

# Risk Management Expert System for Application Development Using the Forward Chaining Method

## [Sistem Pakar Manajemen Risiko untuk Pengembangan Aplikasi Menggunakan Metode Forward Chaining]

Hendrawan Pudjianto<sup>1)</sup>, Arif Senja Fitriani <sup>\*,2)</sup>

<sup>1)</sup>Program Studi Teknik Informatika, Universitas Muhammadiyah Sidoarjo, Indonesia

<sup>2)</sup>Program Studi Teknik Informatika, Universitas Muhammadiyah Sidoarjo, Indonesia

\*Email Penulis Korespondensi: [asfjim@umsida.ac.id](mailto:asfjim@umsida.ac.id)

**Abstract.** *PT XYZ faces challenges in managing risks that affect the company's operations, growth and finances. These risks must be better managed through the implementation of appropriate and efficient mitigation strategies to ensure the effectiveness and sustainability of the company's business. Therefore, this research aims to design software using an expert system with the forward chaining method to manage risks. This research uses an expert system with the forward chaining method that has been successfully applied in agriculture and animal husbandry. Parameters considered include probability of occurrence, financial and operational impact, vulnerability, severity, existing controls, regulatory compliance, and reputational impact. Test results show the software is effective and provides appropriate risk management recommendations with an accuracy rate of 100%. The system functions well and has the potential to be applied more widely to improve risk management and operational efficiency.*

**Keywords** - Risk Management; Forward Chaining; Expert System

**Abstrak.** *PT XYZ menghadapi tantangan dalam pengelolaan risiko yang mempengaruhi operasional, pertumbuhan, dan keuangan perusahaan. Risiko ini harus dikelola dengan lebih baik melalui implementasi strategi mitigasi yang tepat dan efisien untuk memastikan efektivitas dan keberlangsungan bisnis perusahaan. Oleh karena itu penelitian ini bertujuan merancang perangkat lunak menggunakan sistem pakar dengan metode forward chaining untuk mengelola risiko. Penelitian ini menggunakan sistem pakar dengan metode forward chaining yang telah berhasil diterapkan di bidang pertanian dan peternakan. Parameter yang dipertimbangkan meliputi probabilitas kejadian, dampak finansial dan operasional, kerentanan, keparahan, kontrol yang ada, kepatuhan regulasi, dan dampak reputasi. Hasil pengujian menunjukkan perangkat lunak ini efektif dan memberikan rekomendasi manajemen risiko yang tepat dengan tingkat akurasi mencapai 100%. Sistem ini berfungsi dengan baik dan berpotensi diterapkan lebih luas untuk meningkatkan manajemen risiko dan efisiensi operasional perusahaan.*

**Kata Kunci** - Manajemen Risiko; Forward Chaining; Sistem Pakar

## I. PENDAHULUAN

PT XYZ merupakan Perusahaan yang sudah lebih dari 50 tahun berbisnis dalam bisnis manufaktur. PT XYZ juga merupakan induk dari beberapa anak Perusahaan [1]. Di usia 50 tahunnya, inovasi menjadi hal mutlak harus selalu ada dalam operasional PT XYZ terutama inovasi dibidang teknologi. Dalam 3 tahun terakhir, PT XYZ telah melakukan digitalisasi di berbagai aspek lini operasional mereka, mulai dari aspek operasional *back office*, marketing atau pemasaran, keselamatan dan kesehatan kerja hingga menuju integrasi terpusat data Perusahaan dengan menggunakan *Enterprise Resource Planner* (ERP) [2]. Selama ini, PT XYZ menghadapi tantangan besar dalam mengelola risiko proyek yang sering kali belum akurat dan tidak efisien. Ketidakakuratan dalam identifikasi, penilaian, dan mitigasi risiko menyebabkan sejumlah proyek mengalami penundaan, biaya yang membengkak, dan bahkan kegagalan total, seperti halnya melakukan implementasi ERP yang tidak sesuai dengan proses bisnis yang ada [3]. Hal ini menyoroti perlunya sebuah pendekatan yang lebih cerdas dan terstruktur dalam mengelola risiko proyek. Salah satu solusi yang potensial adalah pembangunan Sistem Pakar untuk memberikan rekomendasi yang tepat dan akurat dalam pengelolaan risiko. Sistem Pakar ini akan menggunakan teknik-teknik kecerdasan buatan, seperti pembelajaran mesin dan basis pengetahuan yang mendalam, untuk menganalisis data proyek secara menyeluruh dan memberikan saran yang berdasar pada pola dan tren yang teridentifikasi. Dengan demikian, penelitian ini bertujuan untuk mengatasi masalah ketidakakuratan dalam pengelolaan risiko proyek di PT XYZ melalui penerapan Sistem Pakar, yang diharapkan dapat meningkatkan keandalan dan efisiensi dalam pelaksanaan proyek-proyek di masa depan [4]. Beberapa penelitian sebelumnya menyoroti penggunaan metode *forward chaining* oleh sistem pakar saat mengembangkan aplikasi. Misalnya, penelitian oleh Prasetyo & Wahyudi (2019) [5] tentang sistem pakar diagnosa penyakit ternak sapi dan penelitian oleh Puspaningrum (2020) [6] mengenai penerapan metode *forward chaining* dalam mendiagnosa penyakit tanaman sawi. Kedua penelitian tersebut menunjukkan bahwa pendekatan ini dapat

membantu para praktisi di bidang pertanian dan peternakan untuk mengidentifikasi gejala serta mengambil tindakan pencegahan yang tepat, sehingga dapat meminimalkan risiko dan dampak negatif terhadap keberlangsungan usaha mereka. Berdasarkan temuan-temuan tersebut, penggunaan sistem pakar dengan metode *forward chaining* muncul sebagai solusi yang potensial untuk membantu dalam manajemen risiko dalam pengembangan aplikasi di PT XYZ. Tujuan dari penelitian ini adalah untuk merancang dan mengembangkan perangkat lunak yang dapat memberikan rekomendasi yang tepat dalam mengelola risiko yang muncul dalam pengembangan aplikasi di perusahaan tersebut. Dengan demikian, diharapkan bahwa hasil penelitian ini akan membantu PT XYZ dalam mengelola risiko dengan lebih efektif, serta mengurangi dampak negatifnya terhadap keberlangsungan bisnis perusahaan

## II. METODE

Algoritma *forward-chaining* adalah satu dari dua metode utama reasoning (pemikiran) ketika menggunakan *inference engine* (mesin pengambil keputusan) dan bisa secara logis dideskripsikan sebagai aplikasi pengulangan dari modus ponens (satu set aturan inferensi dan argumen yang valid). Lawan dari *forward-chaining* adalah *backward-chaining*[10].

*Forward chaining* dimulai dengan data yang ada dan menggunakan aturan inferensi untuk memperoleh data lain hingga tercapai tujuan atau kesimpulan. Mesin inferensi yang menggunakan rangkaian maju mencari aturan inferensi hingga menemukan satu kondisi yang benar (pernyataan hipotesis atau pernyataan IF -THEN). Setelah aturan ditemukan, mesin keputusan dapat membuat kesimpulan atau konsekuensi (klausa THEN) yang menghasilkan informasi tambahan baru dari informasi yang diberikan. Mesin mengulangi proses ini hingga target ditemukan.

*Forward chaining* merupakan salah satu contoh konsep umum pemikiran berbasis data, yaitu pemikiran yang fokus perhatiannya diawali dengan informasi yang diketahui. Penerusan dapat digunakan pada agen untuk membuat kesimpulan tentang pengamatan di masa depan, seringkali tanpa pertanyaan spesifik.

Berikut ini adalah algoritma dari metode *forward-chaining*:

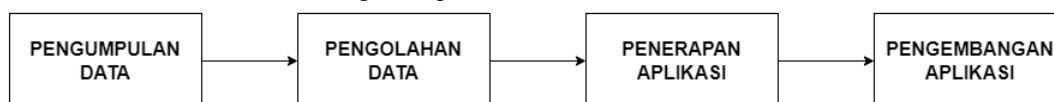
```

function PL-FC-ENTAILS? (KB, Q) returns True or False
inputs: KB, the knowledge base, a set of
propositional definite clauses q, the query, a proposition symbol
lcount←a table, where count( c ) is the number of symbols in c's premise
inferred←a table, where inferred[s] is initially false for all symbols
agenda←a queue of symbols, initially symbols known to be true in KB

while agenda is not empty do p←Pop(agenda)
if p = q then return true
if inferred[p]= false
    then inferred[p]←true
for each clause c in KB where p is in c.PREMISE
    do decrement count[c]
    if count[c] = 0 then add c.CONCLUSION to agenda
return false

```

*Forward chaining* dipilih karena dapat mengatur skor risiko sistem[11]. Gambar 2 menunjukkan bagaimana penelitian ini dilakukan dalam beberapa tahap.



Gambar 2. Tahap metode *Forward Chaining*

Pada pengembangan Sistem Pakar Manajemen Risiko Berbasis Web ini akan dibangun dengan proses perancangan dimulai dari Pengumpulan Data □ Pengolahan Data □ Penerapan Aplikasi □ Pengembangan Aplikasi. Berikut adalah penjelasan dari tahap metode *forward chaining*:

### 1. Pengumpulan Data

Pengumpulan data, adalah data yang relevan dikumpulkan dari berbagai sumber. Data yang berupa fakta-fakta atau informasi awal yang diperlukan untuk memulai proses penalaran.

## 2. Pengolahan Data

Tahap pengolahan data pada metode *forward chaining* merujuk pada proses di mana sistem memproses fakta-fakta yang diketahui, menerapkan aturan-aturan yang relevan, dan memperbarui kumpulan fakta berdasarkan aturan-aturan tersebut.

## 3. Penerapan Aplikasi

Tahap penerapan aplikasi pada metode *forward chaining* merujuk pada fase di mana hasil dari proses *forward chaining* diimplementasikan dalam bentuk aplikasi atau sistem yang berfungsi untuk memecahkan masalah tertentu atau memberikan rekomendasi berdasarkan aturan dan fakta yang ada. Ini merupakan tahap di mana logika inferensi yang telah dikembangkan diintegrasikan ke dalam suatu aplikasi yang bisa digunakan oleh pengguna akhir.

## 4. Pengembangan Aplikasi

Tahap pengembangan aplikasi pada metode *forward chaining* merujuk pada proses di mana konsep dan logika *forward chaining* diimplementasikan dalam sebuah aplikasi perangkat lunak yang dapat digunakan untuk membuat keputusan atau menarik kesimpulan berdasarkan aturan dan fakta yang diketahui.

### A. Data dan Parameter

Data adalah sekumpulan fakta atau informasi yang dapat digunakan untuk analisis, pengambilan keputusan, atau proses komputasi. Data dapat berasal dari berbagai sumber seperti observasi, eksperimen, survei, atau pengumpulan otomatis melalui sensor dan alat teknologi lainnya. Parameter adalah variabel atau nilai yang ditentukan untuk mengatur, mengendalikan, atau mempengaruhi fungsi dari suatu sistem atau model. Parameter sering digunakan dalam model matematika, statistik, dan komputasi untuk mengarahkan bagaimana suatu proses atau sistem harus beroperasi.

#### a. Pengumpulan Data

Pengumpulan data dilakukan melalui wawancara kepada salah satu karyawan divisi IT selaku PIC pengembangan aplikasi di PT XYZ. Wawancara ini bertujuan untuk mengetahui kendala yang dialami dalam rencana pengembangan aplikasi. Selain itu, peneliti juga melakukan teknik observasi, untuk mengetahui penanganan risiko yang cocok sesuai dengan proses bisnis yang berjalan[12].

#### b. Pengolahan Data

Dari hasil analisis dan klasifikasi risiko terdapat dua data yaitu akar penyebab dan strategi penanganan risiko[13] Klasifikasi risiko mengidentifikasi potensi masalah yang mungkin terjadi selama proses pengembangan, implementasi, dan penggunaan sistem informasi. Setiap risiko dijelaskan dengan singkat namun mencakup kemungkinan dampak negatif terhadap proyek. Klasifikasi risiko bisa dilihat pada tabel 1.

Akar penyebab menguraikan faktor-faktor mendasar yang menyebabkan munculnya risiko. Ini bisa berupa kurangnya pemahaman, keterampilan, sumber daya, komunikasi, atau masalah teknis. Mengidentifikasi akar penyebab penting untuk menentukan langkah-langkah yang tepat untuk mitigasi risiko. Akar penyebab bisa dilihat pada tabel 2.

Strategi penanganan adalah langkah-langkah konkret yang diambil untuk mengurangi atau menghilangkan risiko. Strategi ini dirancang berdasarkan akar penyebab yang telah diidentifikasi dan bertujuan untuk meminimalkan dampak risiko terhadap proyek. Strategi penanganan bisa dilihat pada tabel 3.

Tabel 1 Klasifikasi Risiko

Risk ID	Klasifikasi Risiko
A1	Kurangnya pemahaman user tentang ruang lingkup yang ada
A2	Kurangnya dukungan stakeholder di dalam melaksanakan proyek
A3	Data tidak terimplementasi dengan sistem ERP
B1	Tahap implementasi tidak sesuai batas waktu yang ditentukan
B2	Perangkat lunak mengalami error pada saat dilakukan testing
B3	Perangkat lunak error pada saat pelatihan berlangsung
B4	UI sulit dipahami oleh user
B5	Kurangnya testing program ketika proses development
B6	Deliverable yang tidak sesuai
B7	Adanya birokrasi yang menyusahkan
B8	Perangkat pendukung jaringan internet tidak tersedia pada saat pelatihan
C1	Hasil implementasi tidak sesuai dengan harapan stakeholder
C2	Adanya perubahan pada kebijakan dan perencanaan proyek sistem informasi dari stakeholder
C3	Adanya data yang kurang sesuai pada saat proses integrasi sistem

Tabel 2 Akar Penyebab

<b>ID Akar Penyebab</b>	<b>Akar Penyebab</b>
AP1	Kurangnya pemahaman user tentang ruang lingkup aplikasi yang dibutuhkan, sehingga menghasilkan definisi yang kurang jelas.
AP2	Ketidakmampuan user untuk mengartikulasikan kebutuhan dan tujuan aplikasi dengan jelas, sehingga definisi ruang lingkup menjadi tidak jelas.
AP3	Kurangnya waktu atau sumber daya yang diperlukan untuk mendefinisikan ruang lingkup aplikasi secara tepat dan akurat.
AP4	Kurangnya keterampilan pengembang dalam memahami kebutuhan user dalam mendefinisikan ruang lingkup aplikasi.
AP5	Kurangnya dokumentasi kebutuhan
AP6	Stakeholder kurang mengetahui urgensi dan tujuan pembuatan aplikasi
AP7	Rancangan aplikasi yang kurang matang
AP8	Masalah dengan infrastruktur dan teknologi
AP9	Kurangnya pemahaman tentang sistem ERP
AP10	Kompleksitas sistem ERP yang tinggi
AP11	Keterlambatan delivery API atau antarmuka untuk integrasi
AP12	Kurangnya perencanaan dan pengelolaan proyek yang efektif
AP13	Ketidakjelasan persyaratan dan perubahan persyaratan yang sering terjadi
AP14	Kompleksitas proyek
AP15	Kurangnya testing dan QA
AP16	Kurangnya persyaratan performance testing
AP17	Pemrograman yang buruk
AP18	Kurangnya dokumentasi pada proses pengembangan perangkat lunak yang menyebabkan kesulitan dalam proses analisa dampak perubahan
AP19	Fitur aplikasi yang kompleks
AP20	Keterbatasan infrastruktur atau sistem yang tidak memadai
AP21	Tidak memperhitungkan kebutuhan data pengguna
AP22	Terlalu banyak informasi atau elemen yang tidak perlu
AP23	Tidak konsisten dalam penggunaan komponen UI
AP24	Tidak mempertimbangkan penggunaan aplikasi pada berbagai perangkat
AP25	Terlalu banyak tindakan atau proses yang diperlukan
AP26	Kurangnya persyaratan pengujian system
AP27	Kurangnya waktu yang dihabiskan untuk pengujian
AP28	Tidak jelasnya spesifikasi dan persyaratan
AP29	Kurangnya komunikasi dan kolaborasi antara tim
AP30	Kurangnya transparansi dalam proses pengembangan
AP31	Kebijakan dan regulasi yang kompleks
AP32	Ketidakmampuan pengembang untuk mengidentifikasi kebutuhan pengguna
AP33	Tidak adanya cadangan untuk situasi darurat
AP34	Masalah jaringan internet yang tidak stabil
AP35	Kesalahan konfigurasi perangkat pendukung dan jaringan
AP36	Ketidakjelasan kebutuhan dan harapan stakeholder
AP37	Tidak adanya atau kurangnya dokumentasi kebutuhan
AP38	Perubahan kebijakan dan perencanaan yang tidak terkoordinasi dengan baik
AP39	Kurangnya fleksibilitas dalam perencanaan proyek
AP40	Kesalahan dalam proses pengisian data

AP41	Perbedaan format data pada sumber
AP42	Tidak adanya dokumentasi atau deskripsi yang jelas mengenai data
AP43	Perubahan pada struktur data pada sumber data
AP44	Perbedaan waktu sinkronisasi data antar sistem

Tabel 3 Strategi Penanganan

ID Penanganan	Strategi Penanganan
SP1	Melakukan analisis kebutuhan yang lebih mendalam
SP2	Menggunakan teknologi atau alat bantu seperti diagram alir atau mockup aplikasi untuk memperjelas definisi ruang lingkup.
SP3	Mengalokasikan waktu dan sumber daya yang cukup untuk mendefinisikan ruang lingkup aplikasi dengan benar.
SP4	Melibatkan pihak lain seperti pengguna, atau ahli di bidang terkait untuk memberikan umpan balik dan saran dalam memperjelas definisi ruang lingkup.
SP5	Meminta dokumentasi SOP/Kebijakan untuk diterjemahkan ke dalam project charter
SP6	Mengangkat pain point dari user untuk mengetahui urgensi pengembangan aplikasi serta menyajikan laporan proyek yang jelas sesuai target stakeholder
SP7	Mengklarifikasi tujuan proyek dengan stakeholder dan mengembangkan roadmap yang jelas
SP8	Mengkomunikasikan rencana kontinjensi solusi teknologi alternatif
SP9	Menerapkan standar format data pada blueprint agar memudahkan proses integrasi antara Eproc dengan ERP
SP10	Penugasan PIC ERP untuk incharge dalam pengembangan E-Procurement
SP11	Kebutuhan API masuk dalam KAK
SP12	Menggunakan metode manajemen proyek yang efektif seperti scrum atau agile
SP13	Membuat mekanisme pengendalian perubahan persyaratan agar perubahan dapat diatur dan dikomentasikan dengan baik
SP14	Memecah proyek menjadi modul-modul kebutuhan yang lebih kecil dan lebih mudah dikelola.
SP15	Mempersyaratkan QA Engineer dan penggunaan automated testing seperti Selenium (web application) atau Appium (mobile applications) serta dokumentasi yang terukur sesuai kriteria yang telah dibuat
SP16	Melakukan performance testing yang berkaitan dengan kegiatan stress testing, endurance testing, volume testing dan melakukan pdokumentasi hasil pengujian seperti log pengujian, grafik, dan laporan pengujian
SP17	Menggunakan code review tools seperti GitHub, GitLab, BitBucket, and Gerrit secara berkala dan menetapkan standar pengkodean
SP18	Membuat dokumentasi lengkap yang mencakup spesifikasi dan desain aplikasi, dan meng-update secara berkala
SP19	Menyediakan fitur aplikasi yang mudah digunakan dan dipahami oleh pengguna dan dokumentasi panduan yang lengkap
SP20	Memastikan infrastruktur dan sistem yang digunakan dapat menangani beban pengguna yang diharapkan
SP21	Menyiapkan data-data dummy yang bisa digunakan pengguna sebelum proses pelatihan
SP22	Mempertimbangkan aspek tampilan dan organisasi elemen UI secara terstruktur dan mudah dipahami oleh pengguna

---

SP23	Menggunakan UI komponen library
SP24	Mengoptimalkan elemen UI untuk berbagai perangkat yang digunakan pengguna, seperti desktop, laptop, tablet, atau smartphone, serta mempertimbangkan ukuran layar dan tampilan yang sesuai dengan perangkat yang digunakan
SP25	Mengurangi jumlah tindakan atau proses yang diperlukan untuk menggunakan aplikasi, serta mempertimbangkan cara terbaik untuk mengorganisasi elemen UI sehingga lebih mudah dipahami oleh pengguna
SP26	Menetapkan persyaratan pengujian system kepada penyedia, mulai dari pengujian fungsional, keamanan, dan ketahanan aplikasi (performance testing)
SP27	Meningkatkan waktu yang dihabiskan untuk pengujian dengan merencanakan waktu yang cukup untuk pengujian dan memprioritaskan pengujian yang diperlukan sebelum peluncuran aplikasi
SP28	Memastikan bahwa spesifikasi dan persyaratan aplikasi sudah jelas dan disepakati oleh semua pihak terkait sebelum memulai pengembangan, serta melakukan review dan verifikasi secara berkala selama proses pengembangan.
SP29	Meningkatkan komunikasi dan kolaborasi antara tim dengan melakukan rapat reguler, sharing progress, dan pembaruan yang memastikan semua orang terlibat dalam proses pengembangan
SP30	Memberikan akses kepada pihak terkait untuk mengikuti perkembangan pengembangan aplikasi
SP31	Melakukan kajian untuk memahami kebijakan dan regulasi serta memberikan panduan jelas kepada pengguna
SP32	Melakukan survei dan wawancara dengan pengguna untuk memahami kebutuhan mereka
SP33	Memastikan bahwa ada cadangan untuk perangkat pendukung dan jaringan internet dalam situasi darurat
SP34	Menggunakan jaringan internet yang andal dan memastikan bahwa jaringan cadangan tersedia jika terjadi masalah
SP35	Memastikan konfigurasi perangkat dan jaringan dilakukan dengan benar dan sesuai dengan kebutuhan pelatihan
SP36	Melakukan wawancara dengan stakeholder untuk memahami kebutuhan dan harapan mereka dengan jelas
SP37	Membuat dokumentasi kebutuhan yang jelas dan memastikan bahwa stakeholder setuju dengan dokumentasi tersebut
SP38	Membuat dokumentasi perubahan dan evaluasi dampak perubahan
SP39	Memastikan perencanaan proyek mencakup fleksibilitas untuk mengakomodasi perubahan kebijakan dan perencanaan
SP40	Lakukan validasi sumber data dan memberikan laporan hasil migrasi data
SP41	Penyedia wajib memberikan format yang sama pada sumber datanya. Jika diperlukan, lakukan konversi data sebelum diintegrasikan ke dalam sistem.
SP42	Pastikan dokumen yang menjelaskan format dan jenis data yang diintegrasikan dengan sistem telah tersedia dan mudah diakses oleh tim pengembang.
SP43	Pastikan perubahan pada struktur data sumber diinformasikan pada tim pengembang secepat mungkin. Selanjutnya, lakukan evaluasi dan perubahan pada sistem utama sesuai dengan perubahan struktur data tersebut.
SP44	Pastikan waktu sinkronisasi data antar sistem telah disinkronkan dengan baik dan sesuai dengan kebutuhan sistem.

---

### c. Desain Sistem

Desain sistem merujuk pada proses perencanaan dan pengembangan struktur sistem secara keseluruhan. Ini melibatkan pemahaman yang mendalam tentang kebutuhan sistem, perancangan arsitektur yang sesuai, dan implementasi solusi yang efisien dan efektif untuk memenuhi tujuan yang diinginkan.

### d. Model Logik Sistem/roole base

Pada tabel 4 mengaitkan setiap risiko dengan akar penyebabnya dan strategi penanganan yang direkomendasikan. Strategi penanganan dirancang untuk mengatasi akar penyebab yang terkait dengan setiap risiko yang diidentifikasi.

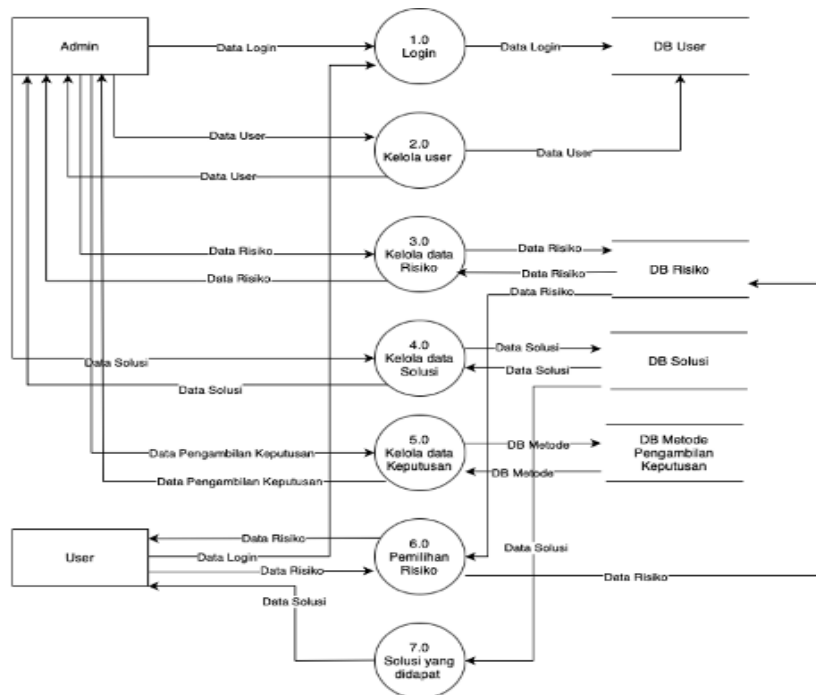
Tabel 4 tabel keputusan

Klasifikasi Risiko	Akar Penyebab	Strategi Penanganan
Kurangnya pemahaman user tentang ruang lingkup yang ada	AP1, AP2, AP3, AP4, AP5	SP1, SP2, SP3, SP4, SP5
Kurangnya dukungan stakeholder di dalam melaksanakan proyek	AP6, AP36, AP37	SP6, SP7, SP36, SP37
Data tidak terimplementasi dengan sistem ERP	AP9, AP10, AP11	SP9, SP10, SP11
Tahap implementasi tidak sesuai batas waktu yang ditentukan	AP12, AP13, AP14	SP12, SP13, SP14
Perangkat lunak mengalami error pada saat dilakukan testing	AP7, AP8, AP15, AP16, AP17, AP18	SP7, SP15, SP16, SP17, SP18
Perangkat lunak error pada saat pelatihan berlangsung	AP8, AP17, AP18, AP20, AP26, AP27	SP20, SP26, SP27, SP33, SP33, SP35
UI sulit dipahami oleh user	AP19, AP21, AP22, AP23, AP24, AP25	SP19, SP22, SP23, SP24, SP25
Kurangnya testing program ketika proses development	AP15, AP26, AP27	SP15, SP26, SP27
Deliverable yang tidak sesuai	AP7, AP12, AP28	SP12, SP13, SP14, SP28
Adanya birokrasi yang menyusahkan	AP31, AP38, AP39	SP31, SP38, SP39
Perangkat pendukung jaringan internet tidak tersedia pada saat pelatihan	AP20, AP33, AP34, AP35	SP33, SP34, SP35
Hasil implementasi tidak sesuai dengan harapan stakeholder	AP36, AP12	SP36, SP37
Adanya perubahan pada kebijakan dan perencanaan proyek sistem informasi dari stakeholder	AP38, AP39, AP31	SP38, SP39, SP31
Adanya data yang kurang sesuai pada saat proses integrasi sistem	AP40, AP41, AP42, AP43, AP44	SP40, SP41, SP42, SP43, SP44

### e. Model Fungsionalitas

Model ini berfokus pada fitur dan operasi yang disediakan oleh sistem serta interaksi antara berbagai komponen sistem untuk menghasilkan output yang diharapkan. Sistem pakar ini dirancang dengan menggunakan DFD level 0 dan DFD level 1 yang dapat dilihat pada Gambar 3 dan 4.





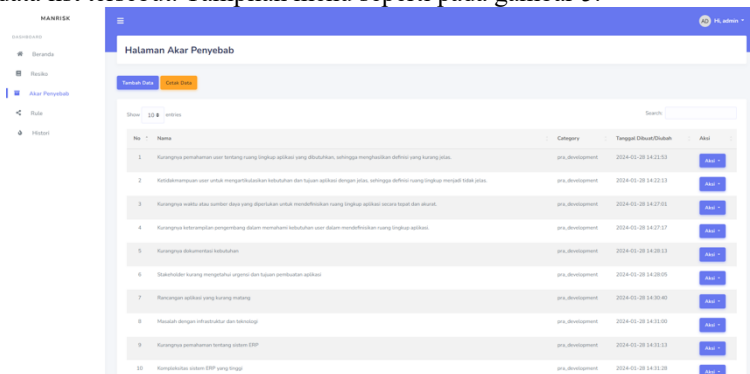
Gambar 4. DFD Level

### III. HASIL DAN PEMBAHASAN

Hasil dari perancangan aplikasi ini dapat dilihat pada gambar berikut dibawah ini yang merupakan tampilan aplikasi MANRISK.

#### 1. Halaman Akar Penyebab

Pada halaman akar penyebab terdapat list data akar penyebab yang akan ditanyakan untuk user. Admin dapat input, edit, dan hapus data list tersebut. Tampilan menu seperti pada gambar 5.

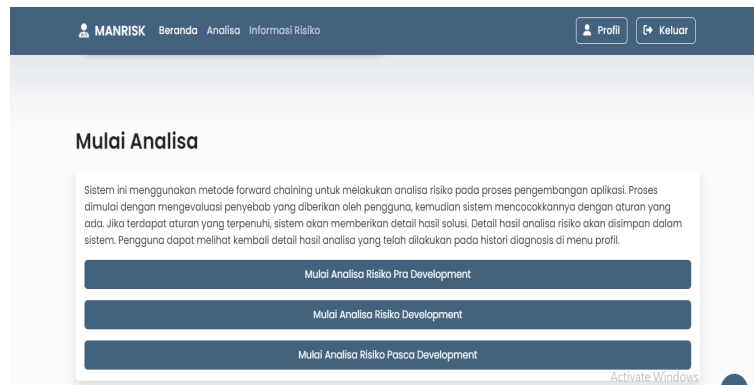


Gambar 5. Tampilan Menu Analisa Risiko

#### 2. Menu Analisa Risiko

Pada halaman Analisa risiko terdapat informasi mengenai list menu analisa risiko meliputi proses pra development, tahap development, pasca development, pada menu ini user harus menjawab pertanyaan dari sistem agar dapat dilakukan pemberian solusi serta hasil analisa dari jawaban yang diberikan. Tampilan menu seperti pada Gambar 6.

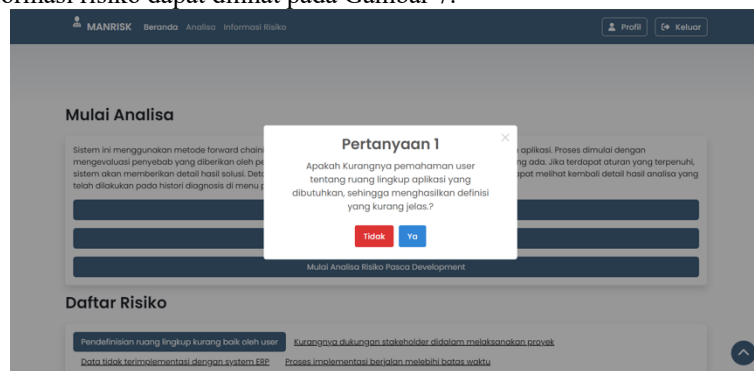




Gambar 6. Tampilan Menu Analisa Risiko

### 3. Mulai Analisa Risiko Pra Development

Saat memulai analisis risiko pra development terdapat beberapa pertanyaan yang harus dijawab oleh user. Tampilan halaman informasi risiko dapat dilihat pada Gambar 7.



Gambar 7. Tampilan Mulai Analisa Risiko Pra Development

### 4. Hasil Detail Solusi Analisis Risiko Pra Development

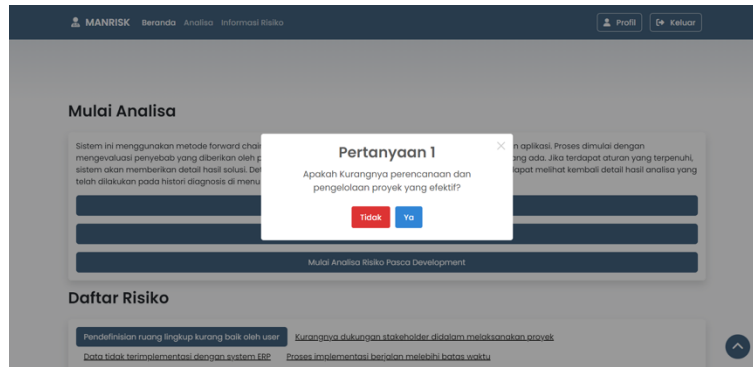
Terdapat hasil detail solusi untuk analisis risiko pra development untuk memberikan solusi kepada user yang telah menjawab pertanyaan dari system. Tampilan halaman informasi risiko dapat dilihat pada Gambar 8.



Gambar 8. Tampilan Hasil Detail Solusi Analisis Risiko Pra Development

### 5. Mulai Analisa Risiko Development

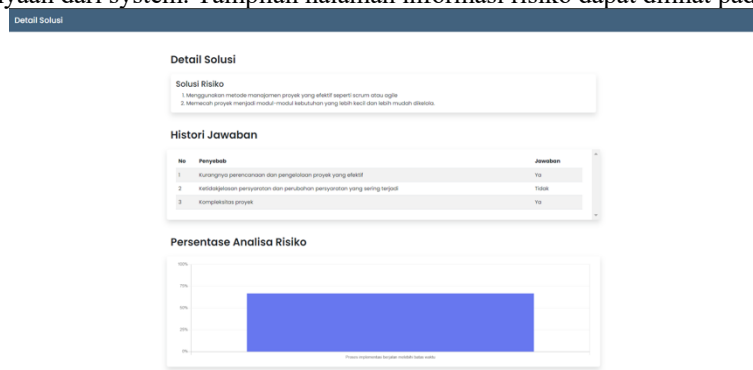
Saat memulai analisis risiko development terdapat beberapa pertanyaan yang harus dijawab oleh user. Tampilan halaman informasi risiko dapat dilihat pada Gambar 9.



Gambar 9. Tampilan Mulai Analisa Risiko Development

## 6. Hasil Detail Solusi Analisa Risiko Development

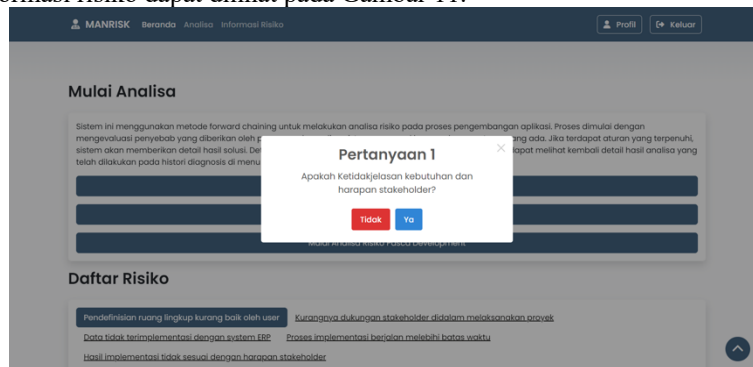
Terdapat hasil detail solusi untuk analisis risiko development untuk memberikan solusi kepada user yang telah menjawab pertanyaan dari system. Tampilan halaman informasi risiko dapat dilihat pada Gambar 10.



Gambar 10. Tampilan Hasil Detail Solusi Analisa Risiko Development

## 7. Mulai Analisa Risiko Pasca Development

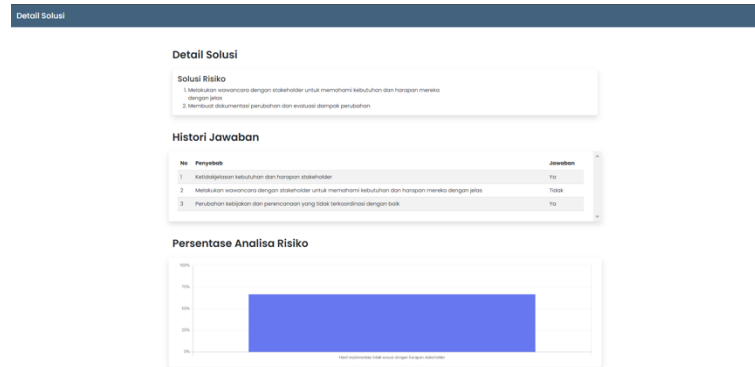
Saat memulai analisis risiko pasca development terdapat beberapa pertanyaan yang harus dijawab oleh user. Tampilan halaman informasi risiko dapat dilihat pada Gambar 11.



Gambar 11. Tampilan Mulai Analisa Risiko Pasca Development

## 8. Hasil Detail Solusi Analisa Risiko Pasca Development

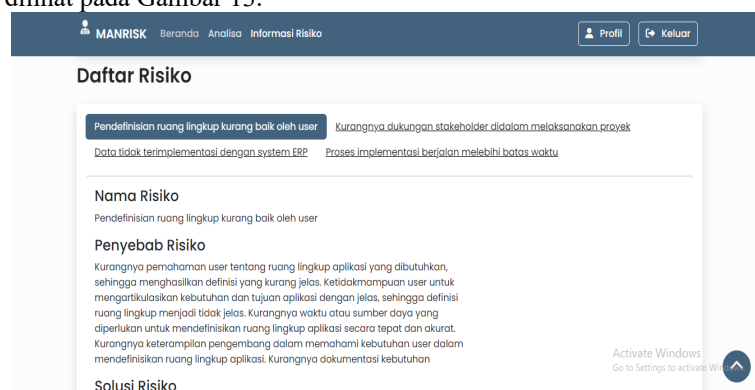
Terdapat hasil detail solusi untuk analisis risiko pasca development untuk memberikan solusi kepada user yang telah menjawab pertanyaan dari system. Tampilan halaman informasi risiko dapat dilihat pada Gambar 12.



Gambar 12. Tampilan Hasil Detail Solusi Analisa Risiko Pasca Development

## 9. Menu Informasi Risiko

Menu informasi risiko memuat informasi daftar risiko yang ada beserta solusinya. Tampilan halaman informasi risiko dapat dilihat pada Gambar 13.



Gambar 13. Tampilan Informasi Risiko

## 10. Pengujian

Pengujian dilakukan untuk mengetahui tingkat kesesuaian hasil antara sistem pakar dan pakar[14]. Hasil pengujian dapat dilihat pada Tabel 5.

Tabel 5 Hasil Pengujian

No Kasus	Jenis Analisa	Parameter Inputan	Hasil Penalaran dari Aplikasi	Hasil Penalaran oleh Pakar	Ket Akurasi
1.	Analisa Risiko Pra Development	1. ya 2. tidak 3. tidak 4. ya 5. ya 6. tidak 7. ya 8. ya 9. tidak 10. ya 11. tidak	- Melakukan analisis kebutuhan yang lebih mendalam - Melibatkan pihak lain seperti pengguna, atau ahli di bidang terkait untuk memberikan umpan balik dan saran dalam memperjelas definisi ruang lingkup. - Meminta dokumentasi SOP/Kebijakan untuk diterjemahkan ke dalam project charter - Mengklarifikasi tujuan proyek dengan stakeholder dan mengembangkan roadmap yang jelas	- Melakukan analisis kebutuhan yang lebih mendalam - Melibatkan pihak lain seperti pengguna, atau ahli di bidang terkait untuk memberikan umpan balik dan saran dalam memperjelas definisi ruang lingkup. - Meminta dokumentasi SOP/Kebijakan untuk diterjemahkan ke dalam project charter - Mengklarifikasi tujuan proyek dengan stakeholder dan mengembangkan roadmap yang jelas	Akurat

			- Mengkomunikasikan rencana kontinjensi solusi teknologi alternatif	- Mengkomunikasikan rencana kontinjensi solusi teknologi alternative	
			- Penugasan PIC ERP untuk incharge dalam pengembangan E-Procurement	- Penugasan PIC ERP untuk incharge dalam pengembangan E-Procurement	
2.	Analisa Risiko Pra Development	1. tidak 2. tidak 3. tidak 4. tidak 5. tidak 6. tidak 7. tidak 8. ya 9. ya 10. ya 11. ya	- Mengkomunikasikan rencana kontinjensi solusi teknologi alternatif - Menerapkan standar format data pada blueprint agar memudahkan proses integrasi antara Eproc dengan ERP - Penugasan PIC ERP untuk incharge dalam pengembangan E-Procurement - Kebutuhan API masuk dalam KAK	- Mengkomunikasikan rencana kontinjensi solusi teknologi alternatif - Menerapkan standar format data pada blueprint agar memudahkan proses integrasi antara Eproc dengan ERP - Penugasan PIC ERP untuk incharge dalam pengembangan E-Procurement - Kebutuhan API masuk dalam KAK	Akurat
3.	Analisa Risiko Pra Development	1. ya 2. tidak 3. ya 4. tidak 5. ya 6. tidak 7. ya 8. tidak 9. ya 10. tidak 11. ya	- Melakukan analisis kebutuhan yang lebih mendalam - Mengalokasikan waktu dan sumber daya yang cukup untuk mendefinisikan ruang lingkup aplikasi dengan benar. - Meminta dokumentasi SOP/Kebijakan untuk diterjemahkan ke dalam project charter - Mengklarifikasi tujuan proyek dengan stakeholder dan mengembangkan roadmap yang jelas - Menerapkan standar format data pada blueprint agar memudahkan proses integrasi antara Eproc dengan ERP - Kebutuhan API masuk dalam KAK	- Melakukan analisis kebutuhan yang lebih mendalam - Mengalokasikan waktu dan sumber daya yang cukup untuk mendefinisikan ruang lingkup aplikasi dengan benar. - Meminta dokumentasi SOP/Kebijakan untuk diterjemahkan ke dalam project charter - Mengklarifikasi tujuan proyek dengan stakeholder dan mengembangkan roadmap yang jelas - Menerapkan standar format data pada blueprint agar memudahkan proses integrasi antara Eproc dengan ERP - Kebutuhan API masuk dalam KAK	akurat
4.	Analisa Risiko Development	1. ya 2. ya 3. tidak	- Menggunakan metode manajemen proyek yang efektif seperti scrum atau agile - Membuat mekanisme pengendalian perubahan persyaratan agar perubahan dapat diatur dan dikomentasikan dengan baik	- Menggunakan metode manajemen proyek yang efektif seperti scrum atau agile - Membuat mekanisme pengendalian perubahan persyaratan agar perubahan dapat diatur dan dikomentasikan dengan baik	Akurat
5.	Analisa Risiko Development	1. tidak 2. tidak 3. ya	- Memecah proyek menjadi modul-modul kebutuhan yang lebih kecil dan lebih mudah dikelola.	- Memecah proyek menjadi modul-modul kebutuhan yang lebih kecil dan lebih mudah dikelola.	Akurat

6.	Analisa Risiko Development	1. ya 2. tidak 3. ya	- Menggunakan metode manajemen proyek yang efektif seperti scrum atau agile - Memecah proyek menjadi modul-modul kebutuhan yang lebih kecil dan lebih mudah dikelola.	- Menggunakan metode manajemen proyek yang efektif seperti scrum atau agile - Memecah proyek menjadi modul-modul kebutuhan yang lebih kecil dan lebih mudah dikelola.	Akurat
7.	Analisa Risiko Pasca Development	1. tidak 2. ya 3. ya	- Melakukan wawancara dengan stakeholder untuk memahami kebutuhan dan harapan mereka dengan jelas - Membuat dokumentasi perubahan dan evaluasi dampak perubahan	- Melakukan wawancara dengan stakeholder untuk memahami kebutuhan dan harapan mereka dengan jelas - Membuat dokumentasi perubahan dan evaluasi dampak perubahan	Akurat
8.	Analisa Risiko Pasca Development	1. ya 2. tidak 3. ya	- Melakukan wawancara dengan stakeholder untuk memahami kebutuhan dan harapan mereka dengan jelas - Membuat dokumentasi perubahan dan evaluasi dampak perubahan	- Melakukan wawancara dengan stakeholder untuk memahami kebutuhan dan harapan mereka dengan jelas - Membuat dokumentasi perubahan dan evaluasi dampak perubahan	Akurat
9.	Analisa Risiko Pasca Development	1. ya 2. tidak 3. tidak	- Melakukan wawancara dengan stakeholder untuk memahami kebutuhan dan harapan mereka dengan jelas	- Melakukan wawancara dengan stakeholder untuk memahami kebutuhan dan harapan mereka dengan jelas	Akurat

Berdasarkan hasil uji pada Tabel 5, maka didapatkan hasil perhitungan akurasi sebagai berikut:

$$\text{Nilai akurasi} = \frac{\text{jumlah data akurat}}{\text{jumlah data}} \%$$

$$\text{Nilai akurasi} = \frac{14}{14} \times 100 \% = 100\%$$

Dari hasil perhitungan diatas, dapat disimpulkan bahwa sistem pakar ini berfungsi dengan baik dan layak untuk digunakan karena tingkat akurasi sistem mencapai 100%[15].

Penelitian ini diawali dengan identifikasi masalah akurasi yang rendah pada metode konvensional untuk mendeteksi penyakit. Hasil pengujian menunjukkan bahwa metode ini mampu meningkatkan akurasi secara signifikan lebih tinggi dibandingkan dengan metode lama. Dengan demikian, metode ini berhasil menyelesaikan masalah utama yang diidentifikasi pada awal penelitian.

#### 1. Penguatan Temuan Terdahulu:

- Anindhita Ari Putri dan Deva Istifadhah Irnanda (2020)[7] : Penelitian mereka berfokus pada risiko teknologi informasi di aplikasi J&T Express. Temuan bahwa mitigasi risiko bisa meningkatkan kinerja aplikasi mendukung temuan kita bahwa metode baru meningkatkan akurasi deteksi penyakit dengan mitigasi kesalahan prediksi. Pertentangan dengan Temuan Terdahulu:
- Ryan Haryo Pangestu dan Ariya Dwika Tanaem Cahyono (2021)[8] : Mereka menunjukkan bahwa risiko gangguan jaringan dan kegagalan implementasi SOP signifikan dalam mengganggu aplikasi SIPP. Temuan ini relevan dengan penekanan kita pada pentingnya infrastruktur yang stabil dalam mencapai akurasi tinggi.

#### 2. Pertentangan Dengan Temuan Terdahulu

Harefa Wirawan (2022)[9] : Mereka menekankan risiko pada kegagalan SOP dan gangguan listrik pada aplikasi ERP. Penelitian kita menunjukkan bahwa metode machine learning dapat mengurangi risiko kesalahan dalam aplikasi medis, menantang asumsi bahwa risiko infrastruktur adalah penghalang utama.

## VII. SIMPULAN

Dari penelitian yang telah dilakukan menghasilkan kesimpulan sebagai berikut sistem pakar manajemen berbasis web dengan metode *forward chaining* telah berhasil dibangun dan dapat memberikan informasi mengenai klasifikasi risiko serta strategi penanganan berdasarkan akar penyebab yang telah dimasukkan oleh pengguna. Berdasarkan hasil pengujian yang dilakukan dengan memasukkan setiap akar penyebab, didapatkan hasil uji akurasi sistem mencapai 100%. Hal ini membuktikan jika sistem pakar ini layak digunakan dan berfungsi dengan baik. Untuk penelitian selanjutnya, diharapkan sistem ini dapat dikembangkan lagi dengan penambahan metode agar nantinya sistem dapat berfungsi dengan lebih maksimal

## UCAPAN TERIMA KASIH

Alhamdulillah, Puji syukur kehadirat Allah SWT telah memberikan rahmat dan hidayah yang saya perlukan untuk menyelesaikan skripsi ini, yang merupakan syarat untuk memperoleh gelar sarjana saya. Pertama – tama saya ingin mengucapkan terima kasih kepada keluarga saya atas dukungan mereka yang tak tergoyahkan selama penyelesaian artikel saya ini. Perguruan tinggi yang telah menyediakan peralatan yang saya butuhkan untuk melanjutkan studi saya, dan teman-teman saya selalu ada untuk mendorong saya supaya terus maju untuk melanjutkan skripsi ini hingga lulus

## REFERENSI

- [1] Z. Zulkarnaen and D. H. Ramdhan, “Analisis Faktor-Faktor Yang Berhubungan Dengan Kecelakaan Kerja Pada Pekerja Bagian Produksi Di PT. XYZ,” *J. Cahaya Mandalika*, vol. 3, no. 2, pp. 728–741, 2023, [Online]. Available: <https://www.ojs.cahayamandalika.com/index.php/JCM/article/view/1745>
- [2] A. . Rahadian, “REVITALISASI BIROKRASI MELALUI TRANSFORMASI BIROKRASI MENUJU E-GOVERNANCE PADA ERA REVOLUSI INDUSTRI 4.0,” *Sustain.*, vol. 11, no. 1, pp. 1–14, 2019.
- [3] M. N. H. Alvianto, N. P. Adam, I. A. Sodik, E. Sedyono, and A. P. Widodo, “Dampak Dan Faktor Kesuksesan Penerapan Enterprise Resource Planning Terhadap Kinerja Organisasi: Systematic Literature Review,” *J. Nas. Teknol. dan Sist. Inf.*, vol. 7, no. 3, pp. 172–180, 2022, doi: 10.25077/teknosi.v7i3.2021.172-180.
- [4] M. Miftakhatun, “Analisis Manajemen Risiko Teknologi Informasi pada Website Ecofo Menggunakan ISO 31000,” *J. Comput. Sci. Eng.*, vol. 1, no. 2, pp. 128–146, 2020, doi: 10.36596/jcse.v1i2.76.
- [5] W. D. Prasetyo and R. Wahyudi, “Sistem Pakar Diagnosis Penyakit Ternak Sapi Menggunakan Metode Forward Chaining Berbasis Website Responsif,” *J. Teknol. dan Terap. Bisnis*, vol. 2, no. 1, pp. 13–21, 2019.
- [6] A. S. Puspaningrum, E. R. Susanto, and A. Sucipto, “Penerapan Metode Forward Chaining untuk Mendiagnosa Penyakit Tanaman Sawi,” *INFORMAL Informatics J.*, vol. 5, no. 3, p. 113, 2020, doi: 10.19184/isj.v5i3.20237.
- [7] A. Journal, A. A. Putri, and D. I. Irnanda, “Volume 4 issue 1 1 Aisyah Journal of Informatics and Electrical Engineering ANALISIS RISIKO TEKNOLOGI INFORMASI MENGGUNAKAN ISO 31000 (STUDI KASUS : APLIKASI J&T EXPRESS INDONESIA),” vol. 4, no. 1, pp. 1–9, [Online]. Available: <http://jti.aisyahuniversity.ac.id/index.php/AJIEE>
- [8] R. H. Pangestu, A. D. Cahyono, and P. F. Tanaem, “Analisis Manajemen Resiko Aplikasi SIPP di Pengadilan Negeri Salatiga Kelas 1B Menggunakan ISO 31000,” *J. Comput. Inf. Syst. Ampera*, vol. 2, no. 1, pp. 43–57, 2021, doi: 10.51519/journalcisa.v2i1.59.
- [9] W. Harefa, “Analisis Manajemen Risiko Dengan Menggunakan Framework ISO 31000:2018 Pada Sistem Informasi Gudang,” *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 9, no. 1, pp. 407–420, 2022, doi: 10.35957/jatisi.v9i1.1478.
- [10] S. H. Lina, M. Marsa, and A. D. D. Opu, “Implementasi Algoritma Certainty Factor dan Forward Chaining untuk Rekomendasi dan Larangan Makanan,” *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 4, no. 1, pp. 340–349, 2024, doi: 10.57152/malcom.v4i1.1215.
- [11] I. R. Yansyah and S. Sumijan, “Sistem Pakar Metode Forward Chaining untuk Mengukur Keparahan Penyakit Gigi dan Mulut,” *J. Sistim Inf. dan Teknol.*, vol. 3, pp. 41–47, 2021, doi: 10.37034/jsisfotek.v3i2.42.
- [12] M. Muafi, A. Wijaya, and V. A. Aziz, “Sistem Pakar Mendiagnosa Penyakit Mata Pada Manusia Menggunakan Metode Forward Chaining,” *COREAI J. Kecerdasan Buatan, Komputasi dan Teknol. Inf.*, vol. 1, no. 1, pp. 43–49, 2020, doi: 10.33650/coreai.v1i1.1669.
- [13] D. Junianti and C. Fibriani, “Analisis Resiko Aplikasi Sistem Informasi Pengelolaan Data Umat Menggunakan ISO 31000 (Studi Kasus: Gereja Katolik Santo Paulus Miki Salatiga),” *J. Comput. Inf. Syst. Ampera*, vol. 2, no. 2, pp. 107–128, 2021, doi: 10.51519/journalcisa.v2i2.68.

- [14] F. Chaining, “Sistem Pakar Diagnosa Penyakit ISPA Berbasis Web Dengan Metode,” vol. 3, no. 28, pp. 81–90, 2022.
- [15] M. Sari, S. Defit, and G. W. Nurcahyo, “Sistem Pakar Deteksi Penyakit pada Anak Menggunakan Metode Forward Chaining,” *J. Sistim Inf. dan Teknol.*, vol. 2, pp. 130–135, 2020, doi: 10.37034/jsisfotek.v2i4.34.

**Conflict of Interest Statement:**

*The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.*