Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id

E - ISSN: 2723-7079 | P - ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari

OPEN ACCESS

https://doi.org/10.35870/jimik.v6i1.1214

Penerapan Manajemen Risiko dalam *Software Quality Assurance*: Studi Kasus Pengujian *Black Box* pada Aplikasi Inventori *Maintenance*

Agung Suderajat [1*], Abraham Leo Contantinus Meze Laki [2], Aldo Anggito Abimanyu [3], Wahyu Hadikristanto [4], Abdul Halim Anshor [5]

[1*,2,3,4,5] Universitas Pelita Bangsa, Kabupaten Bekasi, Provinsi Jawa Barat, Indonesia.

Email: agungst66@mhs.pelitabangsa.ac.id [1*], abraham.leo@mhs.pelitabangsa.ac.id [2], aldoabimanyu548@mhs.pelitabangsa.ac.id [3], wahyu.hadikristanto@pelitabangsa.ac.id [4], abdulhalimanshor@pelitabangsa.ac.id [5]

Histori Artikel:
Dikirim 1 November 2024; *Diterima dalam bentuk revisi* 15 November 2024; *Diterima* 30 Desember 2024; *Diterbitkan* 10 Januari 2025. Semua hak dilindungi oleh Lembaga Penelitian dan Pengabdian Masyarakat (LPPM) STMIK Indonesia Banda Aceh.

Abstrak

Manajemen risiko memainkan peran penting dalam software quality assurance (SQA) untuk mengurangi potensi kegagalan yang memengaruhi kualitas perangkat lunak. Penelitian ini bertujuan untuk mengeksplorasi penerapan manajemen risiko dalam pengujian black box pada aplikasi inventori maintenance, guna mengidentifikasi dan mengelola risiko yang muncul pada sistem. Studi ini menggunakan metode identifikasi risiko, analisis dampak, dan evaluasi frekuensi risiko, yang diterapkan pada skenario pengujian black box. Pada pengujian black box tidak ditemukan kesalahan pada sistem dalam pengujian fitur-fiturnya. Penelitian ini menunjukkan bahwa 18 kemungkinan risiko, seperti kesalahan fitur-fitur pada sistem dan kesalahan fungsi antarmuka, dapat diminimalkan melalui strategi mitigasi, termasuk training user dan preventive & corrective maintenance. Hasil ini menyimpulkan perancangan sistem ini sudah dilakukan dengan baik dan berfungsi secara optimal, namun perlu tetap melakukan tindakan-tindakan pencegahan untuk menjaga sistem dapat berjalan dengan baik.

Kata Kunci: Manajemen Risiko; Software Quality Assurance; Black Box; Aplikasi Inventory.

Abstract

Risk management plays a crucial role in software quality assurance (SQA) to reduce the potential for failures that impact software quality. This study aims to explore the implementation of risk management in black box testing for inventory maintenance applications, focusing on identifying and managing risks that arise within the system. The study employs risk identification, impact analysis, and risk frequency evaluation methods applied to black box testing scenarios. The black box testing results revealed no errors in the system's features during the tests. This research identifies 18 potential risks, such as feature errors and interface function failures, which can be minimized through mitigation strategies, including user training and preventive and corrective maintenance. The findings conclude that the system design has been well-executed and operates optimally. However, preventive measures are still necessary to ensure the system continues to function effectively.

Keyword: Risk Management; Software Quality Assurance; Black Box; Inventory App.

Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id
E - ISSN: 2723-7079 | P - ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214

1. Pendahuluan

Dalam era digitalisasi, sistem informasi dan teknologi informasi memiliki peran strategis sebagai elemen pendukung utama kemajuan perusahaan (Ecleas, 2021). Aplikasi perangkat lunak, seperti sistem pemeliharaan inventaris, menjadi komponen krusial dalam memastikan keakuratan data inventaris serta mendukung proses pemeliharaan yang berkelanjutan. Namun demikian, ketidakakuratan data atau kegagalan aplikasi dapat berujung pada gangguan operasional yang signifikan, termasuk keterlambatan perbaikan dan meningkatnya biaya operasional. Meskipun teknologi informasi mampu meningkatkan efisiensi dan pencapaian tujuan organisasi, risiko kegagalan sistem tetap menjadi ancaman yang dapat menimbulkan kerugian besar (Sylvia Ferrore & Trivena Yulianti, 2016). Penerapan *Software Quality Assurance* (SQA) berbasis pendekatan manajemen risiko diperlukan untuk mengantisipasi dan mengelola potensi risiko yang berpengaruh terhadap kinerja aplikasi. Implementasi *quality assurance* dan pengujian memainkan peran kunci dalam mengidentifikasi risiko secara dini serta mengendalikan dampaknya, dengan perhatian utama pada aspek yang memengaruhi kualitas perangkat lunak secara signifikan (Patel, 2024). Analisis manajemen risiko yang diterapkan dalam teknologi informasi memungkinkan perusahaan untuk mengurangi kemungkinan kegagalan aplikasi, menjaga keandalan data, serta memastikan keberlanjutan sistem inventaris (Husein & Imbar, 2015). Penerapan manajemen risiko dalam SQA meliputi serangkaian aktivitas sistematis, mulai dari pengembangan rencana, peninjauan, analisis, evaluasi, hingga penerapan metodologi untuk mencegah terjadinya masalah kualitas perangkat lunak. Seperti halnya manajemen risiko, SQA harus dilibatkan sejak tahap awal proyek dan diterapkan secara konsisten di setiap fase pengembangan perangkat lunak. Pendekatan ini memungkinkan identifikasi serta mitigasi kesalahan sejak dini, sehingga tidak hanya meningkatkan kualitas perangkat lunak, tetapi juga mengurangi biaya selama siklus hidupnya (Mohsin Nazir, 2020). Teori dan metode pengujian relevan menjadi dasar penting dalam mendukung analisis risiko. Risiko didefinisikan sebagai konsekuensi dari ketidakpastian yang timbul akibat kurangnya informasi atau ketidakpastian terkait peristiwa di masa depan (Ecleas, 2021). Sementara itu, manajemen risiko melibatkan proses analisis dan penanganan kerugian potensial yang dapat memengaruhi organisasi. Proses ini dilakukan oleh pihak manajemen di tingkat eksekutif perusahaan (Atmojo & Manuputty, 2020). Melalui penerapan manajemen risiko, perusahaan diharapkan mampu menyusun prosedur operasional yang lebih efisien, mengurangi potensi risiko, serta menyelesaikan berbagai kendala yang muncul selama proses bisnis berlangsung (Ecleas, 2021).

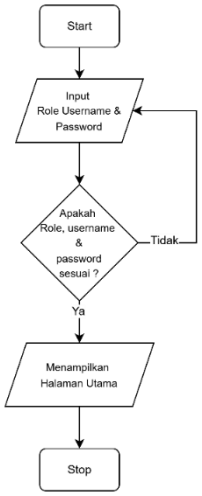
Software Quality Assurance berfokus pada pemeliharaan kualitas perangkat lunak melalui penerapan standar dan metodologi tertentu untuk memastikan perangkat lunak dapat berfungsi secara optimal (Jordan *et al.*, 2024). Dalam konteks pengujian perangkat lunak, metode *Black Box* sering kali digunakan untuk mengevaluasi fungsionalitas perangkat lunak berdasarkan spesifikasi pengguna tanpa memerlukan akses ke kode sumber. Pendekatan ini bertujuan untuk mendeteksi dan memperbaiki kesalahan yang ditemukan dalam sistem (Praniffa *et al.*, 2023). Sistem aplikasi didefinisikan sebagai kumpulan instruksi atau kode yang dirancang untuk memenuhi kebutuhan pengguna, sedangkan inventaris didefinisikan sebagai sumber daya yang tersedia tetapi belum digunakan, seperti stok barang atau aset organisasi (Cahyaningtyas & Sigi, 2015). Penelitian oleh Anisya dkk. yang berjudul "Pengujian *Black Box* dan *White Box* Sistem Informasi Parkir Berbasis Web" menunjukkan bahwa metode pengujian *Black Box* dan *White Box* dapat diterapkan untuk mengevaluasi sistem informasi berbasis web. Penelitian ini menemukan bahwa sistem informasi parkir Universitas Islam Negeri Sultan Syarif Kasim Riau berfungsi dengan baik dan valid sesuai spesifikasi yang diharapkan (Praniffa *et al.*, 2023). Penelitian lainnya, "Analisis Manajemen Risiko Teknologi Informasi Software PEGA Menggunakan ISO 31000" oleh Joshua Ecleas dan Augie David Manuputty, mengidentifikasi 19 risiko yang diklasifikasikan dalam dua kategori berdasarkan matriks kemungkinan (*likelihood*) dan dampak (*impact*). Hasilnya menunjukkan bahwa langkah mitigasi yang diterapkan perusahaan mampu mengurangi dampak risiko yang berpotensi muncul (Ecleas, 2021). Gita Mustika Rahmah berjudul "Analisis Manajemen Risiko Penerapan Sistem Informasi di Politeknik STMI

Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id
E - ISSN: 2723-7079 | P - ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214

Jakarta" mengidentifikasi 24 risiko yang diklasifikasikan berdasarkan matriks kemungkinan (*likelihood*) dan dampak (*impact*). Penelitian tersebut merekomendasikan langkah-langkah mitigasi yang terbagi menjadi dua kategori utama, yaitu *accept* dan *mitigate* (Rahmah, 2019).

2. Metode Penelitian

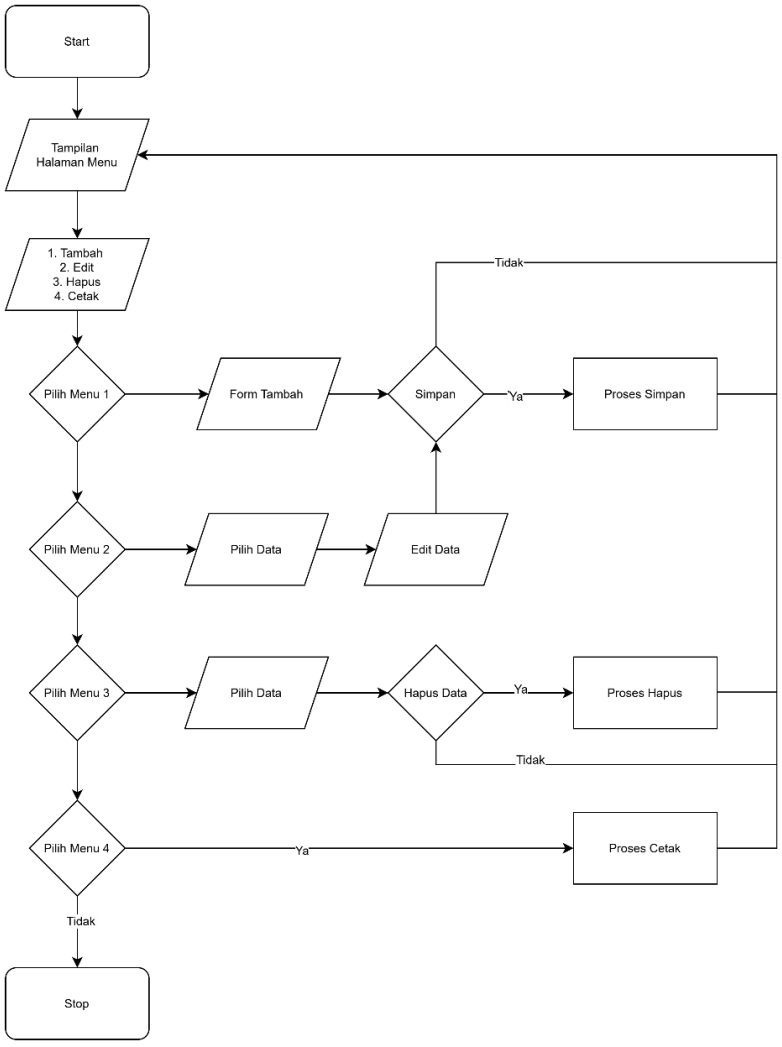
Penelitian ini menerapkan pendekatan *Black Box Testing* untuk mengevaluasi kualitas perangkat lunak aplikasi *inventory maintenance*. Pendekatan ini dipilih karena sesuai dengan tujuan untuk menilai kesesuaian fungsionalitas perangkat lunak berdasarkan spesifikasi pengguna tanpa perlu mengakses kode sumber internal. Pengujian perangkat lunak memegang peranan penting dalam memastikan kualitas produk perangkat lunak. Metode *Black Box Testing* menawarkan berbagai keunggulan, termasuk kecepatan proses pengujian, tidak memerlukan keahlian pemrograman, fokus pada fungsi perangkat lunak sesuai kebutuhan pengguna, serta dukungan dokumentasi yang terstruktur sehingga proses pengujian dapat ditelusuri dan diulang dengan mudah di masa mendatang (Maspupah, 2022). Dalam pengujian sistem informasi, validasi merupakan proses utama untuk memastikan bahwa perangkat lunak telah memenuhi spesifikasi dan tujuan yang diharapkan. Validasi data yang kurang optimal dapat menyebabkan data yang tidak sesuai tersimpan dalam basis data, sehingga memengaruhi keakuratan informasi. Oleh karena itu, validasi yang baik menjadi kebutuhan mendasar dalam pengujian perangkat lunak. Pada penelitian ini, pengujian sistem informasi manajemen inventaris berbasis web dilakukan untuk memastikan bahwa aplikasi berfungsi sesuai dengan kebutuhan pengguna (Ismail & Efendi, 2020). Dalam meningkatkan efektivitas pengujian *Black Box*, penggunaan diagram alir atau *flowchart* sangat disarankan untuk memetakan alur logika sistem informasi yang diuji. *Flowchart* merupakan representasi grafis dari langkah-langkah penyelesaian masalah, menggunakan simbol tertentu untuk menggambarkan proses secara sistematis. Diagram ini memungkinkan pemahaman yang lebih baik mengenai alur logika program, sehingga mendukung identifikasi komponen yang memerlukan evaluasi lebih lanjut. Selain itu, aturan desain dalam diagram alir harus dikomunikasikan dengan jelas agar setiap komponen dapat dipahami dengan lebih baik (Pemrograman, 2020). Pada sistem informasi aplikasi manajemen inventaris berbasis web yang menjadi fokus penelitian ini, diagram alir digunakan untuk menggambarkan urutan langkah atau proses yang diterapkan dalam sistem. *Flowchart* tersebut memetakan logika dan proses sistem secara visual, mulai dari input data, validasi, hingga keluaran yang dihasilkan, sehingga memfasilitasi proses analisis lebih lanjut terhadap fungsionalitas sistem.



Gambar 1. *Flowchart Login*

Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id
E - ISSN: 2723-7079 | P - ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214

Flowchart di atas menjelaskan alur proses login pada sistem informasi manajemen inventaris. Proses dimulai dengan langkah awal yang ditandai dengan *Start*, sebagai penanda inisiasi proses login. Pengguna kemudian diminta untuk memasukkan informasi berupa *role*, nama pengguna (*username*), dan kata sandi (*password*). Setelah data dimasukkan, sistem akan memverifikasi apakah informasi yang diberikan sesuai. Jika *role*, *username*, dan *password* yang dimasukkan benar, proses login dianggap berhasil, dan pengguna akan diarahkan ke halaman utama sistem. Sebaliknya, jika data yang dimasukkan salah, pengguna akan diminta untuk mengulangi langkah sebelumnya dengan memasukkan informasi yang benar. Setelah proses login berhasil, halaman utama sistem ditampilkan kepada pengguna, dan alur ini diakhiri dengan langkah yang ditandai sebagai *Stop*. Proses ini memastikan validasi yang akurat sebelum pengguna dapat mengakses fitur sistem.

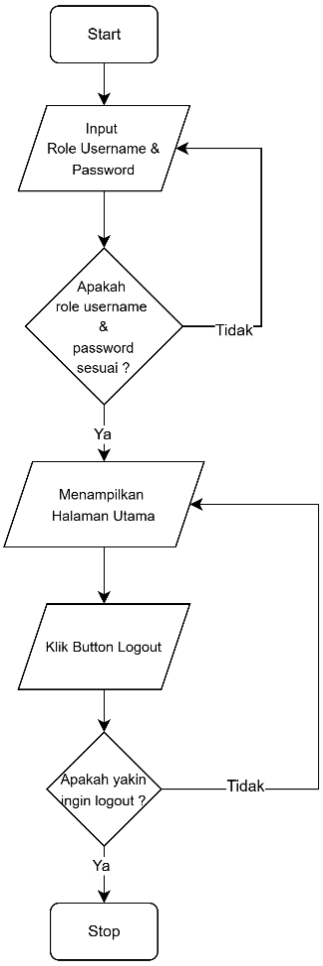


Gambar 2. *Flowchart* Pengolahan Data

Flowchart ini menggambarkan alur program yang dirancang untuk mengelola data dalam sistem informasi. Proses dimulai dengan langkah awal yang ditandai sebagai "Mulai," di mana sistem menampilkan menu utama kepada pengguna. Menu utama ini mencakup beberapa opsi, yaitu menambah data, mengedit data, menghapus data, dan mencetak data. Pengguna dapat memilih salah satu opsi sesuai kebutuhan. Jika pengguna memilih untuk menambahkan data, sistem akan menampilkan formulir yang memungkinkan pengguna untuk mengisi informasi baru. Setelah data

Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id
E - ISSN: 2723-7079 | P - ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214

diisi dan disimpan, program akan kembali ke menu utama. Apabila pengguna memilih opsi untuk mengedit data, sistem akan menampilkan data yang akan diperbarui. Pengguna dapat melakukan perubahan yang diperlukan, menyimpannya, dan sistem akan kembali ke menu utama. Apabila pengguna memilih opsi untuk menghapus data, program akan meminta konfirmasi sebelum menghapus data yang dipilih. Setelah mendapatkan konfirmasi, data akan dihapus, dan sistem kembali ke menu utama. Untuk opsi mencetak data, sistem akan menampilkan data yang tersedia untuk dicetak, dan setelah proses pencetakan selesai, sistem kembali ke menu utama. Proses ini terus berlanjut hingga pengguna memilih untuk keluar dari program, yang akan mengakhiri program dengan langkah yang ditandai sebagai "Selesai." Alur ini memastikan setiap fungsi dijalankan secara sistematis sesuai dengan kebutuhan pengguna.



Gambar 3. *Flowchart Logout*

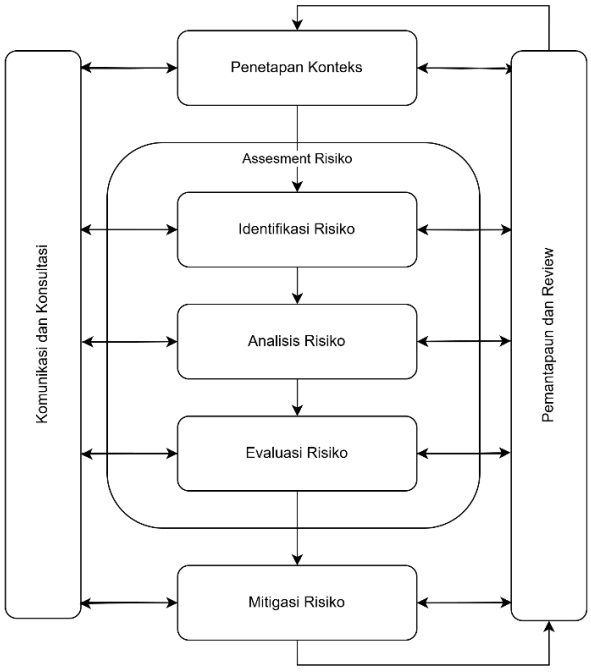
Flowchart ini menguraikan langkah-langkah yang diperlukan untuk melaksanakan proses login dan logout pada sebuah aplikasi dengan validasi terhadap *role*, *username*, dan *password*. Proses dimulai dengan langkah awal yang ditandai sebagai *Start*. Pengguna kemudian diminta untuk memasukkan *role*, *username*, dan *password*. Setelah data dimasukkan, sistem akan memverifikasi apakah informasi yang diberikan sesuai. Jika data valid, pengguna diarahkan ke halaman utama aplikasi. Sebaliknya, jika data tidak valid, sistem akan mengarahkan pengguna untuk kembali ke tahap input guna memperbaiki kesalahan. Setelah berada di halaman utama, pengguna memiliki opsi untuk logout dari sistem dengan mengklik tombol logout. Pada tahap ini, sistem akan meminta konfirmasi dari pengguna mengenai

Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id
E - ISSN: 2723-7079 | P - ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214

keinginan untuk logout. Jika pengguna memastikan ingin *logout*, proses akan berakhir dengan langkah yang ditandai sebagai *Stop*. Namun, jika pengguna membatalkan *logout*, sistem akan mengembalikan pengguna ke halaman utama. Alur ini dirancang untuk memastikan keamanan dan validitas proses login dan logout dalam sistem aplikasi.

2.1 Manajemen Resiko

Pada kasus ini, kerangka kerja ISO 31000 diterapkan sebagai metodologi penelitian. Fokus kajian diarahkan pada sistem informasi aplikasi inventaris suku cadang berbasis web. Prinsip dan pedoman ISO 31000, yang telah diakui secara internasional, sangat relevan untuk digunakan dalam konteks ini. Manajemen risiko didefinisikan sebagai proses sistematis untuk mengidentifikasi, menganalisis, menilai, dan mengendalikan risiko yang berpotensi terjadi dalam suatu organisasi. Tahapan-tahapan tersebut meliputi penilaian risiko (*risk assessment*), yang mencakup identifikasi risiko, analisis risiko, dan evaluasi risiko, sehingga menghasilkan rekomendasi pengelolaan risiko yang tepat bagi bisnis (Andika & Wijaya, 2022).



Gambar 4. Tahapan Manajemen Risiko

1) Penilaian Risiko (*Risk Assessment*)
Tahap penilaian risiko (*risk assessment*) dilakukan melalui tiga langkah utama sesuai dengan pedoman manajemen risiko ISO 31000. Langkah-langkah tersebut meliputi identifikasi risiko (*risk identification*), analisis risiko (*risk analysis*), dan evaluasi risiko (*risk evaluation*). Proses ini bertujuan untuk memberikan pemahaman yang sistematis terhadap risiko yang mungkin terjadi, guna mendukung pengambilan keputusan strategis yang tepat.
2) Identifikasi Risiko
Langkah pertama dalam penilaian risiko adalah melakukan identifikasi risiko secara menyeluruh untuk menentukan apakah risiko yang terkait dengan aplikasi inventaris suku cadang dapat diterima. Dalam proses ini, wawancara dilakukan dengan tiga pihak utama yang terlibat dalam pengembangan aplikasi, yaitu *project administrator*, pemrogram, dan desainer antarmuka pengguna (UI/UX). Hasil wawancara digunakan untuk menyusun daftar risiko yang akan dianalisis lebih lanjut menggunakan pendekatan kuantitatif maupun kualitatif.

Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id
E-ISSN: 2723-7079 | P-ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214

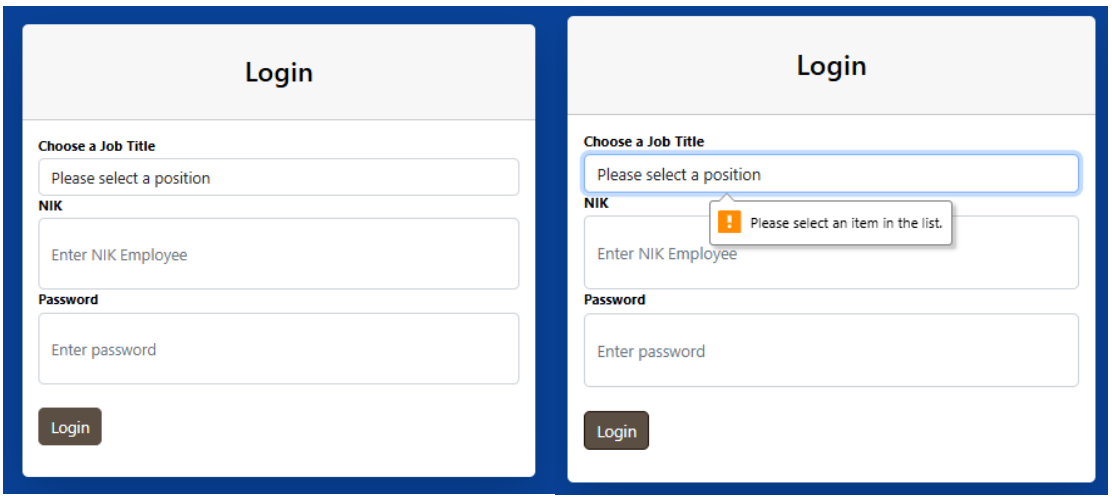
3) Analisis Risiko
Setelah daftar risiko disusun, tahap berikutnya adalah melakukan analisis risiko. Analisis ini mencakup dua pendekatan utama, yaitu kualitatif dan kuantitatif, yang dijelaskan sebagai berikut:
   a) Analisis Kualitatif
   Analisis kualitatif dilakukan dengan menilai risiko menggunakan skor tertentu berdasarkan kemungkinan (*likelihood*) dan konsekuensi (*consequence*). Skor ini membantu mengidentifikasi tingkat risiko secara lebih terukur dan memberikan dasar untuk prioritas mitigasi.
   b) Analisis Kuantitatif
   Analisis kuantitatif dilakukan dengan menghitung dampak dari setiap risiko. Metode ini menggunakan formula tertentu untuk mengidentifikasi faktor-faktor risiko dan mendukung kepala proyek dalam menempatkan sumber daya pada risiko yang memiliki prioritas tertinggi.
4) Evaluasi Risiko
Tahap evaluasi risiko melibatkan analisis menyeluruh terhadap risiko yang telah diidentifikasi dan dianalisis sebelumnya. Proses ini menggunakan skor risiko untuk menentukan tingkat keseluruhan risiko serta mengidentifikasi potensi dampak yang mungkin terjadi. Evaluasi ini bertujuan untuk menetapkan langkah-langkah mitigasi yang dapat dilakukan guna meminimalkan efek negatif dari risiko terhadap keberhasilan proyek. Selain itu, hasil evaluasi ini menjadi dasar bagi pengambilan keputusan strategis untuk memastikan bahwa proyek berjalan sesuai rencana dan mencapai target yang telah ditetapkan.
5) Mitigasi Risiko
Pada tahap mitigasi risiko, peneliti menetapkan langkah-langkah strategis untuk mengurangi atau mengelola berbagai risiko yang muncul selama pelaksanaan proyek pengembangan perangkat lunak. Proses ini dilakukan dengan mengumpulkan opini, masukan, dan rekomendasi dari anggota tim pengembang melalui kuesioner atau wawancara. Tujuannya adalah memastikan bahwa solusi yang diambil efektif dalam meminimalkan dampak negatif risiko terhadap keberhasilan proyek serta mengantisipasi potensi risiko di masa depan. Pendekatan ini melibatkan berbagai perspektif sehingga menghasilkan strategi yang lebih realistis dan sesuai dengan kondisi aktual proyek. Semua langkah strategis tersebut kemudian dirangkum dalam dokumen rencana mitigasi risiko yang berfungsi sebagai panduan utama bagi tim selama proses pengembangan berlangsung (Nuris *et al*., 2022).

3. Hasil dan Pembahasan

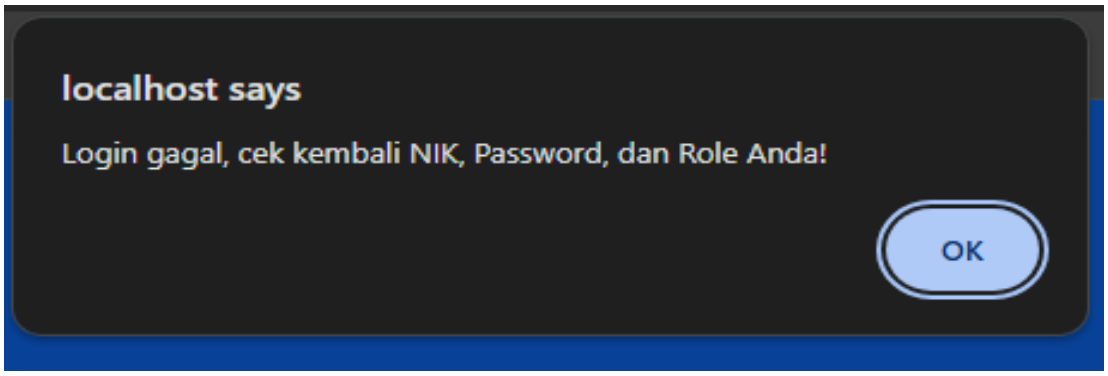
3.1 Hasil

Dalam metode *Black Box Testing*, terdapat dua aspek utama yang menjadi fokus penelitian ini, yaitu *Functionality* dan *Usability*. *Functionality* berfokus pada evaluasi kebutuhan fungsional dari sistem atau perangkat lunak, yang dilakukan dengan menghitung jumlah fitur fungsional yang dirancang untuk aplikasi, kemudian membandingkannya dengan fitur yang berhasil dijalankan selama pengujian. Sementara itu, *Usability* adalah aspek yang berkaitan dengan pengalaman pengguna dalam berinteraksi dengan sistem informasi atau perangkat lunak. Pengujian *usability* sangat penting karena berfungsi untuk mengukur sejauh mana pengguna dapat memahami dan menggunakan antarmuka perangkat lunak secara efektif. Instrumen pengujian *usability* dalam penelitian ini menggunakan kuesioner *Questionnaire Usability Computer System* yang dikembangkan oleh J.R. Lewis sebagai referensi utama untuk mengukur tingkat kegunaan perangkat lunak (Yanti Laily & Triase, 2023). Hasil dari pengujian *functionality* dan *usability* pada sistem informasi aplikasi inventaris suku cadang berbasis web disajikan sebagai berikut:

Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id
E - ISSN: 2723-7079 | P - ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214



Gambar 5. Halaman *Login*



Gambar 6. *Pop Up Error Login*

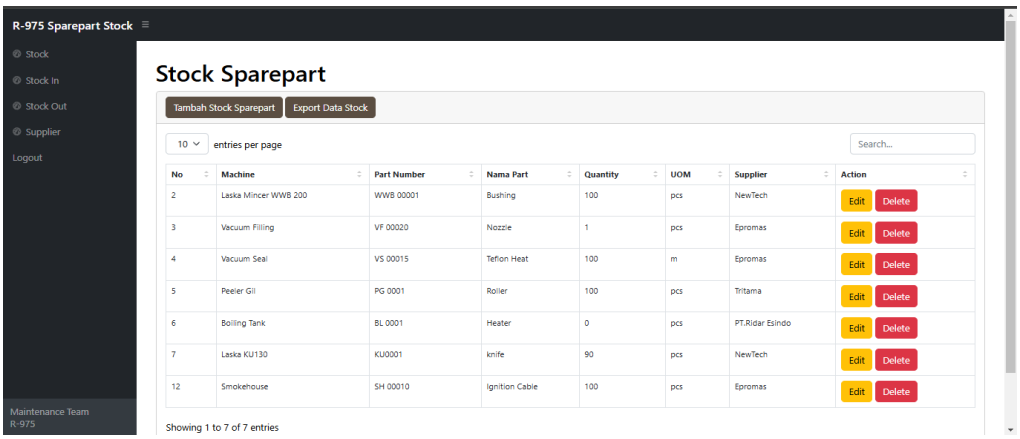
Tabel 1. Pengujian Fitur *Login*

No. ID Test	Deskripsi	Input	Output yang Diharapkan	Output Aktual	Status
001	Login-Input Valid	Role, Username dan Password valid	Menampilkan Halaman Utama	Seusai Harapan	Valid
002	Login-Input Invalid	Role, Username atau Password tidak valid	Menampilkan pesan "Role, Username atau password salah"	Seusai Harapan	Valid
003	Login-Kedua Kolom Kosong	Kosongkan kolom Role, Username dan Password	Menampilkan pesan "Role, Username dan Password harus diisi"	Seusai Harapan	Valid

Tabel 2. Pengujian Halaman Utama

No ID Test	Deskripsi	Input	Outputyang Diharapkan	Output Aktual	Status
001	Tampilkan Menu Utama	Mulai	Menampilkan opsi: 1. Tambah, 2. Edit, 3. Hapus, 4. Cetak	Sesuai Harapan	Valid

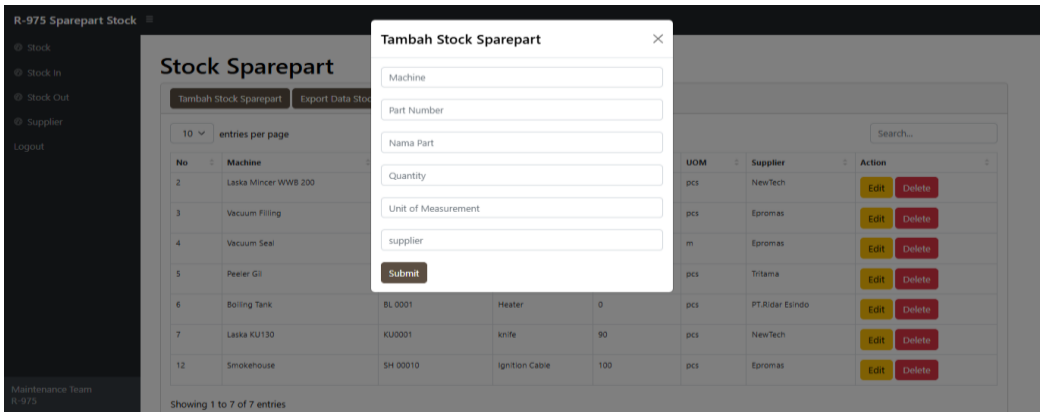
Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id
E - ISSN: 2723-7079 | P - ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214



Gambar 7. Halaman Utama

Tabel 3. Pengujian Fitur Tambah

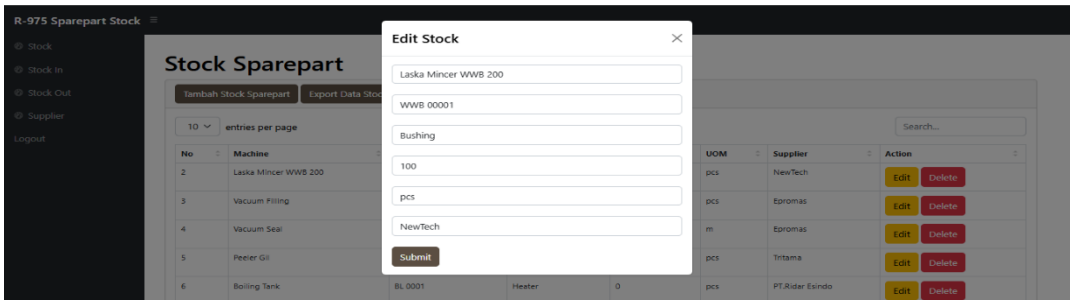
No ID Test	Deskripsi	Input	Output yang Diharapkan	Output Aktual	Status
001	Tambah Data - Akses Form	Pilih ". Tambah"	Menampilkan Form Tambah	Sesuai Harapan	Valid



Gambar 8. Form Modal Fitur Tambah

Tabel 4. Pengujian Fitur Edit

No ID Test	Deskripsi	Input	Output yang Diharapkan	Output Aktual	Status
001	Edit Data - Akses Form	Pilih ". Edit"	Menampilkan Data untuk Diedit	Sesuai Harapan	Valid

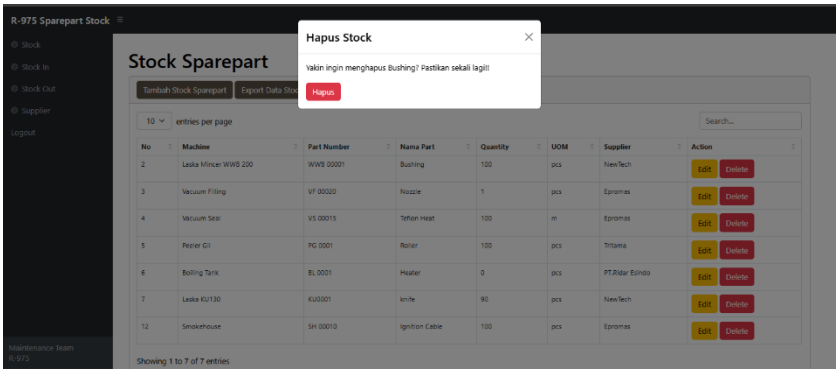


Gambar 9. Form Modal Fitur Edit

Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id
E-ISSN: 2723-7079 | P-ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214

Tabel 5. Pengujian Fitut Hapus

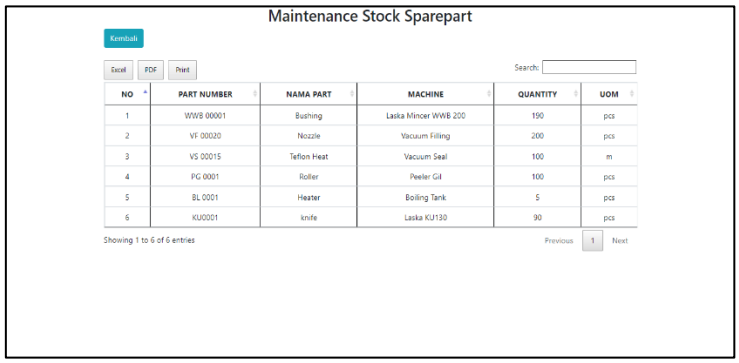
No ID Test	Deskripsi	Input	Output yang Diharapkan	Output Aktual	Status
001	Hapus Data - Konfirmasi	Pilih ". Hapus"	Menampilkan "Konfirmasi Hapus?"	Sesuai Harapan	Valid
002	Hapus Data - Ya	Konfirmasi Ya	Menampilkan "Hapus Valid"	Sesuai Harapan	Valid
003	Hapus Data - Tidak	Konfirmasi Tidak	Kembali ke Menu Utama	Sesuai Harapan	Valid



Gambar 10. *Pop Up* Konfirmasi Fitur Hapus

Tabel 6. Pengujian Fitur Cetak

No ID Test	Deskripsi	Input	Output yang Diharapkan	Output Aktual	Status
001	Cetak Data	Pilih ". Cetak"	Menampilkan "Mencetak..." dan Output Cetak	Sesuai Harapan	Valid

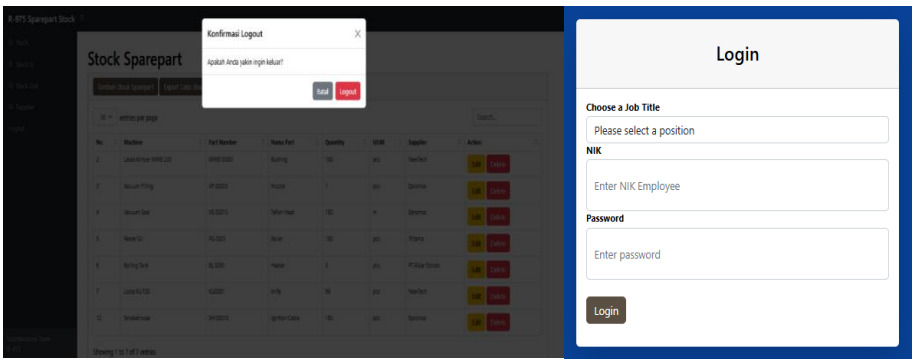


Gambar 11. Halaman Menu Cetak

Tabel 7. Pengujian Fitur *Logout*

No ID Test	Deskripsi	Input	Output yang Diharapkan	Output Aktual	Status
001	Logout - Proses Valid	Klik tombol Logout	Logout Valid dan kembali ke halaman login	Sesuai Harapan	Valid
002	Logout - Membatalkan Logout	Membatalkan konfirmasi logout	Tetap berada di Halaman Utama	Sesuai Harapan	Valid

Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id
E - ISSN: 2723-7079 | P - ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214



Gambar 12. *Pop Up* konfirmasi *Logout* kembali ke halaman *Login*

3.1.1 Manajemen Risiko

1) Identifikasi Risiko dan Dampak Risiko

Hasil identifikasi menunjukkan adanya berbagai potensi risiko yang dapat memengaruhi kelancaran fungsi aplikasi dalam pengembangan perangkat lunak. Risiko utama mencakup gangguan pada aktivitas administrasi akibat kendala teknis pada fitur-fitur seperti antarmuka pengguna (*user interface*), tambah data, edit data, hapus data, dan tombol submit. Selain itu, terdapat risiko pengguna tidak dapat mengakses aplikasi yang disebabkan oleh masalah pada fitur login atau tombol submit, serta pengalaman pengguna yang kurang optimal akibat desain antarmuka yang tidak mendukung kenyamanan penggunaan. Beberapa risiko lainnya termasuk kegagalan dalam memproses data pada fitur tambah dan edit data, tidak adanya bukti konkret yang menunjukkan keberhasilan suatu proses, serta ketiadaan peringatan saat fitur tertentu, seperti logout atau hapus data, dioperasikan. Pada fitur *print preview*, risiko berupa tidak tersedianya data untuk dicetak juga diidentifikasi. Selain itu, penumpukan *cache* pada sistem dapat menyebabkan penurunan efisiensi aplikasi secara keseluruhan. Identifikasi risiko ini menunjukkan kebutuhan akan penerapan strategi mitigasi yang efektif untuk memastikan aplikasi dapat berfungsi dengan baik, mengurangi dampak negatif terhadap operasional, dan meningkatkan pengalaman pengguna secara keseluruhan.

Tabel 8. Identifikasi Risiko

ID Risiko	Sumber Risiko	Deksripsi Risiko
FL001		Input username/password invalid
FL002	Form Login	Login kedua kolom kosong
FL003		Pesan error login tidak informatif
UI001	UI	Tampilan antar muka pada sistem tidak tampil/error
UI002		Tampilan antar muka pada sistem tidak responsif
TD001		Proses tambah data error
TD002	Tambah Data	Form tambah data error
TD003		Notifikasi tambah data berhasil/ error
ED001		Proses edit data error
ED002	Edit Data	Form edit data error
ED003		Notifikasi tambah edit berhasil/ error
HD001		Proses hapus data error
HD002	Hapus Data	Notifikasi hapus error
HD003		Pop up validasi hapus error
PP001	Print Preview	Proses print preview error
FL001	Fitur Logout	Pop up validasi logout error
FL002		Prores logout error
SB001	Submit Button	Button Error

Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id
E - ISSN: 2723-7079 | P - ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214

Tabel 9. Identifikasi Dampak Risiko

ID Risiko	Sumber Risiko	Dampak Risiko
FL001	Form Login	Aktivitas administrasi terhambat
FL002		User tidak dapat mengakses aplikasi
FL003		Informasi yang diterima tidak mudah dipahami
UI001	UI	Aktivitas administrasi terhambat
UI002		User experience yang kurang nyaman
TD001	Tambah Data	Aktivitas administrasi terhambat
TD002		Tidak bisa memproses input data
TD003		Tidak ada bukti congcrete yang menyatakan bahwa proses berhasil
ED001	Edit Data	Aktivitas administrasi terhambat
ED002		Tidak bisa memproses input data
ED003		Tidak ada bukti congcrete yang menyatakan bahwa proses berhasil
HD001	Hapus Data	Aktivitas administrasi terhambat
HD002		Tidak ada bukti congcrete yang menyatakan bahwa proses berhasil
HD003		Tidak ada peringatan terhapap proses yang dilakukan
PP001	Print Preview	Data yang ingin di cetak tidak tersedia
FL001	Fitur Logout	Tidak ada peringatan terhapap proses yang dilakukan
FL002		Penumpukan chace pada sistem
SB001	Submit Button	Tidak dapat mengakses aplikasi dan melakukan proses administrasi

2) Analisis Risiko

Berdasarkan hasil analisis risiko, terdapat dua kriteria utama dalam penilaian risiko, yaitu kemungkinan terjadinya risiko (*likelihood*) dan dampaknya (*impact*). Kriteria *likelihood* mengacu pada frekuensi terjadinya risiko, yang diklasifikasikan menjadi lima tingkatan. Tingkat *Rare* menunjukkan risiko yang sangat jarang terjadi, dengan frekuensi sekitar 3-4 minggu sekali, dan diberikan nilai 1. Tingkat *Unlikely* menggambarkan risiko yang jarang terjadi dalam interval 2-3 minggu, dengan nilai 2. Tingkat *Possible* merujuk pada risiko yang kadang terjadi, yaitu setiap 1-2 minggu sekali, dan diberikan nilai 3. Selanjutnya, tingkat *Likely* menggambarkan risiko yang sering terjadi dalam rentang waktu 1-7 hari, dengan nilai 4. Terakhir, tingkat *Certain* menunjukkan risiko yang pasti akan terjadi setiap hari, dengan nilai tertinggi, yaitu 5. Penilaian terhadap *likelihood* ini memungkinkan untuk menentukan seberapa besar kemungkinan suatu risiko akan mempengaruhi jalannya proyek atau sistem, yang kemudian digunakan untuk menyusun langkah-langkah mitigasi yang tepat.

Tabel 10. Kriteria Kemungkinan (*likelihood*)

Kriteria	Deskripsi	Frekuensi kejadian	Nilai
Rare	Suatu risiko yang hampir tidak pernah terjadi	3 - 4 Minggu	1
Unlikely	Suatu risiko yang jarang terjadi	2 - 3 Minggu	2
Possible	Suatu risiko yang kadang terjadi	1 - 2 Minggu	3
Likely	Suatu risiko yang sering terjadi	1 – 7 Hari	4
Certain	Suatu risiko yang pasti terjadi	Setiap Hari	5

Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id
E-ISSN: 2723-7079 | P-ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214

Sementara itu, tabel *impact* menjelaskan tingkat dampak yang ditimbulkan oleh suatu risiko terhadap kelangsungan proses bisnis atau aktivitas perusahaan. Kategori Insignificant menggambarkan risiko yang memiliki dampak minimal dan tidak mengganggu kelancaran proses bisnis, dengan nilai 1. Minor mengacu pada risiko yang sedikit menghambat aktivitas perusahaan, diberikan nilai 2. *Moderate* merujuk pada risiko yang dapat menghambat sebagian aktivitas bisnis, dengan nilai 3. Major menggambarkan risiko yang memiliki dampak cukup besar, sampai pada penghentian sebagian besar proses bisnis, dengan nilai 4. Pada tingkat tertinggi, *Catastrophic*, risiko ini dapat menghentikan proses bisnis secara keseluruhan, dengan nilai 5. Kedua tabel ini, yaitu *likelihood* dan *impact*, merupakan dasar yang krusial dalam mengevaluasi tingkat keparahan risiko dan merumuskan strategi mitigasi yang efektif untuk meminimalkan dampak negatif terhadap keberhasilan proyek atau kelancaran aktivitas perusahaan.

Tabel 11. Dampak Risiko (*impact*)

Kriteria	Deskripsi	Nilai
Insignificant	Risiko yang dampaknya tidak mengganggu proses bisnis dan jalannya aktivitas dari perusahaan	1
Minor	Risiko yang dampaknya sedikit menghambat proses bisnis dan aktivitas perusahaan	2
Moderate	Risiko yang dampaknya menghambat sebagian jalannya bisnis dan aktivitas perusahaan	3
Kriteria	Deskripsi	Nilai
Major	Risiko yang menghambat seluruh proses bisnis dan aktivitas perusahaan	4
Catastrophic	Risiko yang dampaknya dapat menghentikan proses bisnis dan aktivitas perusahaan secara total	5

Setalah melakukan tahapan menentukan nilai dari kemungkinan nya terjadi risiko (*likelihood*) dan dampak risiko (*impact*), tahapan selanjutnya yaitu menentukan penilaian terhadap kemungkinan dan dampak risiko yang dapat mempengaruhi aplikasi inventori sparepart yang sebelumnya sudah teridentifikasi.

Tabel 12. Penilaian Kemungkinan Risiko menggunakan *Likelihood* dan *Impact*

Kode Risiko	Deksripsi Risiko	LikeliHood	Impact
FL001	Input username/password invalid	2	2
FL002	Login kedua kolom kosong	1	2
FL003	Pesan error login tidak informatif	2	1
UI001	Tampilan antar muka pada sistem tidak tampil/error	2	3
UI002	Tampilan antar muka pada sistem tidak responsif	1	2
TD001	Proses tambah data error	2	3
TD002	Form tambah data error	2	3
TD003	Notifikasi tambah data berhasil/ error	1	2
ED001	Proses edit data error	2	3
ED002	Form edit data error	2	3
ED003	Notifikasi tambah edit berhasil/ error	1	2
HD001	Proses hapus data error	2	3
HD002	Notifikasi hapus error	2	3
HD003	Pop up validasi hapus error	2	4
PP001	Proses print preview error	3	4
FL001	Pop up validasi logout error	2	3
FL002	Prores logout error	3	4
SB001	Button Error	3	5

Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id
E-ISSN: 2723-7079 | P-ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214

3) Evaluasi Risiko

Risiko yang telah diidentifikasi dan dianalisis dimasukkan ke dalam matriks evaluasi risiko untuk menentukan tingkat prioritas penanganannya. Matriks ini dibentuk berdasarkan dua parameter utama, yaitu kemungkinan terjadinya risiko (*likelihood*) dan dampaknya (*impact*). Penyusunan matriks evaluasi risiko dilakukan sesuai dengan parameter yang telah dirumuskan sebelumnya, sehingga menghasilkan representasi yang sistematis dan terstruktur dalam mengukur tingkat keparahan setiap risiko. Pendekatan ini memungkinkan pengkategorian risiko secara lebih jelas, mulai dari risiko dengan dampak kecil hingga risiko dengan potensi kerugian yang signifikan. Dengan pengelompokan yang jelas, matriks ini mempermudah proses pengambilan keputusan dalam merancang strategi mitigasi yang efektif untuk setiap jenis risiko. Evaluasi yang berbasis matriks ini menjadi langkah penting dalam memastikan langkah-langkah mitigasi dapat dilakukan secara prioritas sesuai tingkat urgensi risiko yang dihadapi.

Tabel 13. Matriks Evaluasi Risiko Berdasarkan Kemungkinan (*Likelihood*) dan Dampak (*Impact*)

Likelihood	*Insignificant* (1)	*Minor* (2)	*Moderate* (3)	*Major* (4)	*Catastrophic* (5)
Certain (5)					
Likely (4)					
Possible (3)				PP001 FL002	SB001
Unlikely (2)	FL003	FL001	UI001 TD001 TD002 ED001 ED002 HD001 HD002 FL001	HD003	
Rare (1)		FL002 TD003 UI002 ED003			

Impact

Berdasarkan hasil evaluasi terhadap berbagai kemungkinan risiko yang dianalisis menggunakan matriks yang mempertimbangkan parameter *likelihood* (kemungkinan) dan *impact* (dampak), langkah selanjutnya adalah menyusun risiko-risiko tersebut berdasarkan tingkat keparahannya. Risiko-risiko tersebut diklasifikasikan ke dalam beberapa kategori tingkat risiko, mulai dari tingkat tertinggi (*high*) hingga tingkat terendah (*low*). Klasifikasi ini bertujuan untuk mempermudah proses penentuan prioritas dalam penanganan risiko serta penyusunan strategi mitigasi yang paling tepat dan efisien.

Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id
E - ISSN: 2723-7079 | P - ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214

Pendekatan ini memastikan bahwa risiko-risiko dengan tingkat keparahan yang lebih tinggi ditangani lebih awal, sehingga dampak negatifnya dapat diminimalkan secara efektif.

Tabel 14. Evaluasi Level Kemungkinan Risiko berdasarkan *Likelihood* dan *Impact*

Kode Risiko	Deksripsi Risiko	*LikeliHood*	*Impact*	Level Risiko
FL001	Input username/password invalid	2	2	Low
FL002	Login kedua kolom kosong	1	2	Low
FL003	Pesan error login tidak informatif	2	1	Low
UI001	Tampilan antar muka pada sistem tidak tampil/error	2	3	Medium
UI002	Tampilan antar muka pada sistem tidak responsif	1	2	Low
TD001	Proses tambah data error	2	3	Medium
TD002	Form tambah data error	2	3	Medium
TD003	Notifikasi tambah data berhasil/ error	1	2	Low
ED001	Proses edit data error	2	3	Medium
ED002	Form edit data error	2	3	Medium
ED003	Notifikasi tambah edit berhasil/ error	1	2	Low
HD001	Proses hapus data error	2	3	Medium
HD002	Notifikasi hapus error	2	3	Medium
HD003	Pop up validasi hapus error	2	4	Medium
PP001	Proses print preview error	3	4	Medium
FL001	Pop up validasi logout error	2	3	Medium
FL002	Prores logout error	3	4	Medium
SB001	Button Error	3	5	High

Tabel di atas memuat rincian risiko yang telah diidentifikasi dalam sistem perangkat lunak, mencakup kode risiko, deskripsi, nilai kemungkinan (*likelihood*), dampak (*impact*), dan tingkat risiko (*risk level*). Tingkat risiko diklasifikasikan ke dalam tiga kategori utama, yaitu *Low*, *Medium*, dan *High*, yang ditentukan berdasarkan kombinasi nilai *likelihood* dan *impact*. Risiko dengan tingkat **Low** umumnya memiliki dampak dan kemungkinan yang rendah, seperti kesalahan pada fitur login atau notifikasi. Risiko dengan tingkat *Medium* mencakup masalah yang lebih signifikan, seperti gangguan pada antarmuka pengguna, proses tambah/edit/hapus data, dan validasi. Sementara itu, risiko dengan tingkat *High* ditemukan pada komponen kritis, seperti tombol (*button error*), yang memiliki probabilitas tinggi untuk terjadi serta dampak besar terhadap sistem. Tabel ini berfungsi sebagai alat bantu yang penting dalam menetapkan prioritas mitigasi, memungkinkan pengelolaan risiko dilakukan secara terstruktur dan efektif berdasarkan tingkat keparahan yang telah dianalisis.

4) Mitigasi Risiko

Mitigasi risiko merupakan serangkaian langkah strategis yang dirancang untuk mengurangi atau menghilangkan dampak dari risiko yang berpotensi terjadi. Tahap ini berfokus pada penyusunan rekomendasi yang dapat diimplementasikan guna meminimalkan kemungkinan terjadinya risiko serta memastikan kelancaran proses bisnis perusahaan. Penyusunan strategi mitigasi didasarkan pada tingkat risiko yang telah diidentifikasi dan diklasifikasikan sebelumnya, dimulai dari risiko dengan tingkat keparahan tertinggi (*High*) hingga risiko dengan tingkat keparahan terendah (*Low*). Rekomendasi perlakuan risiko tersebut mencakup berbagai tindakan pencegahan dan penanganan yang relevan untuk setiap kategori risiko.

Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id
E-ISSN: 2723-7079 | P-ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214

Tabel 15. Mitigasi Risiko

Kode Risiko	Deksripsi Risiko	Level Risiko	Mitigasi Risiko
FL001	Input username/password invalid	Low	Reminder User Acoount
FL002	Login kedua kolom kosong	Low	Training User
FL003	Pesan error login tidak informatif	Low	Edit Code Program
UI001	Tampilan antar muka pada sistem tidak tampil/error	Medium	Preventive & Corrective Maintenance Pada Sistem
UI002	Tampilan antar muka pada sistem tidak responsif	Low	Edit Code Program
TD001	Proses tambah data error	Medium	Preventive & Corrective Maintenance Pada Fitur Tambah
TD002	Form tambah data error	Medium	Preventive & Corrective Maintenance Pada Fitur Tambah
TD003	Notifikasi tambah data berhasil/ error	Low	Preventive & Corrective Maintenance Pada Fitur Tambah
ED001	Proses edit data error	Medium	Preventive & Corrective Maintenance Pada Fitur Edit
ED002	Form edit data error	Medium	Preventive & Corrective Maintenance Pada Fitur Edit
ED003	Notifikasi tambah edit berhasil/ error	Low	Preventive & Corrective Maintenance Pada Fitur Edit
HD001	Proses hapus data error	Medium	Preventive & Corrective Maintenance Pada Fitur Hapus
HD002	Notifikasi hapus error	Medium	Preventive & Corrective Maintenance Pada Fitur Edit
HD003	Pop up validasi hapus error	Medium	Preventive & Corrective Maintenance Pada Sistem
PP001	Proses print preview error	Medium	Preventive & Corrective Maintenance Pada Sistem, Database dan Server
FL001	Pop up validasi logout error	Medium	Preventive & Corrective Maintenance Pada Sistem
FL002	Prores logout error	Medium	Preventive & Corrective Maintenance Pada Sistem
SB001	Button Error	High	Preventive & Corrective Maintenance Pada Sistem Fitur Button
TD001	Proses tambah data error	Medium	Preventive & Corrective Maintenance Pada Fitur Tambah
TD002	Form tambah data error	Medium	Preventive & Corrective Maintenance Pada Fitur Tambah
TD003	Notifikasi tambah data berhasil/ error	Low	Preventive & Corrective Maintenance Pada Fitur Tambah
ED001	Proses edit data error	Medium	Preventive & Corrective Maintenance Pada Fitur Edit
ED002	Form edit data error	Medium	Preventive & Corrective Maintenance Pada Fitur Edit

Jurnal Indonesia : Manajemen Informatika dan Komunikasi
https://journal.stmiki.ac.id
E-ISSN: 2723-7079 | P-ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214

3.2 Pembahasan

Penelitian ini menggunakan pendekatan manajemen risiko ISO 31000, yang telah terbukti efektif dalam berbagai studi sebelumnya, termasuk penelitian yang dilakukan oleh Andika dan Wijaya (2022) serta Ecleas (2021). Pendekatan ini memberikan kerangka kerja sistematis untuk mengidentifikasi, menganalisis, mengevaluasi, dan mengelola risiko yang berpotensi memengaruhi keberhasilan proyek teknologi informasi. Dalam konteks penelitian ini, pendekatan tersebut diterapkan pada sistem informasi aplikasi *inventory sparepart* berbasis web untuk meningkatkan keandalan dan efisiensi operasional. Tahap identifikasi risiko, yang merujuk pada penelitian Nuris *et al.* (2022), memungkinkan identifikasi berbagai risiko yang berpotensi mengganggu fungsi aplikasi. Risiko tersebut meliputi masalah teknis pada fitur login, tambah/edit/hapus data, tombol submit, dan antarmuka pengguna. Selain itu, risiko seperti kesalahan pada fitur cetak (*print preview*) dan penumpukan *cache* yang memengaruhi kinerja aplikasi juga diidentifikasi. Proses ini sejalan dengan temuan Husein dan Imbar (2015), yang menekankan pentingnya identifikasi risiko secara komprehensif untuk mendukung pengelolaan risiko yang lebih efektif. Pada tahap analisis risiko, dua parameter utama, yaitu *likelihood* dan *impact*, digunakan untuk mengevaluasi tingkat risiko. Penilaian ini mengikuti pendekatan yang dikemukakan oleh Atmojo dan Manuputty (2020), yang mengklasifikasikan risiko ke dalam kategori *Low, Medium,* dan *High* berdasarkan kombinasi dua parameter tersebut. Risiko tingkat *High*, seperti kesalahan tombol (*button error*), memerlukan perhatian prioritas karena memiliki probabilitas tinggi dan dampak signifikan.

Risiko tingkat *Medium* dan *Low*, seperti masalah pada antarmuka pengguna dan notifikasi, meskipun kurang kritis, tetap memerlukan langkah mitigasi untuk memastikan kelancaran operasional. Evaluasi risiko dilakukan menggunakan matriks risiko, yang memungkinkan visualisasi sistematis tingkat keparahan risiko, sebagaimana dijelaskan oleh Ecleas (2021). Matriks ini memberikan gambaran yang jelas tentang risiko yang memerlukan prioritas mitigasi tertinggi, sehingga mempermudah pengambilan keputusan dalam mengalokasikan sumber daya untuk menangani risiko tersebut. Langkah mitigasi risiko yang disusun dalam penelitian ini merujuk pada pendekatan Nuris *et al.* (2022) dan melibatkan kombinasi langkah preventif dan korektif. Untuk risiko tingkat *High,* seperti *button error*, disarankan dilakukan *preventive* dan *corrective maintenance* pada sistem. Risiko tingkat *Medium*, seperti gangguan pada fitur tambah/edit/hapus data, dapat diminimalkan melalui pelatihan pengguna dan pembaruan kode program. Risiko tingkat *Low*, seperti kesalahan notifikasi, ditangani dengan langkah perbaikan sederhana. Hasil pengujian *Black Box* menunjukkan bahwa pendekatan ini efektif dalam mengevaluasi aspek fungsional dan kegunaan aplikasi. Metode ini, sebagaimana dikemukakan oleh Maspupah (2022) dan Ismail & Efendi (2020), memiliki keunggulan dalam mengidentifikasi masalah tanpa perlu mengakses kode sumber, sehingga cocok untuk sistem yang kompleks. Dengan kombinasi manajemen risiko berbasis ISO 31000 dan pengujian *Black Box*, penelitian ini menunjukkan bahwa pengelolaan risiko dapat dilakukan secara sistematis dan efektif untuk meningkatkan kualitas dan keandalan sistem perangkat lunak.

4. Kesimpulan

Berdasarkan hasil pengujian terhadap fitur aplikasi, termasuk fitur login, fitur utama, dan fitur logout, dapat disimpulkan bahwa seluruh fungsi yang diuji berjalan sesuai dengan spesifikasi dan harapan. Pengujian ini menggunakan metode *Black Box*, yang berfokus pada evaluasi input dan output tanpa mempertimbangkan logika internal aplikasi. Metode ini efektif dalam memastikan bahwa aplikasi dapat beroperasi sebagaimana mestinya sesuai dengan kebutuhan pengguna. Dalam analisis risiko terhadap kualitas sistem *inventory maintenance* menggunakan kerangka kerja ISO 31000, ditemukan 18 potensi risiko yang teridentifikasi. Proses analisis risiko ini dilakukan melalui tiga tahapan utama, yaitu identifikasi risiko (*risk identification*), analisis risiko (*risk analysis*), dan evaluasi risiko (*risk evaluation*). Risiko-risiko tersebut kemudian diklasifikasikan berdasarkan matriks kemungkinan (*likelihood*) dan dampak (*impact*). Hasilnya menunjukkan bahwa terdapat 6 risiko pada

E - ISSN: 2723-7079 | P - ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214

kategori *Low Risk*, 11 risiko pada kategori *Medium Risk*, dan 1 risiko pada kategori *High Risk*. Hasil penelitian ini menegaskan bahwa aplikasi telah dirancang dengan baik dan berfungsi secara optimal sesuai spesifikasi yang ditetapkan. Melalui pengujian *Black Box*, disimpulkan bahwa pengembangan aplikasi telah berhasil memenuhi kebutuhan pengguna dan memberikan kemudahan dalam interaksi dengan sistem. Untuk mengatasi kemungkinan risiko yang teridentifikasi, perusahaan telah menerapkan langkah-langkah strategis seperti pelatihan pengguna (*user training*), pemeliharaan preventif (*preventive maintenance*), pemeliharaan korektif (*corrective maintenance*), serta pembaruan kode program sebagai tindakan korektif ketika terjadi kesalahan. Langkah-langkah ini diharapkan dapat meminimalkan potensi kesalahan dan memastikan proses bisnis berjalan dengan lancar.

5. Daftar Pustaka

Andika, D., & Wijaya, A. (2022). Manajemen risiko teknologi informasi menggunakan framework ISO 31000: 2018 pada PT. Trust Lerinvital Timur. *Jurnal Mnemonic*, *5*(2), 111-118. https://doi.org/10.36040/mnemonic.v5i2.4778.

Atmojo, S. A., & Manuputty, A. D. (2020). Analisis Manajemen Risiko Teknologi Informasi Menggunakan ISO 31000 pada Aplikasi AHO Office. *JATISI (Jurnal Teknik Informatika Dan Sistem Informasi)*, *7*(3), 546-558. https://doi.org/10.35957/jatisi.v7i3.525.

Cahyaningtyas, R., & Sigi, A. L. (2015). Perancangan Aplikasi Pengelolaan Rumah Tangga Laboratorium Komputer STT-PLN. *Kilat*, *4*(1), 10-16. https://doi.org/10.33322/kilat.v4i1.697.

Ecleas, J., & Manuputty, A. D. (2021). Analisis Manajemen Risiko Teknologi Informasi Software PEGA Menggunakan ISO 31000. *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, *8*(1), 209-224. https://doi.org/10.35957/jatisi.v8i1.601.

Ferrore, T. S., & Yulianti, D. T. (2016, August). Pengukuran Tingkat Kematangan Manajemen Risiko Sistem X pada PT. Y Menggunakan Framework Risk IT Domain Risk Governance. In *Seminar Nasional Aplikasi Teknologi Informasi (SNATi)*.

Husein, G. M., & Imbar, R. V. (2015). Analisis Manajemen Risiko Teknologi Informasi Penerapan Pada Document Management System di PT. JABAR TELEMATIKA (JATEL). *Jurnal Teknik Informatika dan Sistem Informasi*, *1*(2). https://doi.org/10.28932/jutisi.v1i2.575.

Ismail, I., & Efendi, J. (2021). Black-Box Testing: Analisis Kualitas Aplikasi Source Code Bank Programming. *Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi)*, *5*(1), 1-6. https://doi.org/10.35870/jtik.v5i1.148.

Jordan, Y., Pranatawijaya, V. H., & Widiatry, W. (2024). ANALISIS KUALITAS DAN PENERAPAN SOFTWARE QUALITY ASSURANCE PADA APLIKASI DANA MENGGUNAKAN MODEL ISO/IEC 9126. *JATI (Jurnal Mahasiswa Teknik Informatika)*, *8*(4), 7687-7691. https://doi.org/10.36040/jati.v8i4.10048.

Laily, D. Y., & Triase, T. (2023). Implementasi Quality Assurance Dalam Pengembangan Aplikasi Ourticle Berbasis Android. *Sibatik Journal: Jurnal Ilmiah Bidang Sosial, Ekonomi, Budaya, Teknologi, Dan Pendidikan*, *2*(3), 793-804. https://doi.org/10.54443/sibatik.v2i3.664.

E - ISSN: 2723-7079 | P - ISSN: 2776-8074
Vol. 6 No. 1 (2025) | Januari
OPEN ACCESS
https://doi.org/10.35870/jimik.v6i1.1214

Maspupah, A. (2024). Literature Review: Advantages And Disadvantages Of Black Box And White Box Testing Methods. *Jurnal Techno Nusa Mandiri*, *21*(2), 151-162. https://doi.org/10.33480/techno.v21i2.5776.

Nazir, M. (2020). Software Quality Assurance and Android Application Development: A Comparison among Traditional and Agile Methodology. *Lahore Garrison University Research Journal of Computer Science and Information Technology*, *4*(4), 1-29. https://doi.org/10.54692/lgurjcsit.2020.0404105.

Nuris, A. M., Maharani, A., & Rachmadita, R. N. (2021). Analisis Risiko Proyek Pengembangan Perangkat Lunak Menggunakan Kerangka Kerja ISO 31000. *Jurnal Metris*, *22*(02), 73-81.

Patel, K. (2024). Exploring the Combined Effort Between Software Testing and Quality Assurance: A Review of Current Practices and Future. *Int. Res. J. Eng. Technol*, *11*(09), 522-529.

Praniffa, A. C., Syahri, A., Sandes, F., Fariha, U., Giansyah, Q. A., & Hamzah, M. L. (2023). Pengujian Black Box Dan White Box Sistem Informasi Parkir Berbasis Web Black Box and White Box Testing of Web-Based Parking Information System. *J. Test. dan Implementasi Sist. Inf*, *1*(1), 1-16.

Rahmah, G. M. (2019). Analisis manajemen risiko penerapan sistem informasi di politeknik stmi Jakarta. *JURNAL TEKNOLOGI dan MANAJEMEN*, *17*(2), 65-77.