

PENGUJIAN WEBSITE MENGGUNAKAN SELENIUM IDE PADA JDIH PESAWARAN MENGGUNAKAN METODE *EQUIVALENCE PARTITIONING*

Fathoni, Muhammad Haidar Afif Mufid, Muhammad Aqil Zidane,
Shopi Attika Putri, Rintan Wirnanti, Ali Ibrahim

Sistem Informasi, Universitas Sriwijaya
Jl. Raya Palembang - Prabumulih No.KM. 32, Ogan Ilir, Indonesia
fathoni@unsri.ac.id

ABSTRAK

Perkembangan teknologi mendorong instansi pemerintah untuk meningkatkan kualitas *website* sebagai wadah informasi publik, salah satunya adalah JDIH Pesawaran yang menyediakan dokumen hukum terorganisir. Namun, peluang *bug* pada fitur-fitur *website* dapat mengurangi kemampuan sistem, terutama dalam melakukan validasi masukan dari suatu data. Penelitian ini bertujuan untuk menguji fungsionalitas *website* JDIH Pesawaran menggunakan Selenium IDE dan metode *Equivalence Partitioning* untuk menguji dan menemukan celah sistem. Metode *Black Box Testing* diterapkan dalam klasifikasi *input valid* dan *invalid* dari empat fitur utama: login, instansi, data admin, dan jabatan. Hasil pengujian 19 *test case* menunjukkan 78,95% sesuai harapan dan 21,05% mengalami ketidaksesuaian terutama pada validasi panjang karakter di kolom instansi dan jabatan yang melebihi 200 karakter. Hal ini menunjukkan perlunya perbaikan batasan *input* dan optimasi tampilan. Penggunaan Selenium IDE terbukti mempercepat proses pengujian secara otomatis dan minim akan kesalahan. Penelitian ini menyarankan untuk mengombinasikan metode pengujian dan *tools automation* lainnya untuk meningkatkan akurasi serta pusat pada pengembangan validasi data dan keamanan sistem di penelitian selanjutnya.

Kata kunci : Pengujian, *Black Box*, *Equivalence Partitioning*, Selenium, JDIH

1. PENDAHULUAN

Dengan meningkat pesatnya perkembangan teknologi di era globalisasi, penggunaan *website* telah menjadi salah satu cara utama untuk menyebarkan informasi dan membantu kegiatan pada suatu organisasi terutama pemerintah. Semakin tinggi kualitas *website* pemerintah maka akan semakin tinggi juga kredibilitasnya [1].

Undang-Undang Nomor 14 Tahun 2008 tentang Keterbukaan Informasi Publik menyatakan bahwa penyebaran setiap informasi hukum adalah kategori informasi publik yang diwajibkan ketersediaannya untuk publik [2].

JDIH (Jaringan Dokumentasi dan Informasi Hukum) adalah salah satu sistem yang ada di pemerintahan, berbentuk sebuah wadah dokumentasi hukum dan merupakan sarana pemberian pelayanan informasi hukum yang dapat diandalkan dan berlandaskan pada Peraturan Presiden No 33 Tahun 2012 [3].

Dalam rangka meningkatkan kualitas pembangunan hukum nasional dan pelayanan kepada publik, Badan Pembinaan Hukum Nasional (BPHN) membangun JDIH sebagai pusat Jaringan dan diberi tugas sebagai penyelenggara latihan pembinaan tenaga, tempat konsultasi, penelitian dan pengembangan sistem jaringan, serta koordinator kegiatan unit-unit jaringan dalam rangka pengembangan jaringan. JDIH juga tersebar ke beberapa daerah di Indonesia termasuk di daerah Pesawaran Kota Lampung.

JDIH (Jaringan Dokumentasi dan Informasi Hukum) Pesawaran merupakan platform digital yang

menyediakan akses terpusat dan terstruktur terhadap berbagai produk hukum daerah Kabupaten Pesawaran. *Website* ini berperan sebagai pusat penyimpanan, pengelolaan, dan diseminasi regulasi daerah, seperti peraturan daerah, keputusan bupati, dan dokumen hukum terkait. Masyarakat umum, akademisi, ataupun instansi pemerintah juga dapat memanfaatkan platform ini untuk mengunduh atau mempelajari kebijakan hukum terbaru secara praktis dan efisien. JDIH Pesawaran berfungsi untuk mengorganisir dokumen hukum dalam bentuk digital.

Pengujian sistem dapat diartikan sebagai sebuah cara dalam memastikan bahwa suatu perangkat lunak sudah memiliki kualitas yang baik serta terhindar dari kecacatan sistem yang mempengaruhi fungsionalitasnya. Penelitian yang telah dilakukan oleh peneliti terdahulu menunjukkan bahwa pengujian *website* menggunakan Selenium IDE dengan metode *Equivalence Partitioning* mampu mencapai akurasi 92% [4].

Dengan demikian, metode tersebut menjadi lebih handal untuk digunakan dalam pengujian sistem. Di dalam sebuah *website*, dalam kasus ini *website* JDIH, terdapat banyak sekali data maupun informasi hukum yang tidak hanya penting namun juga sensitif terhadap kebocoran. Dengan keberadaan informasi penting pada *website* JDIH Pesawaran mengenai hukum, tentunya sebuah *website* tidak bisa lepas dari kesalahan. Untuk meminimalisir kesalahan baik yang kecil maupun besar, maka perlu dilakukan sebuah pengujian khusus terhadap sistem yang ada [10].

2. TINJAUAN PUSTAKA

2.1. Black Box Testing

Dalam melakukan pengujian suatu sistem, terdapat sebuah teknik pengujian suatu perangkat lunak yang berfokus pada spesifikasi fungsional tanpa melihat kode sumber dinamakan *black box testing*. Dalam teknik ini akan ada evaluasi dari input dan output sistem untuk memastikan bahwa perangkat lunak berfungsi sesuai dengan spesifikasi sesuai rancangannya [11].

Dalam melakukan *black box testing*, terdapat 2 cara untuk menjalankan teknik ini, yaitu manual dan otomatis yang dapat bertumpu pada antarmuka aplikasi [20].

Lebih lanjut, *black box testing* unggul dalam mendeteksi kesalahan fungsionalitas aplikasi dan terbatas dalam mengkonfirmasi kesalahan pada tingkat kode jika tidak digabungkan dengan metode pengujian lainnya [19].

2.2. Selenium IDE

Pengujian otomatis dapat memastikan bahwa pengujian tersebut berjalan dengan cepat dan tepat, salah satu software yang digunakan dalam pengujian otomatis adalah Selenium IDE. Alat pengujian otomatis berbasis *browser* yang mendukung pengujian fungsional lintas-*browser*, Selenium IDE membantu proses pengujian yang cepat dengan tetap menjaga akurasi hasil dan dapat mendeteksi kesalahan pada tahap awal pengembangan [7].

Dengan menggunakan metode ini, diharapkan dapat mengidentifikasi kelemahan di suatu situs *web* dan juga dapat memperbaiki situs *web* tersebut guna meningkatkan kualitas serta pengalaman pengguna [5].

2.3. Equivalence Partitioning

Metode pengujian yang digunakan yaitu menggunakan teknik *equivalence partitioning* dengan memastikan sistem menghasilkan output yang sesuai karena dapat mengidentifikasi kelas input [17].

Teknik ini membantu terkait rancangan skenario pengujian untuk penilaian kualitas suatu sistem atau aplikasi dan memudahkan pengujian berdasarkan data yang terperinci sesuai *use case* dan kriteria yang ditentukan [12].

2.4. Penelitian Terdahulu

Penelitian sebelumnya hanya berfokus pada pengujian komponen dan fungsionalitas dasar saja tanpa mempertimbangkan pengelompokan input yang dimana dapat meningkatkan efisiensi pengujian. Oleh karena itu, penelitian ini akan berfokus pada bagaimana penerapan pengujian *website* JDIH Pesawaran dengan metode *equivalence partitioning* menggunakan Selenium IDE. Penelitian ini bertujuan untuk mengidentifikasi dan mengelompokkan input-input yang ada sehingga peneliti dapat merancang skenario pengujian yang lebih efektif dan efisien. Peneliti juga dapat menemukan kesalahan yang

kemungkinan terjadi keterlewatan dalam pengujian secara manual dan dapat memberikan kontribusi terhadap pengembangan metode pengujian secara otomatis menggunakan Selenium IDE yang lebih baik kedepannya [6].

Pada penelitian yang dilakukan Mulyati dkk, penelitian ini melakukan uji menggunakan Selenium IDE dengan metode *black box* pada aplikasi Pawoon. Studi ini menekankan betapa pentingnya pengujian dilakukan untuk perbaikan dan pengecekan secara berkala. Penelitian ini mendapatkan hasil yang baik dimana pada semua *test case* yang dilakukan berhasil melewati pengujian dengan aman [9].

Dalam Penelitian lain, pengujian otomatis menggunakan Selenium untuk mengevaluasi kinerja *website* BPJS Kesehatan yang fokus terhadap waktu respons sehingga mendapatkan hasil pengujian yang cukup memuaskan yaitu rata-rata keseluruhan waktu respons sekitar 1.34 detik. Penelitian ini masih diperlukan perbaikan untuk meningkatkan konsistensi dan responsivitas sistem dalam keadaan beban yang tinggi. Peneliti melibatkan analisis statistik deskriptif untuk mengevaluasi performa sesuai dengan fungsionalitas dan pengalaman pengguna yang lebih baik [8].

Hasil dari penelitian *black-box* yang dilakukan oleh [10] dengan fokus masalah pada fitur form login dan form submit file dengan menggunakan metode *decision table* menunjukkan bahwa kedua fitur berfungsi dengan baik dan bebas dari *bug*. Dengan menggunakan metode *decision table*, evaluasi dapat dijalankan secara sistematis terhadap berbagai kondisi input yang mungkin terjadi. Hal tersebut juga sejalan dengan metode *equivalence partitioning* yang diintegrasikan untuk meningkatkan efisiensi pengujian dengan mengelompokkan input yang dianggap setara. Selanjutnya, penelitian ini juga merekomendasikan untuk dilakukan pengembangan dengan gabungan metode *black-box testing* untuk fitur lainnya.

Sementara itu, penelitian yang menggunakan metode *decision table* untuk melakukan pengujian *blackbox* pada sebuah *website*, mendapatkan hasil yang kurang baik. *Website* yang diuji mendapatkan hasil yang kurang dalam menangani berbagai skenario input terutama dalam validasi data dan interaksi pengguna. Kelebihan dari metode ini adalah kemampuannya dalam menangani berbagai kombinasi kondisi dengan pendekatan yang sistematis dan baik, namun terbatas pada proses penyusunan tabel yang kompleks dan butuh waktu yang relatif lama [14].

Penelitian sebelumnya dilakukannya pengujian *black box* terhadap *website* Sistem Informasi Perpustakaan Berbasis *Web* menggunakan teknik *state transition* dan metode *waterfall* yang dimana pengujian dilakukan dengan mengamati perubahan status dari satu tahap ke tahap berikutnya dalam sistem, tanpa perlu memahami kode sumbernya. Penelitian ini masih memiliki sebuah masalah yaitu masih adanya jalur eksekusi dalam sistem yang belum diuji secara menyeluruh sehingga menyebabkan

kesalahan yang tidak terdeteksi sehingga hasil dari penelitian ini yaitu pada teknik *state transition testing* mampu mengidentifikasi beberapa kendala dalam perpindahan status dalam sistem, seperti kesalahan akses pada beberapa menu bagi pengguna dengan hak akses terbatas dalam *website* tersebut. Metode *waterfall* dapat menguji berbagai alur sistem secara sistematis, terutama dalam aspek validasi transisi antar menu. Namun, teknik ini hanya berfokus pada pengujian alur perpindahan status dan tidak mencakup aspek lain seperti keamanan atau performa sistem. Penelitian ini dapat dikembangkan lebih lanjut dengan mengkombinasikan teknik *state transition testing* dengan alat otomatisasi seperti Selenium IDE untuk meningkatkan efisiensi pengujian, serta melihat aspek validasi *input* dan keamanan data guna meningkatkan kualitas keseluruhan sistem perpustakaan berbasis *web* ini [13].

Lalu ada penelitian yang dilakukan pengujian Sistem Informasi Perpustakaan SMA Wijaya Putra Surabaya apakah berjalan sesuai dengan fungsionalitas yang diharapkan dan untuk mendeteksi adanya kesalahan fungsi di awal sehingga sistem dapat segera diperbaiki agar beroperasi lebih optimal. Dalam pengujian tersebut, peneliti menerapkan metode *black box testing* dengan fokus pada dua teknik khusus, yaitu *equivalence partitioning* dan *boundary value analysis*. Hasil pengujiannya yaitu terdapat ketidaksesuaian pada beberapa skenario, terutama pada fitur komentar yang dimana mengindikasikan kurangnya validasi *input data* dan perhatian terhadap *sensitive case*. Meskipun begitu, hasil utama dari penelitian ini menunjukkan bahwa sistem memperoleh nilai evaluasi sebesar 84,65% yang termasuk dalam kategori sangat efektif, sehingga pengembang disarankan untuk tetap melakukan perbaikan, terutama dengan penambahan validasi *input data* dan penanganan *sensitive case* yang dimana untuk meningkatkan keoptimalan kinerja sistem [18].

3. METODE PENELITIAN

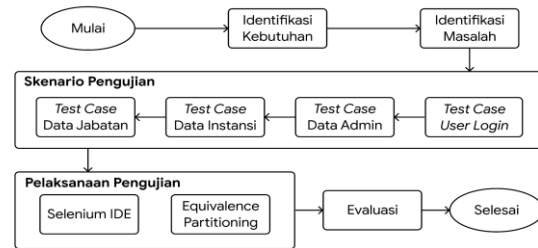
3.1. Identifikasi Kebutuhan

Dalam proses pengujian sistem, terdapat beberapa tahapan yang harus dilakukan supaya memberikan hasil yang maksimal, hal pertama yang harus dilakukan yaitu dengan mengidentifikasi segala kebutuhan dan tujuan utama dari pembuatan *website*. Suatu pengujian terhadap *website* dilakukan ketika suatu *website* tersebut masih tergolong baru dan masih banyak kesalahan atau *bug* di dalamnya. Kemudian setelah mengetahui kebutuhan utama dan tujuan yang dibutuhkan dari aplikasi yang telah dibuat, maka kedua hal tersebut dapat dijadikan sebuah acuan untuk pengujian ketika melakukan pengujian aplikasi [15].

Alur pengujian *website* JDIH Pesawaran menggunakan metode *Equivalence Partitioning* dapat terlihat pada gambar 2.



Gambar 1. Website JDIH Pesawaran



Gambar 2. Tahapan penelitian

3.2. Identifikasi Masalah

Setelah identifikasi kebutuhan dipahami, tahapan yang perlu dilakukan selanjutnya yaitu mengidentifikasi masalah. Pada tahap ini akan ditemukan kendala atau permasalahan di dalam *website* tersebut yang berkaitan dengan fungsionalitas suatu *website*. Tahapan ini dilakukan untuk memastikan bahwa *website* dapat berjalan secara optimal dan sesuai dengan kebutuhan sistem. Apabila ditemukan permasalahan dalam proses pengujian, maka diperlukan solusi untuk analisis lebih lanjut sebelum *website* dinyatakan siap digunakan. Namun, apabila tidak ditemukan permasalahan sistem maka *website* tersebut telah berfungsi sesuai dengan tujuan dan dapat dinyatakan siap digunakan [4].

3.3. Skenario Pengujian

Skenario pengujian merupakan sebuah kasus uji yang memiliki peluang tinggi atau memiliki kemungkinan untuk menemukan kesalahan yang belum pernah ditemukan sebelumnya. Pada tahap ini, Skenario pengujian ini memiliki manfaat bagi pengujian agar selama proses pengujian dapat dilakukan secara tepat dan cepat [4].

Tahapan skenario pengujian pada pengujian sistem juga bertujuan untuk memastikan apakah tiap proses sudah berfungsi sesuai dengan kebutuhan yang diharapkan [16].

3.4. Pelaksanaan Pengujian

Pada tahapan ini setelah dilakukannya skenario pengujian, proses penting dalam pengujian dilakukan sesuai dengan yang telah dirancang sebelumnya. *Black box testing* digunakan untuk menguji fungsionalitas sistem tanpa melihat kode sumbernya dengan teknik *equivalence partitioning* yang cara kerjanya membagi data uji ke beberapa kelas input yang valid ataupun invalid, upaya memastikan sistem dapat menangani

setiap kemungkinan yang terjadi dari masukan yang ada.

Alat yang digunakan dalam pelaksanaan pengujian sistem ini berupa Selenium IDE yang berbasis *record and playback*. Berbagai *input* dari pengguna akan direkam berdasarkan partisi kelas yang telah ditentukan sebelumnya. Jika sistem merespon dengan benar terhadap *valid input* dan menolak *invalid input* sesuai dengan yang telah dirancang, maka *website* tersebut sudah berfungsi dengan semestinya. Sebaliknya, jika ditemukan suatu kendala atau permasalahan terkait respon sistem, maka diperlukan perbaikan agar *website* dapat berjalan secara optimal.

3.5. Evaluasi

Tahapan terakhir dari suatu proses pengujian *website* yaitu dilakukannya evaluasi. Maksud dari evaluasi yaitu apabila penguji pada penelitian ini telah melakukan tahapan-tahapan pengujian *website* sebelumnya seperti tahapan identifikasi kebutuhan, identifikasi masalah, skenario pengujian, dan melaksanakan pengujian. Pada tahapan ini, penguji sudah melakukan proses pengujian dan mendapatkan hasil dari proses pengujian suatu *website*.

Dalam proses evaluasi pengujian, langkah yang dilakukan yaitu dengan menyimpulkan hasil pengujian dari pelaksanaan pengujian yang menggunakan metode *equivalence partitioning* dan memakai Selenium IDE. Kemudian dilakukannya evaluasi dari celah kesalahan pengujian *website*, yang meliputi keakuratan hasil pengujian, kinerja sistem dalam menangani input valid dan invalid, dan kemampuan Selenium IDE dalam menangani input valid dan invalid, serta dapat mendeteksi dan menolak kesalahan, yang dimana pengujian ini bertujuan untuk mengidentifikasi kelemahan sistem dan juga sebagai perbaikan untuk pengembang perangkat lunak agar dapat memperbaiki sistem kedepannya [16].

Proses pengujian pada JDIH Pesawaran dilakukan terhadap 4 fitur utama yaitu *form login*, data admin, data instansi, dan data jabatan. Hasil dari pengujian tersebut apabila ditemukan celah kesalahan maka akan dilakukan evaluasi pada *test case* yang tidak sesuai.

4. HASIL DAN PEMBAHASAN

4.1. Identifikasi Kebutuhan

Pengujian yang digunakan dalam penelitian ini dari *website* JDIH Pesawaran yang diperoleh dari platform Google. Pengujian ini menggunakan pendekatan *black box* dengan menggunakan teknik *equivalence partitioning* yang dimana untuk mengetahui informasi mengenai kebutuhan utama dan tujuan dari *website* yang diuji, serta dapat mengetahui mengenai sisi fungsionalitas dari *website* sehingga penguji dapat mengetahui apakah *website* sudah berjalan sebagaimana mestinya seperti yang diharapkan oleh pihak terkait [4].

4.2. Identifikasi Masalah

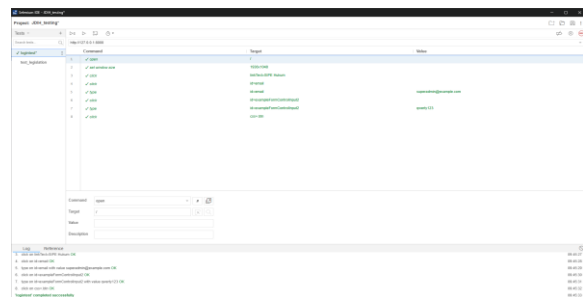
Pada penelitian ini, *website* JDIH Pesawaran merupakan sebuah *website* baru yang terdapat dokumentasi-dokumentasi hukum. Hal ini dapat menjadi permasalahan besar ketika *website* mengalami serangan *cyber* yang mengancam keamanan data hukum. Selain itu terdapat permasalahan lainnya seperti banyaknya *bug* dan *error* pada *website* yang dapat berdampak pada penurunan kinerja sistem, ketidakakuratan data, dan mengganggu dalam penyajian informasi hukum kepada masyarakat.

4.3. Skenario Pengujian

Tahapan dari skenario pengujian yaitu dengan mengidentifikasi fitur-fitur yang ada di dalam sebuah *website* tersebut. Terdapat empat skenario pengujian yang dilakukan dalam penelitian ini, yaitu skenario pengujian pada *user login*, data admin, data instansi, dan data jabatan. Kemudian dilakukan penomoran di setiap skenario pengujian. Setelah itu dilakukan keempat skenario pengujian tersebut, maka dilakukannya harapan hasil dari pengujian tersebut apakah berhasil atau tidak dari pengujian menggunakan Selenium IDE tersebut. Skenario pengujian ini dibuat dari sisi admin agar dapat memastikan fungsionalitasnya apakah telah cukup baik dan dapat digunakan oleh pengguna.

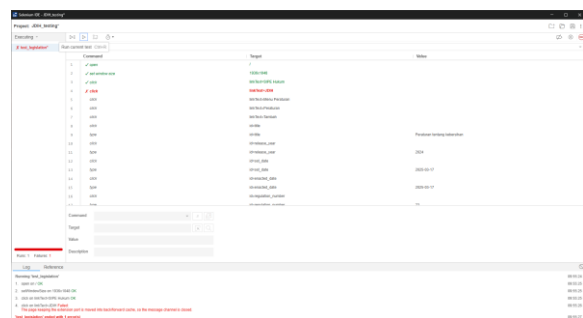
4.4. Pelaksanaan Pengujian

Pada gambar 3 dan 4 menunjukkan tampilan ketika pengujian menggunakan Selenium IDE berhasil dan gagal pada suatu *test case*.



Gambar 3. Tampilan selenium IDE sukses

Gambar 3 merupakan tahapan pengujian berhasil yang terdiri dari serangkaian proses perekaman *input* dan output dari suatu *case* pengujian.



Gambar 4. Tampilan selenium IDE gagal

Kemudian pada Gambar 4, menunjukkan langkah-langkah pengujian dengan kegagalan suatu *case*, sehingga dapat memberikan gambaran tentang

titik-titik kelemahan pada *case* pengujian. Hasil pengujian terhadap *test case* yang telah ditentukan dapat terlihat seperti pada tabel berikut.

Tabel 1. Hasil pengujian *user login*

ID	Skenario Pengujian	Harapan Hasil	Hasil Pengujian	Kesimpulan
TL-01	Mengisi Username dan Password dengan benar	User akan berhasil login	User berhasil login	Sesuai
TL-02	Mengisi Username dan Password salah	User tidak berhasil login	User tidak berhasil login	Sesuai
TL-03	Mengisi Username tanpa '@gmail.com' salah dan password benar	User tidak berhasil login	User tidak berhasil login	Sesuai
TL-04	Mengisi Username dengan '@gmail.com' benar dan password salah	User tidak berhasil login	User tidak berhasil login	Sesuai

Tabel 1 menjelaskan hasil pengujian menggunakan Selenium IDE dari 4 *test case* yang ditentukan. Hasil dari *test case* ini mengimplementasikan metode *equivalence partitioning* yang membagi input menjadi kategori

valid dan invalid. Kesimpulan yang didapatkan dari keempat *test case* mempunyai kesimpulan yang sama dan tidak ditemukan anomali pada saat pengujian, sehingga skenario pengujian adalah 100%.

Tabel 2. Hasil pengujian data admin

ID	Skenario Pengujian	Harapan Hasil	Hasil Pengujian	Kesimpulan
TA-01	Menambahkan data admin dengan benar	Data admin akan berhasil tersimpan di dalam	Data admin akan berhasil tersimpan di dalam	Sesuai
TA-02	Mengisi data <i>email</i> tanpa "@gmail.com"	Sistem akan menolak menyimpan email dan menampilkan "email harus memiliki @"	Sistem menolak menyimpan email dan menampilkan "email harus memiliki @"	Sesuai
TA-03	Mengisi data Nomor HP dibawah 10 digit	Sistem akan menolak menyimpan Nomor HP dan menampilkan "Nomor HP harus berisi 10-15 digit"	Sistem menolak menyimpan email dan menampilkan "Nomor HP harus berisi 10-15 digit"	Sesuai
TA-04	Mengisi data Nomor HP diatas 15 digit	Sistem akan menolak menyimpan Nomor HP dan menampilkan "Nomor HP harus berisi 10-15 digit"	Sistem menolak menyimpan email dan menampilkan "Nomor HP harus berisi 10-15 digit"	Sesuai
TA-05	Mengisi data password dibawah 5 digit	Sistem akan menolak menyimpan Password dan menampilkan "Password harus berisi minimal 5 digit"	Sistem menolak menyimpan Password dan menampilkan "Password harus berisi minimal 5 digit"	Sesuai

Tabel 2 menjelaskan hasil pengujian dalam menambahkan data admin dari 5 *test case*. Dari kelima *test case* yang sudah dilakukan tersebut menghasilkan

kesimpulan bahwa 100% skenario sukses karena semua masukan yang dijalankan menghasilkan keluaran yang sesuai dengan sebagaimana fungsi fitur.

Tabel 3. Hasil pengujian data instansi

ID	Skenario Pengujian	Harapan Hasil	Hasil Pengujian	Kesimpulan
TI-01	Kolom Tambah Instansi tidak berisi	Sistem akan menampilkan pesan "Nama Instansi Wajib Diisi"	Sistem menampilkan pesan "Nama Instansi Wajib Diisi"	Sesuai
TI-02	Kolom Instansi diisi lebih dari 200 karakter	Sistem akan menampilkan pesan "Nama Instansi tidak melebihi 200 karakter"	Kolom instansi terisi	Tidak Sesuai
TI-03	Kolom Edit Instansi tidak berisi	Sistem akan menampilkan pesan "Nama Instansi Wajib Diisi"	Sistem menampilkan pesan "Nama Instansi Wajib Diisi"	Sesuai
TI-04	Kolom Edit Instansi diisi lebih dari 200 karakter	Sistem akan menampilkan pesan "Nama Instansi tidak melebihi 200 karakter"	Kolom instansi terisi	Tidak Sesuai
TI-05	Kolom Tambah Instansi tidak berisi	Sistem akan menampilkan pesan "Nama Instansi Wajib Diisi"	Sistem menampilkan pesan "Nama Instansi Wajib Diisi"	Sesuai

Pada tabel 3 menunjukkan penggunaan metode *equivalence partitioning* pada 5 *test case* di fitur kelola data Instansi. Dari 5 *test case* tersebut dapat

disimpulkan bahwa fitur memiliki *bug* yang karena ada 2 hasil pengujian yang tidak sesuai yang menyebabkan rusaknya tampilan pada *website*.

Tabel 4. Hasil pengujian data jabatan

ID	Skenario Pengujian	Harapan Hasil	Hasil Pengujian	Kesimpulan
TJ-01	Kolom Tambah Jabatan tidak berisi	Sistem akan menampilkan pesan "Nama Jabatan Wajib Diisi"	Sistem menampilkan pesan "Nama Jabatan Wajib Diisi"	Sesuai
TJ-02	Kolom Jabatan diisi lebih dari 200 karakter	Sistem akan menampilkan pesan "Nama Jabatan tidak melebihi 200 karakter"	Kolom Jabatan terisi	Tidak Sesuai
TJ-03	Kolom Edit Jabatan tidak berisi	Sistem akan menampilkan pesan "Nama Jabatan Wajib Diisi"	Sistem menampilkan pesan "Nama Jabatan Wajib Diisi"	Sesuai
TJ-04	Kolom Edit Jabatan diisi lebih dari 200 karakter	Sistem akan menampilkan pesan "Nama Jabatan tidak melebihi 200 karakter"	Kolom Jabatan terisi	Tidak Sesuai
TJ-05	Kolom Nama Instansi tidak terisi	Sistem akan menampilkan pesan "Nama Instansi Wajib Diisi"	Sistem akan menampilkan pesan "Nama Instansi Wajib Diisi"	Sesuai

Pada tabel 4, mendeskripsikan hasil pengujian data jabatan menggunakan metode *equivalence partitioning* dengan total 5 *test case*. Dari keenam *test case* yang telah dilakukan pengujian menghasilkan kesimpulan 3 *test case* yang sesuai dan 2 *test case* yang tidak sesuai. Hal ini mengindikasikan bahwa masih terdapat *bug* dalam pengisian karakter di kolom tambah dan edit jabatan.

4.5. Evaluasi

Tabel 5 menunjukkan hasil *test case* yang dilakukan pada *website* JDIH Pesawaran dengan menggunakan Selenium IDE.

Tabel 5. Keseluruhan hasil pengujian

Test Case	Hasil
TL-01	Sesuai
TL-02	Sesuai
TL-03	Sesuai
TL-04	Sesuai
TA-01	Sesuai
TA-02	Sesuai
TA-03	Sesuai
TA-04	Sesuai
TA-05	Sesuai
TI-01	Sesuai
TI-02	Tidak Sesuai
TI-03	Sesuai
TI-04	Tidak Sesuai
TI-05	Sesuai
TJ-01	Sesuai
TJ-02	Tidak Sesuai
TJ-03	Sesuai
TJ-04	Tidak Sesuai
TJ-05	Sesuai

Dari 19 *test case* yang telah dilakukan pengujian menggunakan metode *Equivalence Partitioning*, mempunyai kesimpulan bahwa terdapat 15 *test case* yang sesuai dan 4 *test case* yang tidak sesuai. Adapun deskripsi terhadap celah kesalahan *website* terdapat pada tabel 6 berikut.

Tabel 6. Celah kesalahan *website*

No	Celah Kesalahan	Deskripsi
1	Pada TI-02, TJ-02, TI-04, dan TJ-04 isi dari kolom jabatan dan instansi tidak dibatasi	Menyebabkan sebuah kesalahan tampilan yang muncul melebihi yang seharusnya

Pada Tabel 6 menunjukkan kesalahan yang ada pada *website* yang diuji. Berdasarkan hal tersebut diperlukan solusi, yaitu membatasi karakter yang bisa dimasukkan ke dalam kolom jabatan dan instansi sebanyak 200 karakter, atau membuat batas pada tabel sehingga tabel tidak melewati batas *div* nya dan juga meminimalisir terjadinya *error* pada tampilan websitenya. Kemudian, hasil pengujian pada *website* JDIH Pesawaran menunjukkan bahwa terdapat 4 *test case* dari total 19 *test case* yang tidak sesuai, ini menghasilkan nilai akurasi sebesar 78,95% yang dapat digolongkan sebagai baik dalam suatu pengujian suatu sistem.

5. KESIMPULAN DAN SARAN

Dari penerapan pengujian metode *black box* dengan teknik *equivalence partitioning* pada *website* JDIH Pesawaran menunjukkan hasil sebanyak 78,95%. *test case* sesuai dan bekerja sebagaimana fungsinya, dan sebanyak 21,05% *test case* masih menunjukkan hasil yang tidak sesuai. Hal tersebut menunjukkan bahwa masih terdapat beberapa fitur yang butuh untuk dilakukan perbaikan agar dapat beroperasi dengan maksimal pada saat penggunaan *website* oleh *user*. Selenium IDE sebagai alat pengujian otomatis yang digunakan membantu proses pengujian secara cepat dan tepat dalam mendeteksi *bug* fitur. Penelitian selanjutnya disarankan untuk menggunakan metode lain seperti *boundary value analysis* atau *all-pairs testing* dan *fuzzing testing* yang dimana metode ini dapat mengkombinasikan pengujian input atau kondisi yang tidak biasa yang tidak terdeteksi oleh metode *equivalence partitioning*, dan juga dapat menggunakan tools automation lainnya seperti Cypress atau Playwright sebagai variabel perbandingan agar dapat membuat hasil pengujian pada suatu *website* semakin akurat.

DAFTAR PUSTAKA

- [1] R. R. E. Nainggolan, "Analisis Penggunaan Website dan Media Sosial Pemerintah untuk Pelayanan Publik," *Jurnal Teknologi dan Komunikasi Pemerintahan*, vol. 6, no. 1, pp. 1–21, 2024, doi: 10.33701/jtkp.v6i1.4221.
- [2] P. Indonesia, *Undang-Undang No 14 Tahun 2008 Tentang Keterbukaan*. 2008.

- [3] P. Indonesia, *Peraturan Presiden No 33 Tahun 2012 Tentang Jaringan Dokumentasi Dan Informasi Hukum Nasional*. 2012.
- [4] E. Hardianto, F. P. Aditiawan, and A. N. Sihananto, "Pemanfaatan Selenium Ide Dalam Otomatisasi Pengujian Black Box Pada Website Umkm Kota Madiun Dengan Teknik Equivalence Partitioning," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 3, pp. 3471–3477, 2024, doi: 10.36040/jati.v8i3.9729.
- [5] I. Kusyadi, A. Firmansyah, C. Irawan, F. A. Aziz, R. F. Sidhiq, and M. Munawaroh, "Pengujian aplikasi website sekolah menggunakan selenium webdriver," *Jurnal Teknologi Sistem Informasi dan Aplikasi*, vol. 6, no. 2, pp. 180–185, 2023, doi: 10.32493/jtsi.v6i2.26634.
- [6] A. P. Ginting, Z. Abidin, A. Asari, and A. Saifudin, "Otomatisasi Pengujian Aplikasi Web Toko Sembako Menggunakan Selenium IDE," *LOGIC : Jurnal Ilmu Komputer dan Pendidikan*, vol. 1, no. 2, pp. 303–309, 2023.
- [7] S. K. Akbar, "Analisis Pengujian Fungsionalitas Otomatis dengan Selenium IDE dan Usability Testing pada Web Portal Berita Radio Republik Indonesia (RRI) dengan Metode Cognitive Walkthrough untuk Meningkatkan Pengalaman Pengguna," *Jurnal Sistem Informasi*, vol. 8, no. 2, 2024.
- [8] M. Fikri, A. Muh. I. H. B. Malolo, M. K. MS, and A. Resky, "Analisis Pengujian Fungsionalitas website BPJS: Eksplorasi Performa dengan Selenium Melalui Pendekatan Statistik Deskriptif," *Konferensi Nasional Ilmu Komputer (KONIK)*, vol. 6, no. 1, 2023.
- [9] S. Mulyati, A. Kurniawan, H. Ramadhan, H. Saputra, S. Siregar, and N. Ratama, "Otomatisasi Pengujian Aplikasi Pawoon Menggunakan Selenium IDE," *Jurnal Teknologi Sistem Informasi dan Aplikasi*, vol. 6, no. 2, pp. 168–172, 2023, doi: 10.32493/jtsi.v6i2.26312.
- [10] G. I. Marthasari, A. T. Wahyuningsih, M. R. Aviansyah, M. A. Ramadhani, and Z. Rahmatullah, "Pengujian Website Infotech Menggunakan Teknik Black Box Decision Table," *Jurnal Informatika Universitas Pamulang*, vol. 7, no. 1, 2022, doi: 10.32493/informatika.v7i1.17315.
- [11] Fahrullah, "Implementasi Pengujian Black Box Pada Sistem Informasi Monitoring Akademik Dengan Pendekatan Teknik Equivalence Partitions," *Jurnal Teknosains Kodepena*, vol. 1, no. 2, 2021.
- [12] M. R. Maulana, "Pengujian Black Box dengan Teknik Equivalence Partitioning pada Aplikasi Monitoring Pemberian Obat Filariasis Berbasis Android," *KLIK: Kajian Ilmiah Informatika dan Komputer*, vol. 4, no. 4, 2024.
- [13] M. Mintarsih, "Pengujian Black Box Dengan Teknik Transition Pada Sistem Informasi Perpustakaan Berbasis Web Dengan Metode Waterfall Pada SMC Foundation," *Jurnal Teknologi Dan Sistem Informasi Bisnis*, vol. 5, no. 1, pp. 33–35, 2023, doi: 10.47233/jteksis.v5i1.727.
- [14] S. W. Nurjihan *et al.*, "Analisis Decision Table Testing Untuk Pengujian Blackbox Website Pusat Studi Bencana IPB," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 11, no. 4, 2024, doi: 10.35957/jatisi.v11i4.9618.
- [15] A. Samdono, A. Puspita Sari, and F. Prima Aditiawan, "Pengujian Black Box Pada Sistem Informasi Stok dan Penjualan Berbasis Website Menggunakan Metode Equivalence Partitioning," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 1, pp. 880–885, Mar. 2024, doi: 10.36040/jati.v8i1.8893.
- [16] D. B. Muslimin, D. Kusmanto, K. F. Amilia, M. S. Ariffin, S. Mardiana, and Y. Yulianti, "Pengujian Black Box pada Aplikasi Sistem Informasi Akademik Menggunakan Teknik Equivalence Partitioning," *Jurnal Informatika Universitas Pamulang*, vol. 5, no. 1, p. 19, Mar. 2020, doi: 10.32493/informatika.v5i1.3778.
- [17] Y. Yulianti, A. A. Khaidar, R. Fazriansyah, S. G. Ramadhan, W. A. Putra, and S. L. M. Sitio, "Pengujian Black Box pada Website MyUnpam menggunakan Teknik Equivalence Partitioning," *Jurnal Teknologi Sistem Informasi dan Aplikasi*, vol. 6, no. 2, pp. 154–161, Apr. 2023, doi: 10.32493/jtsi.v6i2.25501.
- [18] A. S. Aqmarina, F. P. Aditiawan, and H. E. Wahanani, "Pengujian Sistem Informasi Perpustakaan SMA Wijaya Putra Surabaya Menggunakan Metode Black Box Testing dengan Teknik Equivalence Partitioning dan Boundary Value Analysis," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 1, pp. 855–860, Mar. 2024, doi: 10.36040/jati.v8i1.8898.
- [19] M. A. M. A. Rachman *et al.*, "Penerapan Black Box Testing Untuk Evaluasi Fungsionalitas Website Maggoplast," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 9, no. 1, pp. 169–176, Dec. 2024, doi: 10.36040/jati.v9i1.12177.
- [20] B. A. Priyaungga, D. B. Aji, M. Syahrone, N. T. S. Aji, and A. Saifudin, "Pengujian Black Box pada Aplikasi Perpustakaan Menggunakan Teknik Equivalence Partitions," *Jurnal Teknologi Sistem Informasi dan Aplikasi*, vol. 3, no. 3, p. 150, Aug. 2020, doi: 10.32493/jtsi.v3i3.5343.