

BAB I PENDAHULUAN

1.1 Latar Belakang

Dalam dunia pengembangan perangkat lunak, pengujian merupakan langkah krusial yang tidak boleh diabaikan. Jika sebuah aplikasi tidak diuji secara menyeluruh sebelum peluncuran, risiko terjadinya kesalahan atau cacat sistem akan meningkat secara signifikan. Seiring dengan perkembangan teknologi yang semakin pesat dan persaingan pasar yang semakin ketat, para pengembang perangkat lunak dihadapkan pada tantangan untuk menghasilkan produk berkualitas tinggi dalam waktu yang lebih singkat. Banyak perusahaan belum sepenuhnya memahami pentingnya pengujian dalam siklus pengembangan perangkat lunak, yang dapat menyebabkan produk berkualitas rendah dan ketidakpuasan pengguna. Akibatnya, banyak masalah dan cacat yang baru terungkap setelah produk diluncurkan, yang dapat berdampak pada kerugian finansial serta reputasi perusahaan.

Pengujian perangkat lunak merupakan elemen krusial dalam siklus pengembangan perangkat lunak untuk memastikan kualitas, keandalan, dan keamanan sistem [1]. Pengujian manual melibatkan keterlibatan manusia secara langsung untuk mengevaluasi perangkat lunak berdasarkan skenario tertentu, sedangkan pengujian otomatis memanfaatkan perangkat lunak untuk menjalankan pengujian secara otomatis sesuai skenario yang telah ditentukan [2].

PT Adhikari Inovasi Indonesia merupakan perusahaan yang bergerak di bidang teknologi dengan fokus pada pengembangan perangkat lunak, termasuk layanan undangan digital, sistem *point of sales*, dan *photobooth*. Dalam upaya meningkatkan efisiensi pengujian, PT Adhikari Inovasi Indonesia telah mulai mengadopsi pengujian otomatis menggunakan *Katalon Studio* sebagai bagian dari strategi peningkatan kualitas produk. *Katalon Studio* dipilih karena kemampuannya dalam mendukung pengujian berbasis *UI*, *API*, dan *mobile* dengan antarmuka yang *user-friendly* serta integrasi yang fleksibel dengan berbagai alat lainnya.

Dengan menerapkan *Automation testing* menggunakan *Katalon Studio*, diharapkan waktu yang dibutuhkan untuk proses pengujian dapat dikurangi secara signifikan, sementara akurasi dan keandalan pengujian meningkat. Selain itu, pengujian otomatis memungkinkan pengujian yang lebih menyeluruh terhadap setiap perubahan dalam aplikasi sebelum dirilis ke pengguna. Dengan demikian, PT Adhikari Inovasi Indonesia dapat menghadirkan produk yang lebih stabil dan berkualitas, meningkatkan kepuasan pelanggan, serta memperkuat reputasi perusahaan di industri teknologi.

Laporan ini akan membahas lebih lanjut tentang implementasi pengujian otomatis menggunakan *Katalon Studio* pada Website Undangan Digital *by awh* di PT Adhikari Inovasi Indonesia , termasuk langkah-langkah yang diterapkan, manfaat yang diperoleh, serta tantangan yang dihadapi dalam penerapannya. Berdasarkan hasil pengujian yang dilakukan terhadap beberapa fitur utama seperti *createdata* tamu, *update* data tamu, *search* data tamu, dan *delete* data tamu, didapatkan perbandingan antara metode *Manual testing* dan *Automation testing*.

1.2 Rumusan Masalah dan Solusi

Berdasarkan latar belakang diatas maka dapat dinyatakan rumusan masalahnya adalah bagaimana cara meningkatkan efisiensi pengujian aplikasi di PT Adhikari Inovasi Indonesia?

Adapun solusi yang diusulkan adalah mengadopsi pengujian otomatis menggunakan *Katalon Studio*. Penggunaan *tools* ini diharapkan dapat meningkatkan efisiensi dan akurasi pengujian aplikasi, sehingga mengurangi risiko kesalahan manusia, menghemat waktu, dan memastikan peluncuran aplikasi yang lebih andal dan berkualitas.

1.3 Tujuan

Berdasarkan sub bab sebelumnya, tujuan yang ingin dicapai dalam proyek ini adalah:

1. Mengalisis kebutuhan pengujian dalam sistem perangkat lunak di PT Adhkari Inovasi Indonesia.
2. Mengimplementasikan *Katalon Studio* dalam lingkungan pengembangan perangkat lunak di PT Adhikari Indonesia untuk meningkatkan efisiensi pengujian aplikasi.

1.4 Batasan Masalah

Batasan masalah pada laporan ini berfokus pada:

1. Implementasi pengujian antar muka secara otomatis menggunakan *Katalon Studio* di PT Adhikari Inovasi Indonesia.
2. Pengujian yang dilakukan hanya difokuskan pada website undangan digital *By awh*, sehingga tidak mencakup pengujian pada produk atau layanan lain yang dikembangkan oleh PT Adhikari Inovasi Indonesia.

1.5 Metodologi

Metodologi yang digunakan dalam tugas akhir ini mengacu pada *Software testing life cycle (STLC)*. *STLC* merupakan serangkaian tahapan sistematis yang dilakukan dalam proses pengujian perangkat lunak, mulai dari perencanaan hingga penutupan pengujian. Metodologi ini dipilih karena mampu memberikan struktur kerja yang jelas

dalam proses *Quality assurance*, khususnya pada pengujian otomatis antarmuka aplikasi berbasis web [3].



Gambar 1. 1 Software Testing Life Cycle

Adapun tahapan-tahapan dalam *STLC* yang digunakan dalam penelitian ini adalah sebagai berikut:

1. *Requirement Analysis*

Requirement analysis adalah tahap awal dalam siklus pengujian perangkat lunak yang berfokus pada pemahaman kebutuhan sistem dan fitur yang akan diuji. Pada tahap ini, tim QA menganalisis dokumen kebutuhan untuk menentukan apa saja yang perlu diuji dan mengidentifikasi potensi risiko dalam proses pengujian [4].

2. *Test Planning*

Test planning merupakan proses perencanaan strategis untuk seluruh aktivitas pengujian. Tahap ini mencakup penentuan pendekatan pengujian, pemilihan *tools*, estimasi waktu dan sumber daya, penetapan peran dan tanggung jawab, serta definisi kriteria keberhasilan pengujian [5].

3. *Test Case Design*

Test case design adalah proses pembuatan dan dokumentasi skenario pengujian berdasarkan kebutuhan fungsional. *Test case* mencakup langkah-langkah pengujian, data uji, kondisi awal, dan ekspektasi hasil untuk memastikan semua fungsi sistem diuji secara menyeluruh [6].

4. *Test Environment Setup*

Test environment setup adalah tahap penyiapan lingkungan teknis yang diperlukan untuk melaksanakan pengujian. Ini meliputi konfigurasi perangkat keras, perangkat lunak, jaringan, serta *tools* pendukung agar pengujian dapat dilakukan dalam kondisi stabil dan sesuai kebutuhan [7].

5. *Test Execution*

Test execution merupakan pelaksanaan *test case* yang telah disusun sebelumnya. Pada tahap ini dilakukan pengujian manual atau otomatis, dan hasil aktual dibandingkan dengan hasil yang diharapkan untuk mendeteksi adanya cacat atau penyimpangan [8].

6. *Test Cycle Closure*

Test cycle closure adalah tahap akhir dalam siklus pengujian yang mencakup evaluasi terhadap hasil pengujian, dokumentasi *bug*, penyusunan laporan akhir, serta analisis efektivitas proses pengujian. Tahap ini juga mencakup pembelajaran dan perbaikan untuk proses pengujian di masa mendatang [9].

Dengan menerapkan *STLC*, proses pengujian dapat dilakukan secara sistematis, terdokumentasi, dan terukur, sehingga mendukung peningkatan kualitas perangkat lunak secara menyeluruh.

1.6 Penjadwalan Kerja

Dalam rangka mendukung implementasi pengujian sistem menggunakan metode manual dan otomatisasi di PT Adhikari Inovasi Indonesia, penjadwalan pelaksanaan kerja disusun secara terstruktur. Pelaksanaan magang ini berlangsung dari 12 Agustus 2024 hingga 20 Juni 2025. Penjadwalan mencakup dua aspek, yaitu jadwal harian aktivitas kerja yang dilaksanakan selama satu minggu kerja, serta jadwal mingguan yang merinci aktivitas pengembangan keterampilan dan penerapan pengujian, mulai dari studi dokumentasi hingga penerapan *Automation testing* menggunakan *Selenium* dan *Katalon Studio*. Penyusunan jadwal ini bertujuan untuk memastikan seluruh tahapan pengujian dapat berjalan sistematis, terarah, dan sesuai dengan kebutuhan perusahaan.

Berikut ini jadwal pelaksanaan magang dalam satuan waktu minggu:

Tabel 1. 1 Jadwal Kerja

Hari	Jadwal Kerja	Keterangan
Senin	08.00 – 17.00	Aktivitas kantor normal
Selasa	08.00 – 17.00	Aktivitas kantor normal
Rabu	08.00 – 17.00	Aktivitas kantor normal
Kamis	08.00 – 17.00	Aktivitas kantor normal
Jumat	08.00 – 17.00	Aktivitas kantor normal

Berikut ini jadwal pelaksanaan magang dalam kurun waktu 10 bulan:

Tabel 1. 2 Jadwal Pelaksanaan Magang

No	Deskripsi Kerja	Bulan									
		1	2	3	4	5	6	7	8	9	10
1	Studi dokumentasi dan teori dasar <i>testing</i>										
2	Implementasi pengujian manual dan dokumentasi <i>testing</i>										
3	Eksplorasi dasar <i>Automation testing</i>										
4	<i>Automation testing</i> menggunakan <i>Selenium</i>										
5	Eksplorasi <i>Automation testing</i> dengan <i>Katalon Studio</i>										
6	Penerapan <i>Automation testing</i> dengan <i>Katalon Studio</i>										
7	<i>Maintenance, Regression Testing</i> , dan evaluasi berkala										