

1. Pendahuluan

1.1 Latar Belakang

Perkembangan teknologi informasi yang pesat mendorong berbagai sektor, terutama sektor keuangan, untuk memanfaatkan aplikasi berbasis digital guna meningkatkan efisiensi dan keandalan. Berbagai aktivitas kini semakin mengandalkan teknologi digital. Termasuk dalam transaksi digital global yang mengalami pertumbuhan signifikan hingga mencapai 118% selama 5 tahun. Tahun terakhir menurut Statista pada 2021. Di Indonesia, pertumbuhan sektor digital mengalami peningkatan yang signifikan [1].

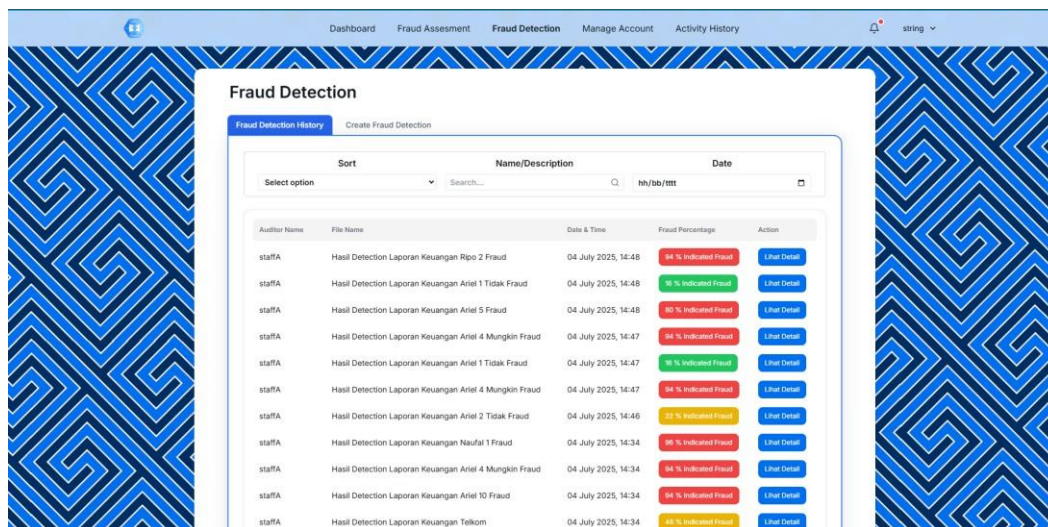
Fraud Deterrence Propeller (FDP) adalah salah satu aplikasi yang dirancang untuk mendeteksi potensi kecurangan dalam laporan keuangan. FDP membantu organisasi mengidentifikasi potensi kecurangan lebih awal, sehingga dapat mengurangi kerugian finansial dan reputasi. Aplikasi ini dikembangkan sebagai respons terhadap meningkatnya kasus kecurangan keuangan dewasa ini. Dalam investigasi penipuan di Indonesia, Association of Certified Fraud Examiners (ACFE) Indonesia (2020) menemukan jumlah kasus penipuan di Indonesia sebanyak 239 kasus, yang terdiri dari 167 kasus korupsi, 50 kasus penyalahgunaan aset, dan 22 kasus penipuan. Terungkap bahwa itu adalah Laporan Penipuan Penyalahgunaan Keuangan [2].

Pada Aplikasi Fraud Deterrence Propeller V.1 (FDP V.1) masih memiliki keterbatasan pada proses asesmen. Aplikasi Fraud Deterrence Propeller V.1 (FDP V.1) hanya bisa untuk memeriksa satu kuesioner saja dalam satu asesmen yang ada, kekurangan ini menyebabkan efisiensi evaluasi dan fleksibilitas berkurang. Kekurangan tersebut teridentifikasi berdasarkan hasil evaluasi yang sudah dilakukan bersama tim pengembang Fraud Deterrence Propeller V.1 (FDP V.1)



Gambar 1 FDP V.1

Pengembangan aplikasi Fraud Deterrence Propeller V.2 (FDP V.2) dilakukan oleh tim yang terdiri dari pengembang front-end, back-end, ui/ux, dan analis sistem. Setiap sprint dalam metode agile dirancang untuk menghasilkan iterasi aplikasi yang bisa diuji secara menyeluruh. Pendekatan Agile berfokus pada kolaborasi antara pemangku kepentingan dan pengembang serta mendorong tim pengembang untuk menjadi tim yang mengorganisasi diri sendiri [3]. Proses pengembangan dimulai dari tahap System Analysis, yang melibatkan pembuatan Software Requirements Specification (SRS). SRS berisi deskripsi lengkap tentang fitur FDP V.2, seperti deteksi laporan keuangan, antarmuka pengguna yang ramah, dan keamanan tingkat tinggi. SRS juga mencakup spesifikasi teknis, seperti integrasi API dan enkripsi data.



Gambar 2 FDP V.2

dalam pengembangan perangkat lunak, salah satu aspek penting yang tidak dapat diabaikan adalah pengujian perangkat lunak. Dengan melakukan pengujian, pengembang dapat memastikan bahwa setiap fitur dari aplikasi dapat bekerja sesuai dengan harapan sehingga meningkatkan kepuasan pengguna dan mengurangi risiko kerugian akibat kegagalan aplikasi [4]. Pengujian pada FDP V.2 sangat diperlukan untuk memastikan aplikasi memenuhi requirement yang telah ditetapkan. Jika requirement gagal terpenuhi, aplikasi berisiko mengalami berbagai masalah, seperti kesalahan dalam mendeteksi kecurangan, atau bahkan hilangnya kepercayaan pengguna.

Selain itu, FDP V.2 juga mengadopsi fitur dari aplikasi sebelumnya, di mana beberapa fungsi lama tetap dipertahankan untuk menjamin kelangsungan operasional. Oleh karena itu, pengujian yang komprehensif mencakup pembuatan test case dan test plan yang efektif menjadi langkah yang tidak dapat diabaikan.

Penggunaan black box testing dan white box testing diterapkan untuk mengidentifikasi potensi masalah pada FDP V.2. Test case dan test plan dirancang berdasarkan skenario penggunaan nyata, mencakup berbagai kemungkinan interaksi pengguna dengan aplikasi..

Urgensi pengujian perangkat lunak ini menjadi sangat krusial, mengingat aplikasi FDP V.2. bertujuan untuk mendeteksi potensi kecurangan yang dapat berdampak luas jika terjadi kesalahan. Dengan pengujian yang baik, aplikasi dapat memberikan hasil yang andal, melindungi data keuangan, serta meningkatkan kepercayaan pengguna terhadap sistem [5].

1.2 Topik dan Batasannya

1. Bagaimana merancang dan menghasilkan rencana pengujian perangkat lunak serta kasus uji yang sesuai dengan skenario penggunaan aplikasi FDP V.2 ?
2. Bagaimana hasil pengujian perangkat lunak pada aplikasi FDP V.2 dapat memastikan aplikasi sesuai dengan requirement?.

1.3 Batasan Masalah

1. Pengujian white box unit testing akan berfokus pada pengujian modul fraud detection yang digunakan pada aplikasi fdp V2 dan pengujian modul assessment pada aplikasi FDP V.2 dan Pengujian white box integration testing dilakukan pada seluruh modul yang saling terintegrasi, mencakup fitur baru, fitur yang diperbarui, serta fitur lama yang berkaitan, misalnya fraud detection, fraud assessment, dan notifikasi.
2. Pengujian black box akan menguji fitur terbaru FDP V.2 dan fitur lama yang memiliki kaitan dengan fitur baru

1.4 Tujuan

1. Merancang dan menerapkan langkah-langkah pengujian perangkat lunak untuk aplikasi FDP V.2 (Fraud Deterrence Propeller) guna memastikan aplikasi bekerja sesuai dengan spesifikasi dan kebutuhan pengguna.
2. Mengembangkan dan menguji kasus uji (test case) berdasarkan skenario penggunaan nyata, dengan tujuan untuk mengidentifikasi dan mengatasi potensi kesalahan dalam aspek fungsionalitas, FDP V.2.
3. Mengevaluasi aplikasi FDP V.2 melalui metode pengujian black box dan white box testing, serta memastikan aplikasi memenuhi requirement yang telah ditentukan tanpa kerentanan kritis yang dapat memengaruhi operasionalnya

1.5 Organisasi Tulisan

Bagian pertama pada jurnal ini mencakup latar belakang, topik batasan, dan tujuan pada bagian kedua berisi penelitian terdahulu yang menjadi acuan dalam pengujian, dan penjelasan teoritis yang berisi penjelasan teknik pengujian yang akan digunakan. Pada bagian ketiga membahas terkait metode penelitian yang digunakan. metode penelitian mencakup langkah langkah yang akan diambil pada saat pengujian. Pada bagian ke empat berisi hasil dari penelitian yang akan berisi pengumpulan data dan analisis data. Pada bagian ke lima akan berisi kesimpulan dari penelitian yang telah dilakukan