuh dan mendeteksi adanya deadlock dan jalur yang tidak dapat dijangkau (unreachable tasks) pada model proses bisnis. Implementasi analisis CPN dilakukan menggunakan CPN Tools, dan hasilnya menunjukkan bahwa proses verifikasi dapat dilakukan secara konsisten tanpa intervensi manual. Namun, hasil keluaran yang diperoleh masih terbatas pada status verifikasi formal tanpa menghasilkan skenario pengujian yang deskriptif, sehingga pengguna tidak memperoleh gambaran langkah pengujian yang dapat diterapkan secara langsung.

Selain itu, penelitian lain juga membahas pendekatan pengujian dan verifikasi model BPMN untuk meningkatkan akurasi serta keandalan proses bisnis[7]. Penelitian ini menghimpun dan membandingkan beragam metode yang telah digunakan dalam pengujian BPMN, mulai dari pendekatan berbasis graf hingga transformasi model formal seperti *petri net* dan *automata*. Hasil tinjauan menunjukkan bahwa sebagian besar penelitian sebelumnya masih berfokus pada validasi struktur model dan pengujian regresi, sedangkan dukungan terhadap otomatisasi penuh serta penerapan konsep *continuous testing* masih terbatas. Berdasarkan analisis tersebut, penelitian tersebut mengusulkan *bPERFECT framework*

sebagai kerangka konseptual untuk mendukung otomatisasi pengujian BPMN secara berkelanjutan pada tahap pengembangan perangkat lunak.

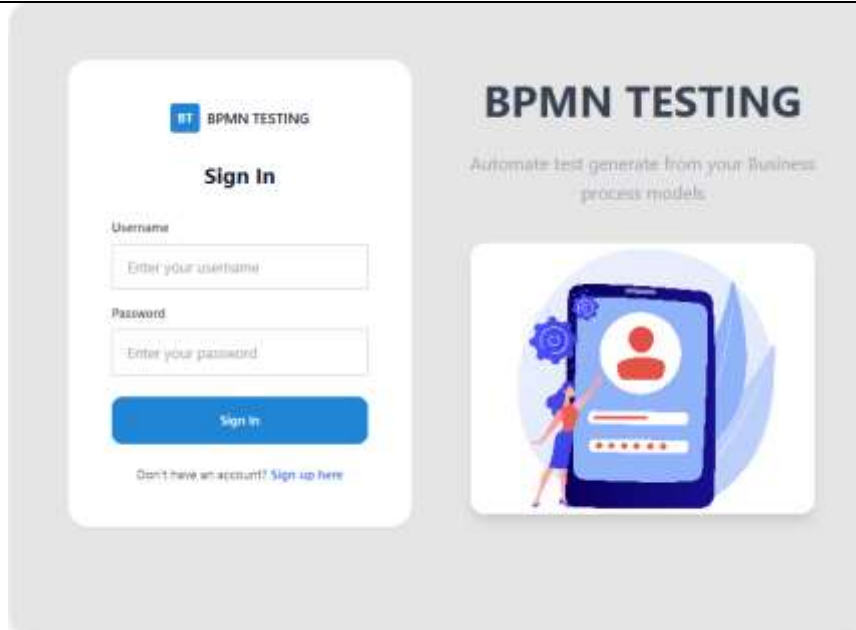
Di sisi lain, pendekatan berbasis *Large Language Model* (LLM) juga telah digunakan untuk memperluas otomatisasi dalam pengujian perangkat lunak [12]. Pada penelitian tersebut model yang digunakan, yaitu Llama-2 7B dengan metode QLoRA quantization, diterapkan untuk menghasilkan unit test secara otomatis dari struktur program. Hasil pengujian menunjukkan nilai F1-score sebesar 33,95 %, yang mengindikasikan efektivitas penerapan LLM dalam proses pengujian berbasis kode. Pendekatan ini menunjukkan adanya pergeseran dari metode pengujian berbasis aturan menuju pendekatan yang memanfaatkan kecerdasan generatif. Meskipun demikian, penerapan LLM tersebut masih terbatas pada pengujian berbasis kode program dan belum diterapkan pada konteks pemodelan proses bisnis seperti BPMN, sehingga belum mampu menghasilkan skenario pengujian yang mempresentasikan alur proses secara utuh dan mudah dipahami oleh pengguna.

Sistem yang dikembangkan dalam penelitian ini menghasilkan proses pembangkitan skenario pengujian otomatis dari model BPMN secara menyeluruh. Melalui algoritma DFS, sistem mampu menelusuri seluruh jalur proses bisnis dari awal hingga akhir dan menghasilkan keluaran berupa skenario pengujian terstruktur yang berisi deskripsi, langkah pengujian, data pengujian, dan hasil yang diharapkan. Selain itu, integrasi dengan LLM memungkinkan sistem menyusun narasi pengujian secara kontekstual sesuai alur proses, sehingga hasilnya lebih mudah dipahami oleh pengguna. Fitur tambahan seperti penyimpanan riwayat hasil pengujian dan ekspor laporan dalam format Excel dan PDF diterapkan untuk mempercepat proses validasi model BPMN dan mendukung pengujian secara berkelanjutan.

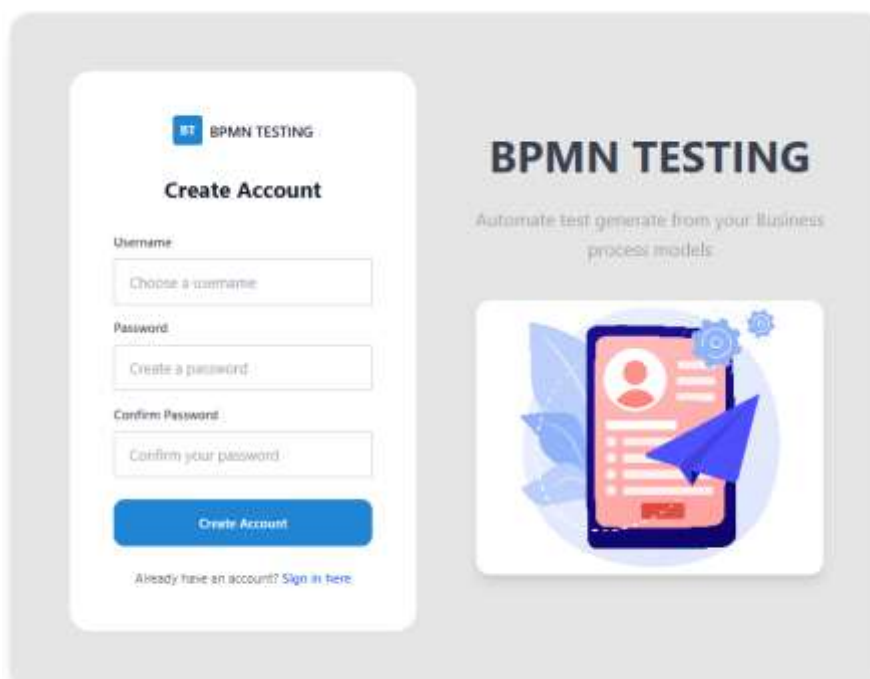
3.2 Hasil Implementasi Sistem

Implementasi sistem dibuat sesuai dengan rancangan arsitektur yang telah disusun pada tahap sebelumnya. Pada bagian ini berfokus pada pengembangan fitur – fitur utama yang saling terintegrasi untuk mendukung fungsionalitas dalam pembangkitan skenario pengujian dari model BPMN secara otomatis. Pada tahap awal implementasi melibatkan pengembangan modul untuk membaca dan mengurai berkas BPMN dengan format (.bpnm). Selanjutnya, alat ini menggunakan pustaka Camunda Model API untuk mengekstraksi elemen-elemen dari BPMN, seperti *Start Event*, *End Event*, *Task*, *Exclusive Gateway*, *Sub Proses*, *Message Flow*, dan *Sequence Flow*. Setiap elemen yang diekstraksi kemudian direpresentasikan dalam struktur data berbasis objek untuk memfasilitasi analisis lanjutan dan manipulasi data yang efisien. Tahap selanjutnya adalah implementasi DFS yang digunakan untuk mengeksplorasi seluruh jalur eksekusi pada model BPMN dari *Start Event* hingga *End Event*. Algoritma ini berfungsi menelusuri jalur secara rekursif agar semua kombinasi alur tercakup, sehingga memastikan tidak ada kombinasi alur proses yang terlewat. Jalur-jalur yang ditemukan selama proses penelusuran disimpan dalam struktur data untuk menjaga urutan dan memudahkan pemrosesan berikutnya.

Hasil penelusuran jalur DFS, beserta konteks diagram BPMN, akan dikirimkan ke LLM melalui integrasi API. Model api yang digunakan ialah GPT4o yang nantinya model tersebut akan menghasilkan narasi deskriptif untuk skenario pengujian, lengkap dengan data pengujian dan hasil yang diharapkan (*expected result*). Hasil keluaran disusun dalam format JSON terstruktur agar dapat digunakan kembali untuk dokumentasi maupun analisis. Untuk mendukung proses pembangkitan skenario pengujian dari model BPMN secara intuitif dan terstruktur. Mekanisme autentikasi pengguna digunakan melalui fitur login dan registrasi, yang tidak hanya berfungsi sebagai kontrol akses, tetapi juga memungkinkan setiap pengguna menyimpan riwayat file BPMN dan hasil pengujian yang telah dilakukan. Setelah berhasil masuk, pengguna dapat mengunggah berkas BPMN dengan format .bpnm untuk diproses oleh sistem.

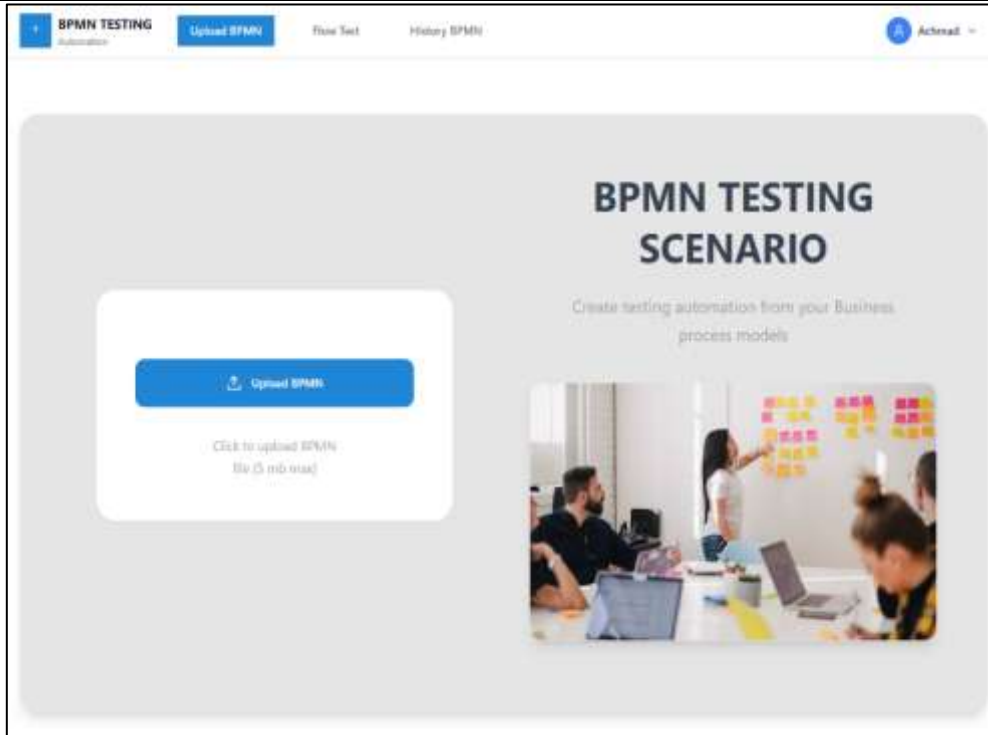


Gambar 3. Menu Login

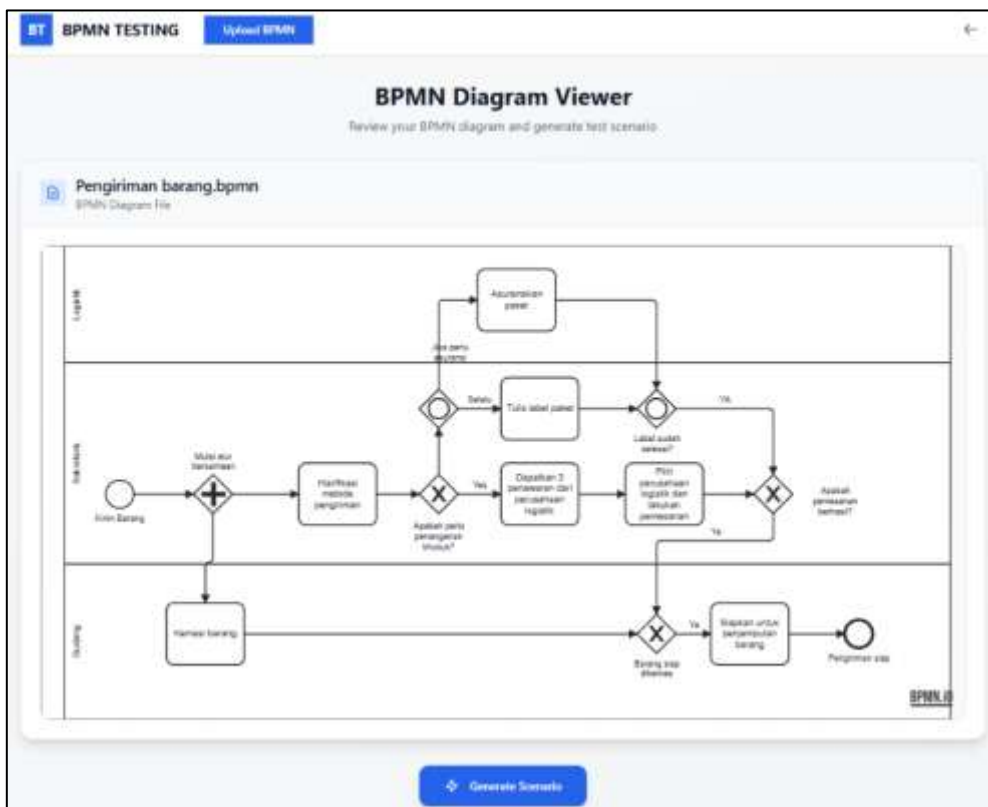


Gambar 4. Menu Register

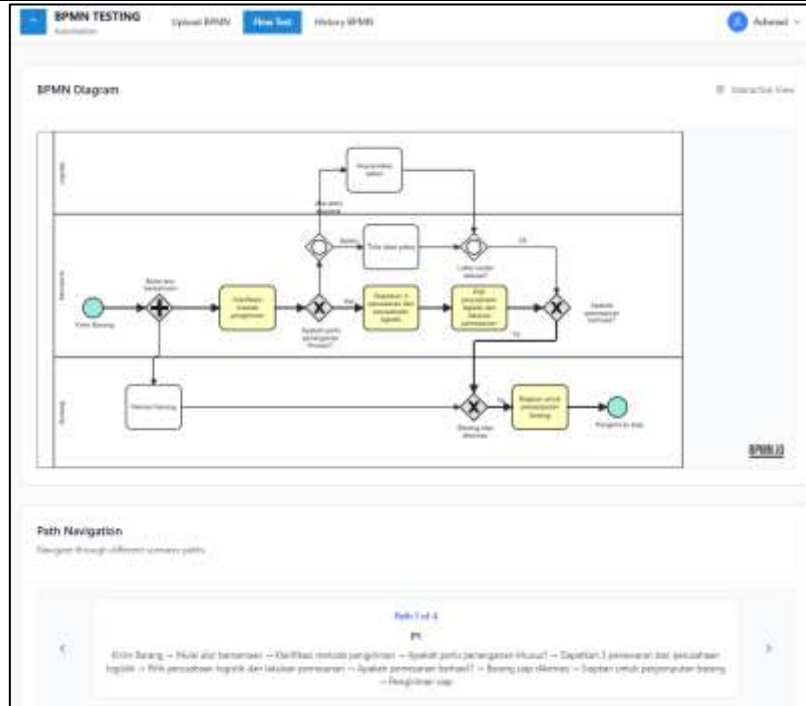
Model BPMN yang diproses kemudian akan divisualisasikan secara interaktif menggunakan pustaka *bpmn-js*. Visualisasi ini mempertahankan struktur asli diagram, namun ditambahkan fitur penyorotan (highlight) pada jalur pengujian yang dilewati hasil analisis Depth-First Search (DFS). Warna berbeda diterapkan untuk membedakan setiap jalur, dan navigasi antarjalur dapat dilakukan secara mudah melalui kontrol interaktif. Dengan demikian, pengguna dapat memperoleh pemahaman menyeluruh mengenai semua kemungkinan alur eksekusi proses bisnis.



Gambar 5. Menu Unggah File BPMN



Gambar 6. Menu Pembuatan Skenario Pengujian

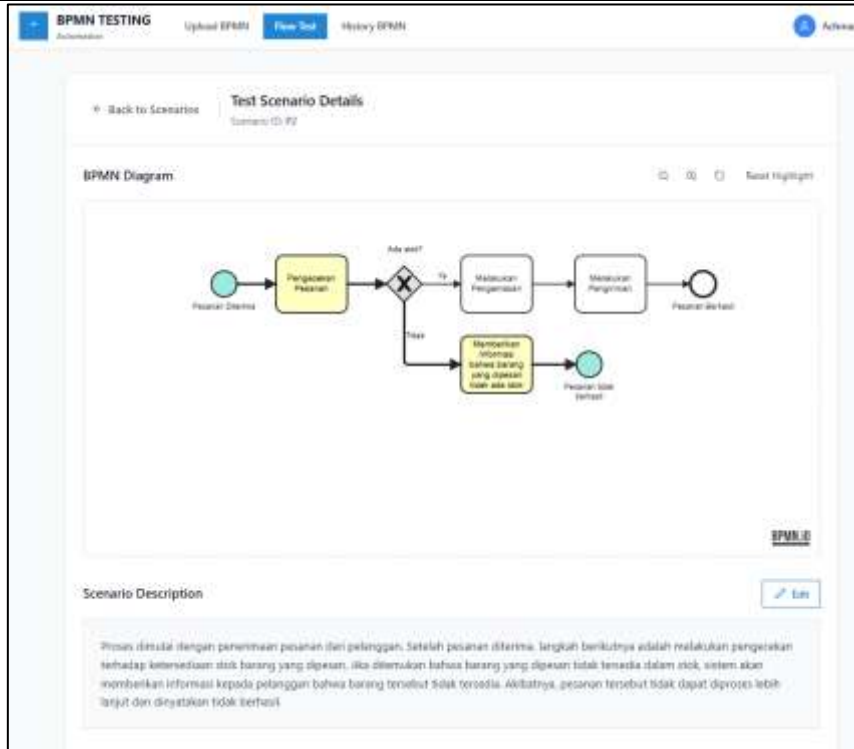


Gambar 7. Visualisasi Diagram BPMN Hasil Pengujian

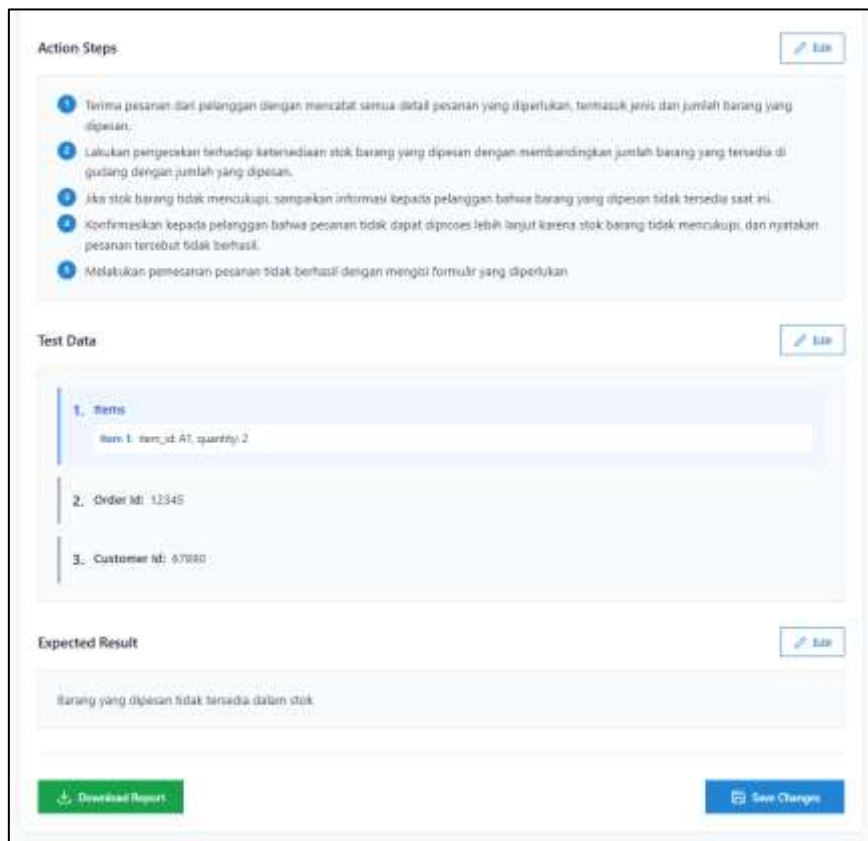
Selain visualisasi pada model BPMN, sistem juga menyajikan hasil dalam bentuk tabel skenario pengujian. Tabel ini berisi deskripsi naratif yang diolah oleh Large Language Model (LLM), data pengujian, serta hasil yang diharapkan (*expected result*) untuk setiap jalur. Penyajian dalam format tabel, sehingga hasil pengujian dapat dipahami baik oleh penguji maupun pemangku kepentingan non-teknis seperti analis bisnis atau manajer proyek. Untuk mendukung dokumentasi formal, tabel skenario pengujian dilengkapi dengan fitur ekspor ke format PDF atau Excel. Fitur ini memungkinkan hasil uji disimpan secara rapi, digunakan kembali pada tahap pengujian berikutnya, maupun dijadikan lampiran laporan eksternal.

PATH ID	SCENARIO OVERVIEW	STATUS	ACTIONS
P1	Proses pengiriman barang dimulai dengan sekretaris yang mengirimkan barang dan 10 steps	Pengiriman siap	Detail Download
P2	Proses pengiriman barang dimulai dengan sekretaris yang mengirim barang dan m 11 steps	Pengiriman siap	Detail Download
P3	Proses pengiriman barang dimulai dengan pengiriman oleh sekretaris yang memas 11 steps	Pengiriman siap	Detail Download
P4	Proses pengiriman barang dimulai dengan sekretaris yang mengirimkan barang ke 6 steps	Pengiriman siap	Detail Download

Gambar 8. Daftar Alur Pengujian BPMN

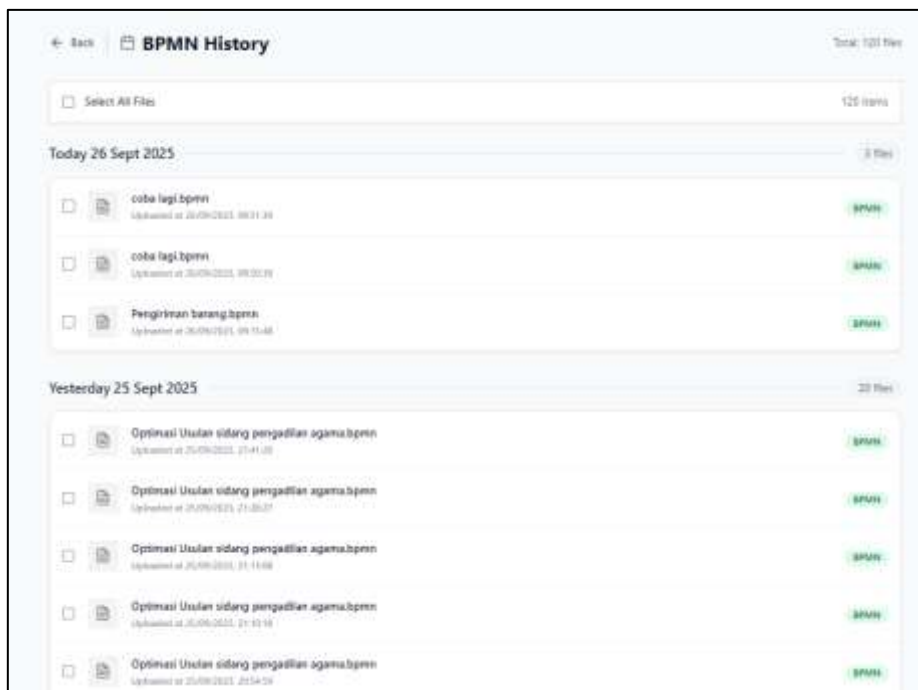


Gambar 9. Hasil deskripsi pengujian



Gambar 10. Hasil Langkah Pengujian dan Hasil yang Diharapkan

Kemudian, menu riwayat file BPMN berfungsi untuk menampilkan daftar model proses bisnis yang etlah diunggah dan diuji oleh pengguna sebelumnya. Komponen ini dirancang untuk mendukung efisiensi penggunaan sistem, di mana pengguna dapat mengakses kembali hasil pengujian tanpa perlu mengunggah ulang model yang sama. Selain itu, fitur ini juga memfasilitasi pengguna dalam melakukan perbandingan antarversi model BPMN, meninjau hasil skenario pengujian terdahulu, serta mengunduh kembali dokumen hasil uji yang diperlukan.



Gambar 11. Menu Riwayat File BPMN

Selain fitur riwayat yang memudahkan pengguna dalam meinjau kembali hasil pengujian sebelumnya, sistem juga menyediakan fitur ekspor hasil pengujian yang berfungsi untuk menghasilkan laporan pengujian dalam format terstruktur. Fitur ini menjadi tahap lanjutan setelah pengguna meninjau hasil pada riwayat pengujian, sehingga seluruh data yang telah tersimpan dapat diekspor dan dibagikan untuk keperluan dokumentasi maupun validasi eksternal. Hasil ekspor yang dihasilkan dalam format Excel maupun PDF disajikan dalam bentuk tabel terstruktur yang memuat seluruh informasi penting hasil pengujian. Setiap skenario pengujian direpresentasikan dalam satu baris, sedangkan kolom – kolom di dalamnya menampilkan detail pengujian yang dihasilkan oleh sistem.

Tabel 1. Deskripsi Kolom Hasil Pengujian BPMN

Kolom	Keterangan
Path ID	Menunjukkan identitas unik atau kode dari setiap jalur pengujian (contohnya P1, P2 dan seterusnya) yang mempresentasikan urutan proses BPMN yang diuji
Summary Step	Berisi ringkasan dari keseluruhan alur pengujian yang dihasilkan berdasarkan analisis jalur menggunakan algoitma DFS dan model LLM
Action Step	Menampilkan daftar langkah – langkah pengujian yang disusun secara berurutan sesuai dengan alur proses bisnis yang diuji
Test Data	Memuat nilai – nilai input yang digunakan selama proses pengujian, yang diperoleh dari hasil analisis jalur pengujian
Expected Result	Memuat hasil yang diharapkan dari pengujian terhadap jalur proses bisnis yang dianalisis
Actual Result	Merupakan kolom yang digunakan untuk mencatat hasil aktuan yang diperoleh saat pengujian dilakukan secara langsung oleh pengguna
Tester	Menampilkan nama pengguna atau penguji yang menjalankan proses pengujian

TEST SCENARIO REPORT						
File: BPM F11a Generated: Fri Sep 26 09:58:00 WIB 2025						
Path ID	Summary Step	Action Step	Data Uji	Expected Result	Actual Result	Tester
P2	Proses dimulai dengan penerimaan pesanan dari pelanggan. Setelah pesanan diterima, langkah berikutnya adalah melakukan pengecekan terhadap ketersediaan stok barang yang dipesan. Jika ditamakan bahwa barang yang dipesan tidak tersedia dalam stok, sistem akan memberikan informasi kepada pelanggan bahwa barang tersebut tidak tersedia. Akibatnya, pesanan tersebut tidak dapat diproses lebih lanjut dan dinyatakan tidak berhasil.	Step 1: Terima pesanan dari pelanggan dengan mencatat semua detail pesanan yang diperlukan, termasuk jenis dan jumlah barang yang dipesan. Step 2: Lakukan pengecekan terhadap ketersediaan stok barang yang dipesan dengan membandingkan jumlah barang yang tersedia di gudang dengan jumlah yang dipesan. Step 3: Jika stok barang tidak mencukupi, sampaikan informasi kepada pelanggan bahwa barang yang dipesan tidak tersedia saat ini. Step 4: Konfirmasikan kepada pelanggan bahwa pesanan tidak dapat diproses lebih lanjut karena stok barang tidak mencukupi, dan nyatakan pesanan tersebut tidak berhasil. Step 5: Melakukan pemesanan pesanan tidak berhasil dengan mengisi formulir yang diperlukan	Item: - Item 1: item_id: A1, quantity: 2 Order Id: 12345 Customer Id: 67890	Barang yang dipesan tidak tersedia dalam stok		Achmad

Gambar 12. Hasil Keluaran dari pengujian model BPMN

Secara keseluruhan, sistem yang dikembangkan telah berhasil menjalankan fungsinya secara efektif dalam menelusuri seluruh jalur proses bisnis pada model BPMN menggunakan algoritma DFS, serta menghasilkan skenario pengujian otomatis melalui integrasi dengan model LLM. Hasil yang diperoleh menunjukkan bahwa pendekatan ini mampu memberikan keluaran yang deksriptif, terstruktur, dan relevan terhadap konteks proses bisnis yang diuji. Meskipun demikian, penelitian ini masih memiliki beberapa keterbatasan yang perlu diperhatikan. Salah satu kendala utama adalah fenomena *path explosion*, yaitu peningkatan jumlah kombinasi jalur eksekusi secara eksponensial seiring dengan bertambahnya jumlah *gateway* dan percabangan pada model BPMN. Kondisi ini menyebabkan waktu pemrosesan meningkat secara signifikan pada model dengan kompleksitas tinggi, terutama ketika sistem harus mengeksplorasi seluruh kemungkinan jalur untuk mencapai cakupan penuh. Selain itu, variasi redaksi narasi yang dihasilkan oleh LLM masih belum sepenuhnya konsisten, meskipun isi langkah pengujian, data uji, dan hasil yang diharapkan tetap konsisten secara semantik. Keterbatasan lainnya terletak pada cakupan elemen BPMN yang belum sepenuhnya lengkap seperti *data object* atau *text annotation* yang belum diakomodasi secara penuh dalam proses penelusuran jalur

Untuk mengurangi dampak *path explosion* yang muncul pada model dengan banyak percabangan, sistem menerapkan mekanisme pembatasan eksplorasi jalur melalui pengaturan batas maksimum jumlah jalur dan waktu eksekusi. Pembatasan ini bertujuan untuk mencegah proses penelusuran berlangsung terlalu lama akibat meningkatnya kombinasi jalur yang bersifat eksponensial, sehingga hasil yang diperoleh tetap representatif dan sistem dapat beroperasi secara stabil. Meskipun demikian, mekanisme ini masih bersifat sementara dan belum sepenuhnya mengatasi kompleksitas *path explosion*. Dengan demikian, meskipun sistem ini telah efektif dalam mendukung proses penelusuran jalur dan pembangkitan skenario pengujian otomatis dari BPMN, diperlukan pengembangan lanjutan untuk mengatasi dampak *path explosion*, mengoptimalkan waktu pemrosesan pada model yang memiliki kompleksitas tinggi, menstandarkan hasil keluaran dari LLM agar lebih konsisten, serta memperluas dukungan terhadap elemen BPMN agar sistem dapat diterapkan secara lebih menyeluruh dan efisien.

4. KESIMPULAN

Alat pembangkit skenario pengujian secara otomatis dari model Business Process Model and Notation (BPMN) dikembangkan dengan mengintegrasikan algoritma Depth-First Search (DFS) dan Large Language Model (LLM) untuk mendukung otomatisasi proses bisnis. Mekanisme DFS digunakan untuk menelusuri seluruh jalur proses dari *start event* hingga *end event*, sehingga seluruh kemungkinan alur dapat teridentifikasi. Hasil penelusuran tersebut kemudian akan diolah oleh LLM menjadi skenario pengujian yang disajikan secara naratif dan terstruktur, yang mencakup deskripsi, langkah pengujian, data uji, serta hasil yang diharapkan. Implementasi sistem menunjukkan bahwa proses pembangkitan skenario pengujian dapat dilakukan secara otomatis dan menghasilkan keluaran yang sesuai dengan konteks model BPMN. Fitur tambahan seperti visualisasi jalur pada diagram BPMN, penyimpanan riwayat pengujian, serta ekspor laporan ke dalam format Excel dan PDF mendukung efisiensi validasi dan dokumentasi pengujian. Integrasi antara DFS dan LLM memberikan pendekatan yang adaptif terhadap perubahan model bisnis serta memperkaya aspek pemahaman hasil uji oleh berbagai pengguna.

Namun demikian, sistem ini masih memiliki beberapa keterbatasan, antara lain fenomena *path explosion* yang menyebabkan peningkatan jumlah jalur secara eksponensial pada model dengan banyak percabangan, peningkatan waktu pemrosesan pada model dengan kompleksitas tinggi, variasi redaksi hasil keluaran LLM yang belum sepenuhnya konsisten, serta cakupan elemen BPMN yang belum mencakup elemen kompleks. Oleh sebab itu, pengembangan lanjutan perlu diarahkan pada penerapan optimasi eksplorasi jalur untuk mengatasi *path explosion*, peningkatan efisiensi pemrosesan, standarisasi keluaran LLM, serta perluasan dukungan terhadap elemen BPMN agar sistem dapat beroperasi secara lebih optimal, stabil dan komprehensif.

UCAPAN TERIMAKASIH

Penulis mengucapkan terima kasih kepada para responden penelitian, yang terdiri dari praktisi di bidang System Analyst, Business Development Analyst Officer, dan Product Manager, atas waktu dan kontribusi mereka dalam memberikan data serta wawasan yang sangat membantu dalam pengembangan dari hasil penelitian alat ini.

REFERENCES

- [1] K. Schneid, L. Stapper, S. Thone, and H. Kuchen, "Automated Regression Tests: A No-Code Approach for BPMN-based Process-Driven Applications," in *Proceedings - 2021 IEEE 25th International Enterprise Distributed Object Computing Conference, EDOC 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 31–40. doi: 10.1109/EDOC52215.2021.00014.
- [2] Comidor Team, "10 Must-Know Business Process Management Statistics." Accessed: Jul. 30, 2025. [Online]. Available: <https://www.comidor.com/news/industry-news/bpm-statistics/>
- [3] Camunda and Regina Corso Consulting, "2024 State of Process Orchestration," 2024.
- [4] T. M. Paiva, T. C. Oliveira, R. M. Pillat, and P. S. C. Alencar, "Supporting the Automated Generation of Acceptance Tests of Process-Aware Information Systems," in *International Conference on Web Information Systems and Technologies, WEBIST - Proceedings*, Science and Technology Publications, Lda, 2023, pp. 128–139. doi: 10.5220/0012211400003584.
- [5] T. G. Lopes, "FlowTGE: Automating Functional Testing of Executable Business Process Models Based on BPMN."
- [6] Gopinath Kathiresan, "Automated Test Case Generation with AI: A Novel Framework for Improving Software Quality and Coverage," *World Journal of Advanced Research and Reviews*, vol. 23, no. 2, pp. 2880–2889, Aug. 2024, doi: 10.30574/wjarr.2024.23.2.2463.
- [7] Tomas Lopes and Sergio Guerreiro, "bpmj-11-2022-0557," Oct. 2023.
- [8] R. R. Riskiana, Ryan Oktaviandi Susilo Wibowo, and Arpriansah Yonathan, "FlowForge: A Prototype for Generating User Stories and Gherkin Test Cases from BPMN with DMN Integration and Pattern Matching," *International Journal on Information and Communication Technology (IJoICT)*, vol. 10, no. 2, pp. 195–213, Jan. 2025, doi: 10.21108/ijoint.v10i2.1015.
- [9] C. Dechsupa, W. Vatanawood, and A. Thongtak, "An automated framework for bpmn model verification achieving branch coverage," *Engineering Journal*, vol. 25, no. 2, pp. 135–150, 2021, doi: 10.4186/ej.2021.25.2.135.
- [10] P. Von Olberg and L. Strey, "Approach to Generating Functional Test Cases from BPMN Process Diagrams," in *Proceedings of the IEEE International Conference on Requirements Engineering*, IEEE Computer Society, 2022, pp. 185–189. doi: 10.1109/REW56159.2022.00042.
- [11] A. M. Sami, Z. Rasheed, M. Waseem, Z. Zhang, H. Tomas, and P. Abrahamsson, "A Tool for Test Case Scenarios Generation Using Large Language Models," Jun. 2024, [Online]. Available: <http://arxiv.org/abs/2406.07021>
- [12] S. Rehan, B. Al-Bander, and A. Al-Said Ahmad, "Harnessing Large Language Models for Automated Software Testing: A Leap Towards Scalable Test Case Generation," *Electronics (Switzerland)*, vol. 14, no. 7, Apr. 2025, doi: 10.3390/electronics14071463.
- [13] L. Leal, L. Montecchi, A. Ceccarelli, and E. Martins, "Using Metamodels to Improve Model-Based Testing of Service Orchestrations."
- [14] Y. Falcone, G. Salaün, and A. Zuo, "Probabilistic Runtime Enforcement of Executable BPMN Processes," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Science and Business Media Deutschland GmbH, 2024, pp. 56–76. doi: 10.1007/978-3-031-57259-3_3.
- [15] H. Kourani, A. Berti, D. Schuster, and W. M. P. van der Aalst, "Evaluating large language models on business process modeling: framework, benchmark, and self-improvement analysis," *Softw Syst Model*, 2025, doi: 10.1007/s10270-025-01318-w.