

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Penelitian ini menggunakan beberapa sumber pustaka yang berhubungan dengan kasus atau metode yang akan diteliti. Diantaranya yaitu :

Penelitian Ardi & Putro (2020). Pengujian Black Box Aplikasi Mobile Menggunakan *Katalon Studio* (Studi Kasus: ACC Partner PT. Astra Sedaya Finance). Penelitian ini menunjukkan bahwa pengujian fungsional aplikasi menggunakan metode black box dengan alat otomatisasi *Katalon Studio* memberikan hasil yang lebih baik dalam mendeteksi kesalahan dibandingkan dengan pengujian manual. Penggunaan *Katalon Studio* meningkatkan efisiensi dan akurasi dalam proses pengujian aplikasi mobile.

Penelitian oleh Zulianto et al (2021). Memanfaatkan *Katalon Studio* untuk otomatisasi pengujian black box pada aplikasi iPosyandu. Studi ini menemukan bahwa pengujian otomatis menghasilkan waktu eksekusi yang lebih cepat dibandingkan dengan pengujian manual, dengan peningkatan kecepatan sekitar 2,54 kali. Selain itu, pengujian otomatis juga membantu meminimalisir kesalahan manusia dan meningkatkan efisiensi proses pengujian.

Salah satu studi oleh Tempomona dan Hendry (2022). Menerapkan metode *black box testing* menggunakan *Katalon Studio* pada aplikasi absensi di PT Astra Sedaya Finance. Hasilnya menunjukkan bahwa pengujian otomatis dengan *Katalon Studio* berhasil mempersingkat waktu pengujian berulang dengan data yang banyak serta menghasilkan laporan pengujian yang rinci dan mudah dipahami.

Penelitian Dewi, Rika L (2023). Perbandingan Hasil *Automation Testing Robot Framework Dan Katalon Studio* Pada E-Learning SD Negeri 1 Dibal Boyolali. Pengujian otomatis dengan *Katalon Studio* berhasil menguji aplikasi E-Learning SD Negeri 1 Dibal Boyolali.

Penelitian oleh Fadillah & Baidawi (2023). Analisis Efektivitas Penggunaan *Katalon Studio* untuk Automasi Pengujian Application Programming Interface (API) pada Sistem Informasi Telekomunikasi . Studi ini menemukan bahwa

Katalon Studio efektif dalam mengotomatisasi pengujian API pada sistem informasi telekomunikasi. Penggunaan *Katalon Studio* berhasil mengurangi waktu pengujian secara signifikan, meningkatkan jumlah bug yang terdeteksi, dan meningkatkan kepuasan penguji berkat kemudahan penggunaan dan fitur-fitur yang disediakan. Rangkuman penelitian dapat dilihat pada tabel 2.1.

Tabel 2.1. Perbandingan Tinjauan Pustaka

Penulis	Tools	Hasil
Ardi & Putro (2020)	<i>Katalon Studio</i>	Hasil penelitian menunjukkan bahwa penerapan pengujian fungsional dengan metode Black Box menggunakan <i>Katalon Studio</i> menghasilkan laporan pengujian yang lebih komprehensif dan efektif dibandingkan dengan pengujian manual. Otomatisasi pengujian ini berhasil mengidentifikasi kesalahan yang sering terjadi selama proses enhancement, sehingga membantu dalam meminimalkan bug dan meningkatkan performa aplikasi.
Zulianto et al. (2021)	<i>Katalon Studio</i>	Hasil penelitian menunjukkan bahwa penerapan pengujian black-box secara otomatis dengan <i>Katalon Studio</i> menghasilkan waktu eksekusi yang lebih cepat dibandingkan dengan pengujian manual. Untuk 13 kasus uji yang dicoba, waktu eksekusi pengujian otomatis adalah 283,08 detik, sedangkan pengujian manual memakan waktu 719,27 detik. Hal ini menunjukkan peningkatan kecepatan sebesar 2,54 kali lipat
Tempomona & Hendry (2022)	<i>Katalon Studio</i>	Hasil penelitian menunjukkan bahwa penggunaan <i>Katalon Studio</i> berhasil mempersingkat waktu pengujian berulang dengan jumlah data yang besar. Selain itu, alat ini mampu menghasilkan laporan pengujian yang lebih rinci secara otomatis, memudahkan identifikasi dan pemahaman terhadap kesalahan yang muncul.
Dewi, Rika L (2023)	<i>Robot Framework dan katalon Studio</i>	Hasil dari masing – masing pengujian Robot Framework dan <i>katalon Studio</i> akan dibandingkan dari sisi waktu. Penelitian ini memiliki hasil 54 berhasil 2 gagal pada fungsionalitas yang diuji.
Fadillah & Baidawi (2023)	<i>Katalon Studio</i>	Hasil penelitian menunjukkan bahwa penerapan <i>Katalon Studio</i> secara signifikan mengurangi waktu pengujian, meningkatkan jumlah bug yang terdeteksi, dan meningkatkan kepuasan penguji berkat kemudahan penggunaan serta fitur-fitur yang disediakan. Secara khusus, <i>Katalon Studio</i> mampu mengotomatisasi berbagai skenario pengujian API dengan efisien, memungkinkan deteksi dini terhadap potensi kesalahan dalam sistem informasi telekomunikasi.

Lanjutan Tabel 2.1. Perbandingan Tinjauan Pustaka

jamlean (2025)	<i>Katalon Studio</i>	Pada penelitian ini adalah melakukan pengujian terhadap website himasi dimana dalam pengujian menggunakan <i>katalon studio</i> yang diuji 2 halaman yaitu halaman admin dan halaman pengunjung dari masing-masing halaman tersebut diuji dan dianalisa dari hasil pengujian menggunakan <i>katalon studio</i>
----------------	-----------------------	--

2.2 Landasan Teori

Landasan adalah dasar pemikiran ilmiah yang digunakan untuk mendukung dan memperkuat penelitian atau kajian tertentu. Landasan ini berisi kumpulan teori, konsep, prinsip, serta hasil penelitian terdahulu yang relevan dengan topik yang dibahas. Tujuannya adalah untuk memberikan kerangka acuan yang jelas, membangun argumentasi yang logis, dan membantu peneliti dalam menganalisis data serta menarik kesimpulan yang tepat. Dengan adanya landasan teori, penelitian menjadi lebih terarah dan memiliki dasar yang kuat secara akademis.

2.2.1 Pengujian Perangkat Lunak

Pengujian perangkat lunak (*software testing*) merupakan proses evaluasi suatu sistem atau komponen perangkat lunak untuk menemukan kesalahan dan memastikan bahwa perangkat lunak berjalan sesuai dengan spesifikasi yang telah ditentukan. Pengujian ini bertujuan untuk meningkatkan kualitas perangkat lunak dengan mendeteksi dan memperbaiki bug sebelum sistem digunakan oleh pengguna akhir.

Quality Assurance atau biasa disebut *QA* adalah seseorang yang bertanggung jawab terhadap perencanaan jaminan kualitas, kesalahan, penyimpanan rekaman, analisis dan pelaporan.

Jaminan kualitas perangkat lunak adalah aktivitas pelindung yang diaplikasikan pada seluruh proses perangkat lunak. Proses ini dilaksanakan oleh seorang *QA Tester*. *QA Tester* memiliki tugas utama yaitu melakukan pengujian terhadap perangkat atau emulator, membuat alur pengujian, serta membuat laporan hasil pengujian yang terkadang bertugas untuk membuat program pengujian otomatis, membuat laporan pengujian, memberikan masukan atas aplikasi yang

diuji, serta berkomunikasi dengan pihak – pihak yang berkepentingan, seperti pengembang *UI/UX*, *Back-End* atau *Project Manager* (PM) (Zakiyah, 2018). Tujuan utama dari proses SQA adalah memastikan proses akhir dari sebuah produk berkualitas tinggi sesuai dengan persyaratan yang dinyatakan dalam standar yang telah ditetapkan. Sebuah perusahaan pengembang perangkat lunak mempunyai banyak proyek yang dikerjakan bersamaan, dan setiap pekerjaan memiliki rekrutment khusus dan tingkat kesulitan yang berbeda, oleh karena untuk perusahaan pengembang perangkat lunak harus memiliki standarisasi kustom terhadap perangkat lunak untuk menghasilkan hasil yang baik dan kualitas yang dapat diterima seperti yang diungkapkan (Zope et al (2015)).

Hariyanto, H., & Dirgahayu, T. (2020) menyatakan bahwa Kegagalan proyek pengembangan software dapat diakibatkan karena tidak adanya proses yang menjamin kualitas software sehingga dapat meminimalisir risiko – risiko yang ada pada setiap tahapan pengembangan *software*. Risiko ini bersifat teknis dan non teknis. Risiko teknis misalnya adalah *software* tidak dapat berfungsi sebagaimana mestinya, atau sulit untuk dioperasikan, dimodifikasi atau dikelola. Adapun risiko non teknis misalnya adalah proyek melebihi biaya yang dianggarkan, atau jadwal melewati batas yang direncanakan. Untuk mengurangi risiko – risiko tersebut diperlukan serangkaian aktivitas yang memastikan bahwa setiap proses dilakukan sesuai standar. *Software Quality Assurance (SQA)* merupakan serangkaian kegiatan yang terencana dan sistematis yang memastikan bahwa proses dan produk *software* sesuai dengan persyaratan, standar, dan prosedur. Pengembangan *software* yang benar menentukan kualitas sebuah *software* yang dikembangkan. Dengan adanya aktivitas *SQA* ini diharapkan kebutuhan dan target *software* yang dikembangkan dapat tercapai. Selain itu, proses pengembangan *software* juga dapat dibuktikan bahwa proses pengembangan telah dilakukan dengan baik karena *SQA* mendokumentasikan setiap prosesnya dengan standar.

2.2.2 Black Box Testing

Black box testing didefinisikan sebagai metode pengujian tanpa melihat struktur kode internal, rincian implementasi, dan pengetahuan tentang jalur internal

perangkat lunak. Jenis pengujian ini didasarkan sepenuhnya pada persyaratan dan spesifikasi perangkat lunak. Dalam *black box testing* hanya berfokus pada input dan output dari sistem perangkat lunak tanpa peduli tentang pengetahuan internal dari program perangkat lunak.

Menurut (Roni Setiawan, 2021) *Black box testing* atau dapat disebut juga *Behavioral Testing* adalah pengujian yang dilakukan untuk mengamati hasil input dan output dari perangkat lunak tanpa mengetahui struktur kode dari perangkat lunak.

Pengujian ini dilakukan di akhir pembuatan perangkat lunak untuk mengetahui apakah perangkat lunak dapat berfungsi dengan baik. Untuk melakukan pengujian, penguji tidak harus memiliki kemampuan menulis kode program. (Setiawan, 2021).

2.2.3 *Katalon Studio*

Katalon Studio adalah aplikasi open source untuk pengujian secara otomatis yang dikembangkan oleh *Katalon LLC*. *Katalon* mendukung 3 platform pengujian secara automasi, yaitu *Web testing*, *API testing*, dan *Mobile testing*. *Katalon* sudah terintegrasi dengan beberapa teknologi luar seperti *test*, *JIRA*, *kobiton*, *github*, dan lain – lain (*Katalon LLC* (2020)).

Kebutuhan proses pengujian dilakukan dengan menggunakan *tool* yaitu *Katalon Studio*. *Tools* tersebut merupakan aplikasi *open source* untuk melakukan pengujian secara otomatis seperti *Windows*, *MAC os* dan *linux*. *Katalon Studio* menggunakan antarmuka IDE khusus untuk melakukan pengujian. *Katalon Studio* memiliki tiga fitur utama untuk menunjang beberapa pengujian di beberapa platform seperti *Web testing*, *API testing*, dan *Mobile testing* (*Studio*, 2020).

Selain itu, penelitian oleh Kosasih (2021) menunjukkan bahwa *Katalon Studio* mendukung pengujian berbasis data (*data-driven testing*), yang memungkinkan penguji menggunakan berbagai format sumber data seperti CSV, Excel, dan database untuk mengotomatisasi skenario pengujian yang kompleks. Fitur ini mempermudah pengujian skala besar dan meningkatkan efisiensi dalam mendeteksi bug. Keunggulan lainnya adalah kemampuannya dalam mengelola

laporan hasil pengujian secara otomatis, sehingga mempermudah tim pengembang dalam menganalisis kinerja aplikasi yang diuji.

Menurut Sari et al. (2023), *Katalon Studio* juga memiliki keunggulan dalam mendukung pengujian aplikasi berbasis teknologi modern seperti *Angular*, *React*, dan *Vue.js*. Dengan antarmuka yang ramah pengguna dan fitur canggih seperti test object repository dan smart wait, *Katalon Studio* dapat meningkatkan efektivitas pengujian perangkat lunak. Dengan berbagai fitur yang ditawarkan, alat ini menjadi pilihan yang tepat untuk organisasi yang ingin meningkatkan kualitas perangkat lunak melalui pengujian otomatisasi yang lebih sistematis dan efisien.

2.2.4 Website Himpunan Mahasiswa Sistem Informasi (HIMASI)

Website Himpunan Mahasiswa Sistem Informasi (HIMASI) Universitas Teknologi Digital Indonesia Yogyakarta adalah sebuah platform digital yang dikembangkan untuk mendukung berbagai kegiatan akademik dan non-akademik mahasiswa. Website ini dibuat pada tahun 2022 dengan tujuan utama untuk menyediakan informasi terkait organisasi HIMASI, termasuk program kerja, berita terbaru, serta pengumuman penting bagi mahasiswa Sistem Informasi. Dengan adanya website ini, mahasiswa dapat lebih mudah mengakses informasi dan berpartisipasi dalam kegiatan yang diselenggarakan oleh HIMASI. Selain sebagai media informasi, website HIMASI juga memiliki fitur interaktif seperti penyampaian aspirasi mahasiswa dan *open recruitment*. Fitur penyampaian aspirasi memungkinkan mahasiswa untuk menyampaikan saran, kritik, atau keluhan terkait kehidupan kampus secara langsung kepada HIMASI. Sementara itu, fitur *open recruitment* digunakan untuk menjaring anggota baru HIMASI melalui proses pendaftaran dan seleksi yang dilakukan secara online. Dengan fitur-fitur ini, website HIMASI berperan sebagai jembatan komunikasi antara mahasiswa dan organisasi. Agar dapat memberikan pengalaman pengguna yang optimal, website HIMASI perlu diuji keandalannya melalui metode pengujian perangkat lunak seperti *Black Box Testing* menggunakan *Katalon Studio*. Pengujian ini bertujuan untuk memastikan bahwa setiap fitur pada website berjalan sesuai dengan spesifikasi yang telah dirancang

dan bebas dari kesalahan yang dapat mengganggu fungsionalitasnya. Dengan adanya pengujian yang tepat, website HIMASI diharapkan dapat menjadi platform yang andal dan efektif dalam mendukung aktivitas mahasiswa Sistem Informasi. Berikut adalah tampilan website dari halaman pengunjung dan halaman admin.

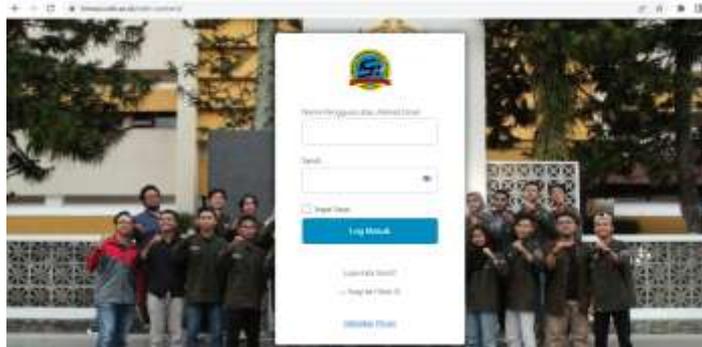
Website HIMASI juga terus dikembangkan agar selaras dengan perkembangan teknologi dan kebutuhan mahasiswa. Desain antarmuka yang responsif memudahkan akses dari berbagai perangkat, baik desktop maupun mobile. Selain itu, integrasi dengan media sosial dan kalender kegiatan mempermudah penyebaran informasi serta pengingat agenda penting bagi anggota HIMASI. Dengan upaya pengembangan yang berkelanjutan, website ini diharapkan mampu menjadi sarana utama dalam membangun komunitas mahasiswa Sistem Informasi yang aktif, kolaboratif, dan informatif.



Gambar 2.1 Tampilan Website Pengunjung

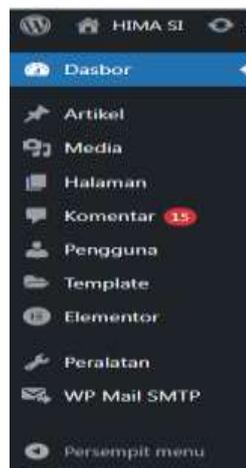
Pada gambar 2.1 merupakan tampilan dari halaman website pengunjung web HIMASI, yang di halaman web tersebut terdapat menu beranda, tentang kami, aspirasi mahasiswa, *open recruitment*, galeri dan kontak kami. Dalam menu ini yang akan diuji adalah aspirasi mahasiswa dan *open recruitment*, dalam pengujiannya dengan cara menginputkan data pada halaman aspirasi mahasiswa dan *open recruitment* dengan melihat *response* pada halaman tersebut setelah melakukan klik simpan action nya seperti apa dan akan dibahas pada skenario pengujian. Untuk fitur yang ada di halaman pengunjung (mahasiswa) terdiri dari :

1. Menu *open recruitment* fitur ini berfungsi sebagai pendaftaran calon anggota baru HIMASI.
2. Menu aspirasi fitur ini berfungsi untuk mahasiswa yang mempunyai aspirasi guna untuk memberikan feedback / informasi bagi HIMASI.



Gambar 2.2 Tampilan Form Login Admin

Pada gambar 2.2 merupakan tampilan form login admin HIMASI, pada halaman ini nantinya juga akan diuji menggunakan *black box testing* menggunakan *katalon studio* untuk pengujiannya dengan menginputkan data username dan password salah dan benar *response* nya seperti apa setelah melakukan klik tombol login yang akan di bahas di skenario pengujian.



Gambar 2.3 Tampilan Daftar Menu Admin Himasi

Pada gambar 2.3 merupakan tampilan daftar menu dashboard admin HIMASI, untuk yang diuji adalah menu artikel, media, halaman, komentar, pengguna, template, elementor, peralatan dan wp mail smtp, pada menu tersebut adalah yang akan di test tambah, update, delete, lihat semua data dan lihat detail, dari masing masing pengujian tersebut setelah melakukan action pada tombol responsenya seperti apa, akan di bahas lebih lengkapnya di skenario pengujiannya. Untuk fitur yang ada di admin terdiri dari yang digunakan untuk pengujian :

1. Menu artikel fitur ini berfungsi sebagai menampilkan artikel di halaman

pengunjung (mahasiswa) fitur ini terdiri dari tambah data, lihat data, hapus data, edit data dan searching data.

2. Menu media fitur ini berfungsi sebagai menampilkan media informasi di halaman pengunjung (mahasiswa) fitur ini terdiri dari tambah data, lihat data, hapus data, edit data dan searching data.

3. Menu halaman fitur ini berfungsi sebagai menampilkan laman menu informasi di halaman pengunjung (mahasiswa) fitur ini terdiri dari tambah data, lihat data, hapus data, edit data dan searching data.

4. Menu komentar fitur ini berfungsi sebagai menampilkan hasil komentar dari mahasiswa dari halaman pengunjung (mahasiswa) fitur ini terdiri dari lihat data, hapus data, edit data dan searching data.

5. Menu komentar fitur ini berfungsi sebagai menambahkan pengguna / pada halaman admin fitur ini terdiri dari tambah data, lihat data, hapus data, edit data dan searching data. Sedangkan fitur admin yang tidak masuk dalam pengujian :

1. Menu template fitur ini tidak masuk dalam pengujian karena di dalam fitur ini tidak ada berkaitan dengan data seperti tambah data, hapus data, lihat data, detail data maupun searching data melainkan untuk tampilan dari wordpress. Fungsi dari fitur ini digunakan sebagai pengganti template wordpress pada website.

2. Menu elementor fitur ini tidak masuk dalam pengujian karena di dalam fitur ini tidak ada berkaitan dengan data seperti tambah data, hapus data, lihat data, detail data maupun searching data melainkan untuk tampilan dari wordpress. Fungsi dari fitur ini digunakan untuk install plugin tampilan pada wordpress.

3. Menu peralatan : not found ketika di klik

4. Menu email fitur ini tidak masuk dalam pengujian karena di dalam fitur ini tidak ada berkaitan dengan data seperti tambah data, hapus data, lihat data, detail data maupun searching data melainkan untuk setting dari wordpress. Fungsi pada fitur ini digunakan sebagai setting email pada wordpress.

2.2.5 Groovy

Groovy adalah bahasa pemrograman dinamis yang berjalan di atas Java Virtual Machine (JVM), dan memiliki sintaks yang ringkas serta fleksibel. Groovy

dapat berinteraksi langsung dengan kode Java dan sangat cocok digunakan untuk scripting, termasuk pengujian otomatis (automation testing) seperti pada Katalon Studio (https://en.wikipedia.org/wiki/Apache_Groovy).