

## Pengujian *Black Box* pada Aplikasi Ecampus Menggunakan Metode *Equivalence Partitioning*

Rully Pramudita <sup>1,\*</sup>

<sup>1</sup> Manajemen Informatika; Universitas Bina Insani; Jl. Siliwangi No. 6 Rawa Panjang Bekasi Timur 17114, Telp. (021) 824 36 886 / (021) 824 36 996. Fax. (021) 824 009 24; e-mail: [rullypramudita@binainsani.ac.id](mailto:rullypramudita@binainsani.ac.id)

\* Korespondensi: e-mail: [rullypramudita@binainsani.ac.id](mailto:rullypramudita@binainsani.ac.id)

Diterima: 12 Mei 2020; Review: 15 Mei 2020; Disetujui: 30 Mei 2020

Cara sitasi: Pramudita, P. 2020. Pengujian Black Box pada Aplikasi Ecampus Menggunakan Metode *Equivalence Partitioning*. *Informatics for Educators and Professionals*. Vol 4 (2): 193-202.

**Abstrak** : Salah satu tahapan penting dalam pengembangan sebuah perangkat lunak yaitu tahap pengujian. Teknik testing dibagi menjadi dua yaitu *black box testing* dan *white box testing*. Pada penelitian akan menerapkan teknik pengujian *black box*. Teknik pengujian *black box* terdiri dari, *Equivalence Partitioning*, *Boundary Value Analysis / Limit Testing*, *Comparison Testing*, *Sample Testing*, *Robustness Testing*, *Behavior Testing*, *Requirement Testing*, *Performance Testing*, *Endurance Testing*, *Cause-Effect Relationship Testing*. Salah satu yang akan diterapkan dalam penelitian ini yaitu teknik *equivalence partitioning*. Teknik ini ujicoba yang dilakukan berdasarkan tujuh tahapan, yaitu menentukan use case yang diuji, tentukan kriteria, definisikan partisi, buat data uji, buat kasus uji, lakukan pengujian dan evaluasi. Hasil pengujian nantinya akan menghasilkan *feedback* untuk masukan bagi pengelola ecampus.

**Kata kunci**: Aplikasi, *Black box*, Ecampus, *Ekuivalence partitioning*, Pengujian

**Abstract**: One important stage in the development of a software is the testing phase. Testing techniques are divided into two namely *black box testing* and *white box testing*. In the study will apply *black box testing* techniques. *Black box testing* techniques consist of, *Equivalence Partitioning*, *Boundary Value Analysis / Limit Testing*, *Comparison Testing*, *Sample Testing*, *Robustness Testing*, *Behavior Testing*, *Requirement Testing*, *Performance Testing*, *Endurance Testing*, *Cause-Effect Relationship Testing*. One that will be applied in this research is *equivalence partitioning* technique. This technique is based on seven stages of testing, namely determining the tested use case, determining criteria, defining partitions, creating test data, making test cases, testing and evaluating. The test results will later produce *feedback* for input for ecampus managers.

**Keywords**: Application, *Black box*, Ecampus, *Ekuivalence partitioning*, Testing

### 1. Pendahuluan

Aplikasi ecampus merupakan aplikasi yang diterapkan dalam pengelolaan akademik dan keuangan perguruan tinggi. Dengan adanya aplikasi ini diharapkan proses bisnis perguruan tinggi dapat dikelola dengan lebih baik. Regulasi pemerintah yang semakin ketat dan dinamis mengakibatkan perguruan tinggi harus selalu dengan cepat beradaptasi dengan regulasi yang diterapkan. Sehingga dengan adanya aplikasi seperti ini perguruan tinggi setidaknya dapat terbantu dalam mengelola dan melaporkan kinerja kepada para *stakeholders*.

Sistem Informasi dalam sebuah organisasi memiliki peranan yang cukup penting dalam memberikan informasi kepada manajemen dalam rangka mengendalikan berbagai proses yang kompleks dalam sebuah organisasi [1]. Aplikasi ini memiliki beragam fitur dan level akses yang disesuaikan dengan kewenangan dalam struktur organisasi. Dalam proses bisnis utama sebuah perguruan tinggi yaitu pelaksanaan tridharma. Salah satunya yaitu pelaksanaan pendidikan atau pengajaran. Beberapa fitur utamanya yaitu pelaksanaan belajar mengajar yang dilakukan oleh

dosen bersama mahasiswa. Dari mulai dosen memulai pembelajaran sampai dosen membuat nilai akhir sebuah matakuliah.

Namun sebelum itu perlu dilakukannya sebuah pengujian aplikasi dengan tujuan memberikan feedback mengenai kelebihan maupun adanya kekurangan yang masih ada pada aplikasi tersebut. Dalam penelitian-penelitian terdahulu yang telah melakukan pengujian dikatakan bahwa pengujian merupakan sebuah tahapan yang dilakukan oleh tester untuk menemukan kesalahan yang ada di suatu perangkat lunak [2]. Pengujian perangkat lunak merupakan elemen kritis dalam menentukan kualitas suatu perangkat lunak [3]. Pengujian fungsional dilakukan untuk menemukan apakah ada kesalahan-kesalahan pada aplikasi yang telah dibangun [4].

Tahap pengujian adalah salah satu tahapan pada siklus pengembangan sebuah perangkat lunak. Tahap ini merupakan rangkaian kegiatan yang dilakukan untuk menemukan kesalahan atau kekurangan dari sebuah perangkat lunak atau aplikasi komputer [5]. Aryandana bahkan menyebutkan bahwa *software Testing is helpful to alleviate the developers' task in improving the system* [6]. Dapat disimpulkan bahwa kegiatan pengujian ini merupakan tahapan yang sangat penting dalam proses pengembangan perangkat lunak. Metoda Black Box Testing terdiri atas beberapa cara antara lain Equivalence Partitioning, Boundary Value Analysis, Comparison Testing, Sample Testing, Robustness Tesing, dan lain-lain [7]. Metode BlackboxTesting merupakan salah satu metode yang mudah digunakan karena hanya memerlukan batas bawah dan batas atas dari data yang di harapkan,Estimasi banyaknya data uji dapat dihitung melalui banyaknya field data entri yang akan diuji, aturan entri yang harus dipenuhi serta kasus batas atas dan batas bawah yang memenuhi [8]. *Black box testing is a system or application testing that focuses on the functional specifications of the system or application. System testers can define a set of input conditions and test the functional specifications of the system* [9].

Teknik pengujian black box yang digunakan dalam penelitian ini adalah Teknik Equivalence Partitioning, teknik ini memiliki 7 tahapan diantaranya yaitu menentukan use case yang diuji, tentukan kriteria, definisikan partisi, buat data uji, buat kasus uji, lakukan pengujian dan evaluasi [10]. Menurut Krismadi dalam penelitiannya mengatakan saat melakukan pengujian terhadap aplikasi promosi jabatan mengaktakan bahwa teknik *equivalence partitioning* ialah menguji kualitas aplikasi yang akan dilakukan dokumentasi pengujian perangkat lunak dengan ditemukannya kesalahan pada setiap form yang dibagi menjadi tiga model kesalahan, yaitu kesalahan pada fungsi, struktur data dan antarmuka [11].

Sedangkan dari penelitian yang dilakukan oleh Ningrum menyimpulkan bahwa pengujian dengan metode Black Box berbasis Equivalence Partitions dapat membantu proses pembuatan case pengujian, uji kualitas dan menemukan kesalahan yang tidak terdeteksi yang disebabkan oleh kesalahan pengetikan [12]. Adapun yang dikemukakan Hidayat bahwa pengujian *Equivalence Partitioning* merupakan pengujian berdasarkan inputan setiap menu yang terdapat pada sistem [13]. Metode Black Box berbasis *Equivalensi partitions* menguji tingkat akurasi yang akan dilakukan dokumentasi pengujian perangkat lunak dengan ditemukannya kesalahan pada setiap form yang dibagi menjadi lima model kesalahan, diantaranya : kesalahan pada Fungsi, Struktur data, Interface, Inisialisasi, dan Performance [14]. Dalam penelitian yang dilakukan oleh Aryandana bahwa teknik equivalence partitioning itu memiliki kelebihan yaitu dapat menentukan *test case* yang lebih spesifik dan dapat mengenali cacat sistem berdasarkan kasus uji yang sedang diuji [6]. Dalam teknik *Equivalence Class Partitioning* (ECP) perlu menguji hanya satu syarat dari setiap subset atau kelas yang dipartisi [15]. Teknik ini membantu kita mengurangi jumlah kasus uji. Ini pada dasarnya berfungsi membagi domain input program berdasarkan nilai input ke dalam kelas kesetaraan. Uji kasus dihasilkan dari kelas ekivalensi ini yang diturunkan dari input domain [16]. *Equivalence partitioning strives to define a test case that uncover classes of errors and thereby reduces the number of test cases needed. It is based on an evaluation of equivalence classes for an input condition* [17].

## 2. Metode Penelitian

Pada tahapan ini akan dijelaskan mengenai metode penelitian yang digunakan sehingga terlihat jelas langkah-langkah yang dilakukan dalam penelitian ini, khususnya dalam melakukan pengujian mengguakan teknik equivalence partitioning. Pada gambar 1 dijelaskan langkah-langkah yang akan dilakukan dalam melakukan pengujian dengan teknik *equivalence partitioning*. Terdapat tujuh tahapan menggunakan teknik ini [10], yaitu pertama melakukan penentuan use case yang akan diujikan. Kedua menentukan kriteria apa saja yang akan diujikan berdasarkan

pilihan use case ditahap pertama. Ketiga mendefinisikan partisi dari masing-masing kriteria yang telah ditentukan. Keempat membuat data uji berdasarkan partisi yang telah dibuat dilangkah ketiga. Kelima yaitu membuat kasus uji berdasarkan data uji yang telah dirancang pada tahap keempat. Keenam yaitu melakukan pengujian kepada perangkat lunak ecampus berdasarkan daftar kasus uji yang telah dibuat pada langkah kelima. Tahap terakhir yaitu langkah ketujuh melakukan evaluasi dari hasil pengujian yang telah dilakukan, tahap ini akan menghasilkan sebuah umpan balik bagi pengelola apakah masih terdapat kekurangan atau memang perangkat lunak sudah tidak ada lagi cacat.



Sumber : Ikhlashi (2019)

Gambar 1. Tahapan Pengujian *Equivalence Partitioning*

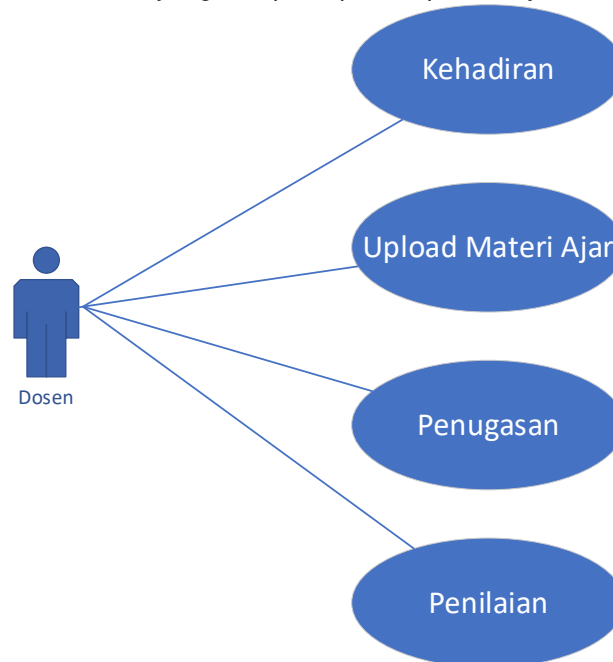
Berdasarkan gambar 1, tahap pertama yaitu melakukan penentuan *use case* yang akan diujikan. *Use case* ini dipilih dari keseluruhan *use case* yang ada. Setelah ditentukan *use case* maka dilanjutkan ke tahap kedua, yaitu menentukan kriteria berdasarkan *use case* yang telah ditentukan pada tahap pertama. Setelah kriteria ditentukan maka dapat melanjutkan ke tahap mendefinisikan partisi, yaitu per kriteria akan dipecah menjadi beberapa partisi sesuai dengan karakteristik kriteria tersebut. Tahap keempat setelah melakukan pendefinisian partisi maka selanjutnya membuat data uji berbentuk tabel, tabel ini berisikan daftar data uji yang akan menjadi bahan penentuan kasus uji. Tahap kelima yaitu membuat kasus uji berdasarkan daftar data uji yang telah dibuat sebelumnya. Kasus uji dapat dibuat dalam bentuk tabel agar memudahkan pengujian dalam melakukan pengujian di tahap selanjutnya. Tahap keenam merupakan tahap pengujian langsung pada aplikasi menggunakan tabel kasus uji yang telah dibuat ditahap lima. Nanti pengujian dapat mengisi hasil uji pada tabel tersebut. Setelah melakukan pengujian, tahap akhir yaitu mengevaluasi hasil pengujian, pada tahap ini dapat dipaparkan apa saja temuan saat pengujian dan menjelaskan kekurangan serta kelebihan dari aplikasi serta dapat menambahkan rekomendasi agar aplikasi menjadi jauh lebih baik lagi.

### 3. Hasil dan Pembahasan

Bagian hasil dan pembahasan berisi mengenai pembahasan hasil pelaksanaan penelitian berdasarkan metodologi penelitian yang digunakan.

### Penentuan *Use Case*

Tahapan pertama yang harus dilakukan sebelum melakukan pengujian dengan teknik *equivalence partitioning* yaitu melakukan penentuan *use case* yang akan diujikan. Dalam konteks dosen sebagai pengguna aplikasi ecampus, memiliki *use case* yang beragam diantaranya penilaian, kehadiran, *upload* materi ajar, dan penugasan. Dari beberapa *use case* yang ada, selanjutnya akan ditentukan *use case* yang akan dijadikan acuan pengujian yaitu penilaian. Pada gambar 2 tergambar *use case* yang ada pada proses pembelajaran.



Sumber : Hasil Penelitian (2020)

Gambar 2. *Use Case* Proses Pembelajaran

### Penentuan Kriteria

Setelah ditentukan *use case*, maka selanjutnya yaitu menentukan kriteria, tabel 1 menjelaskan detail apa saja kriteria yang akan diujikan. Kriteria ini diambil dari beberapa fitur yang terdapat pada proses penilaian.

Tabel 1. Kriteria Uji

Kode	Kriteria
T01	Mengisi nilai pada kolom nilai
T02	Mengunci nilai
T03	Cetak nilai

Sumber : Hasil Penelitian (2020)

Berdasarkan tabel 1, terdapat tiga kriteria yang akan dijadikan dasar pengujian, yaitu mengisi nilai pada kolom nilai, mengunci nilai dan mencetak nilai. Dari tiga kriteria tersebut selanjutnya akan dibuat partisi untuk masing-masing kriteria input nilai, kunci nilai dan cetak nilai dengan nilai invalid (iv) dan valid (v). Pembuatan partisi akan dilakukan pada tahap ketiga.

### Pendefinisian Partisi

Kriteria pertama memiliki partisi berbentuk range nilai integer atau bilangan bulat positif yang diberi rentang nilai 0-100, diluar itu dapat dikatakan data invalid. Kriteria kedua dan ketiga memiliki partisi berbentuk boolean yakni hanya ada dua opsi antara *true* dan *false*.

Pada tahap ini akan dibuat tujuh partisi seperti yang ada pada gambar 2. Pada kriteria input nilai terdapat tiga partisi yaitu T011, T012, dan T013, dimana kondisi valid terdapat pada rentang nilai 0-100 dan kondisi invalid yaitu nilai kurang dari nol serta nilai lebih dari seratus. Pada kriteria kunci nilai terdapat dua partisi yaitu T021 dan T022, kondisi valid terdapat pada kondisi nilai terkunci sedangkan kondisi invalid terjadi saat nilai tidak terkunci. Kriteria ketiga memiliki dua partisi yaitu T031 dan T032, dimana kondisi valid berada pada kondisi nilai dapat tercetak sedangkan kondisi invalid terjadi saat nilai tidak dapat tercetak.



Sumber : Hasil Penelitian (2020)

Gambar 3. Partisi Uji

### Data Uji

Tahapan data uji merupakan langkah lanjutan setelah sebelumnya telah dibuatkan partisi uji sebanyak tujuh partisi. Di tahap ini akan dibuatkan data uji dalam bentuk tabel yang menjelaskan apa saja data yang akan diujikan. Berikut pada tabel 2 penjelasan detailnya.

Tabel 2. Data Uji

No Data Uji	Field	Data Uji	Kode Partisi	Tipe Uji
D001	Input Nilai	Nilai < 0	T011	iv
D002	Input Nilai	$0 \leq \text{Nilai} \leq 100$	T012	v
D003	Input Nilai	Nilai > 100	T013	iv
D004	Kunci Nilai	Dikunci	T021	v
D005	Kunci Nilai	Tidak Dikunci	T022	iv
D006	Cetak Nilai	Dicetak	T031	v
D007	Cetak Nilai	Tidak Dicetak	T032	iv

Sumber : Hasil Penelitian (2020)

Berdasarkan tabel 2, dapat dilihat bentukan data uji yang dihasilkan sebanyak D001, D002, D003, D004, D005, D006 dan D007.

### Kasus Uji

Tahapan kasus uji merupakan langkah kelima dari teknik *equivalence partitioning*. Pada tahap ini akan ditentukan kasus uji, yaitu kasus yang akan diujikan berdasarkan data uji sebelumnya. Nantinya kasus uji ini akan menjadi acuan saat melakukan tahap pengujian langsung terhadap aplikasi ecampus. Tabel 3 menjelaskan rancangan kasus uji yang akan digunakan dalam pengujian.

Tabel 3. Kasus Uji

No	Data Uji	Tipe Uji	Output	Hasil
TC10	Input Nilai (D001)	iv		
TC11	Input Nilai (D002)	v		
TC12	Input Nilai (D003)	iv		
TC13	Kunci Nilai (D004)	v		
TC14	Kunci Nilai (D005)	iv		
TC15	Cetak Nilai (D006)	v		
TC16	Cetak Nilai (D007)	iv		

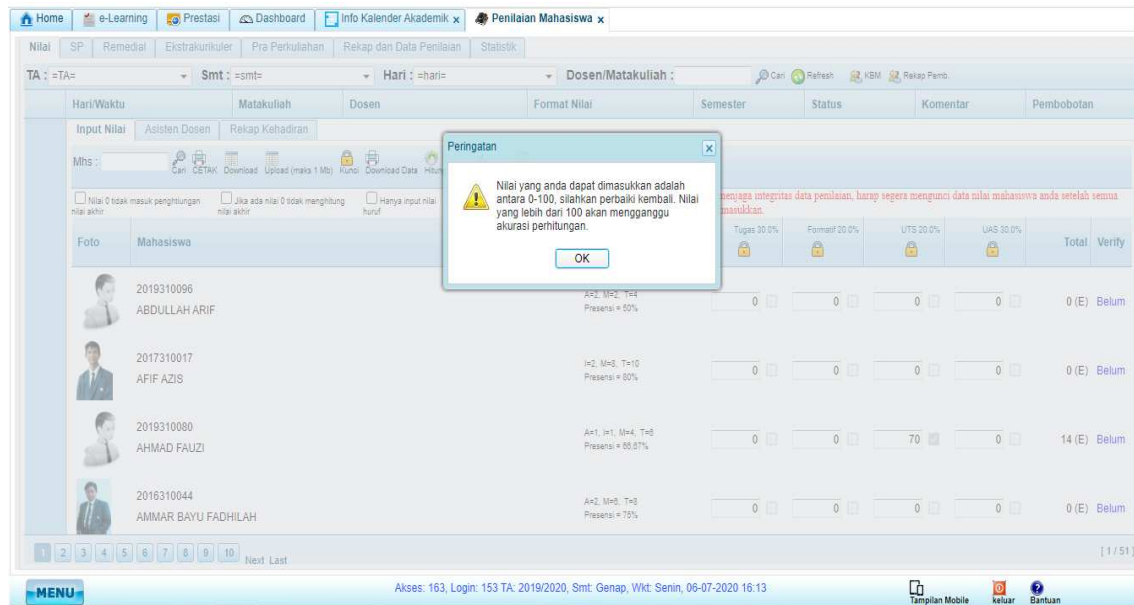
Sumber : Hasil Penelitian (2020)

Berdasarkan tabel 3, dapat dilihat bahwa terdapat tujuh kasus uji yang akan menjadi bahan pengujian langsung, yaitu TC10, TC11, TC12, TC13, TC14, TC15 dan TC16. Selanjutnya tabel kasus uji ini akan digunakan saat melakukan pengujian.

## Pengujian

Tahapan ini dilakukan berdasarkan rancangan kasus uji yang telah ditentukan pada langkah sebelumnya. Terdapat tujuh kasus uji yang akan digunakan sebagai acuan kasus apa saja yang akan diujikan secara langsung pada aplikasi ecampus. Pertama pengujian input nilai yang terbagi menjadi tiga kasus uji yaitu nilai kurang dari nol, nilai antara nol dan seratus, serta nilai lebih dari seratus.

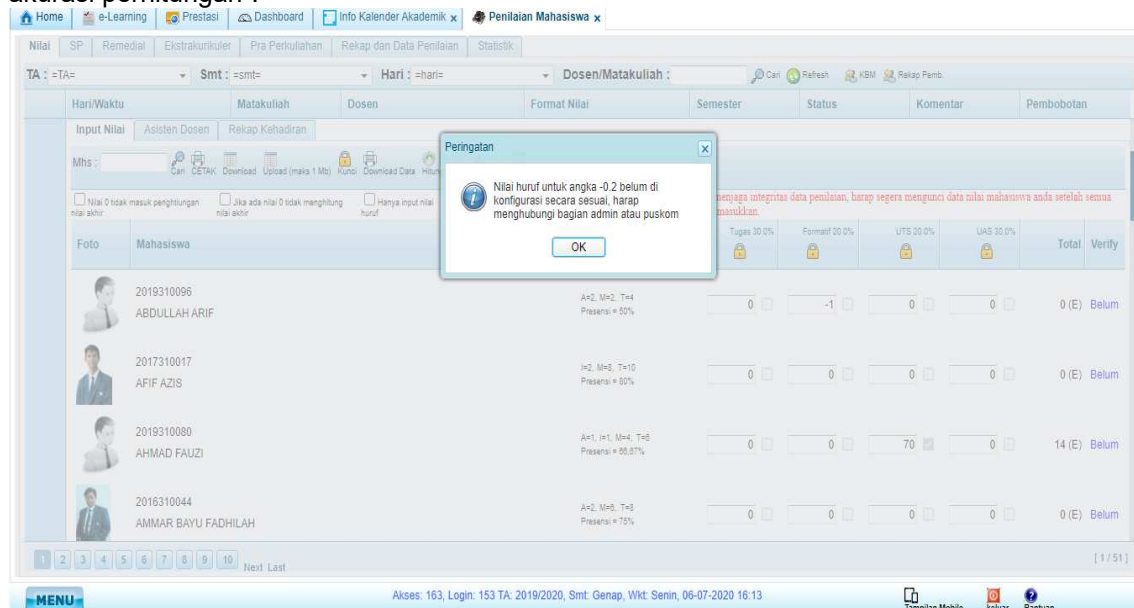
Kedua pengujian saat mengunci nilai, terdapat dua kasus uji yaitu pengguna mengunci nilai dan tidak mengunci nilai. Ketiga pengujian saat mencetak nilai, terdapat dua kasus uji yaitu saat klik tombol mencetak nilai.



Sumber : Hasil Penelitian (2020)

Gambar 4. Uji Input Nilai > 100

Pada gambar 3 terdapat hasil keluaran aplikasi ketika melakukan uji input nilai, pengujian ini dilakukan dengan melakukan input nilai menggunakan angka lebih dari 100. Setelah dimasukan ternyata muncul pesan peringatan berupa kalimat "Nilai yang anda dapat dimasukkan adalah antara 0-100, silahkan perbaiki kembali. Nilai yang lebih dari 100 akan mengganggu akurasi perhitungan".

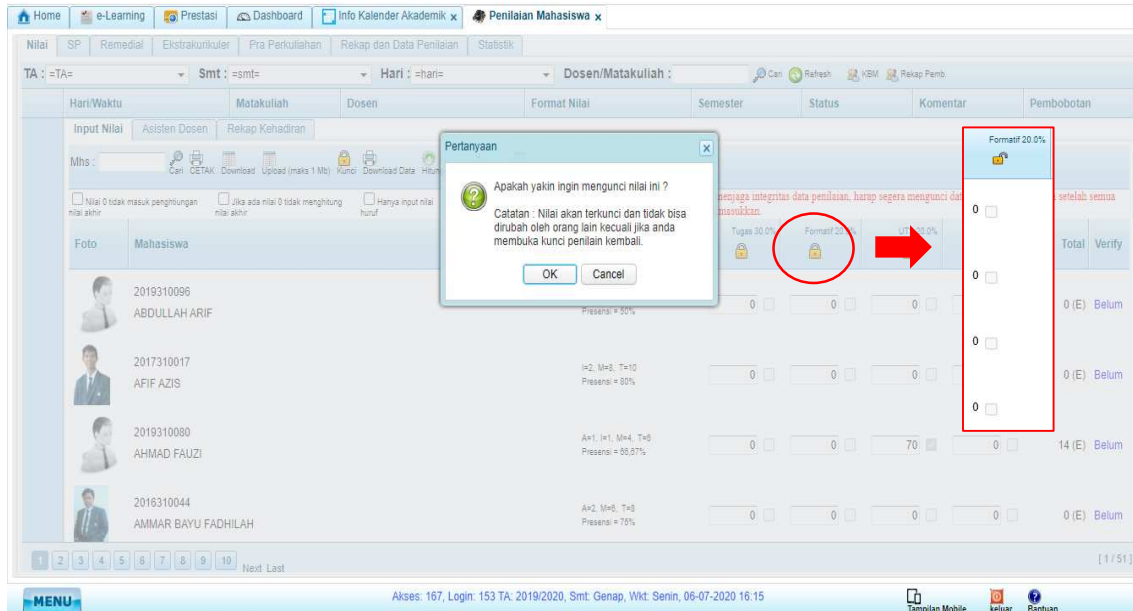


Sumber : Hasil Penelitian (2020)

Gambar 5. Uji Input Nilai < 0



Pada gambar 4 terdapat hasil keluaran aplikasi ketika melakukan uji input nilai, pengujian ini dilakukan dengan melakukan input nilai menggunakan angka kurang dari nol. Setelah dimasukan ternyata muncul pesan peringatan berupa kalimat “Nilai huruf untuk angka -0 2 belum di konfigurasi secara sesuai, harap menghubungi bagian admin atau puskom”.



Sumber : Hasil Penelitian (2020)

Gambar 6. Uji Kunci Nilai

Pada gambar 5 terdapat hasil keluaran aplikasi ketika melakukan uji kunci nilai, pengujian ini dilakukan dengan melakukan kunci nilai menggunakan tombol kunci. Setelah tombol kunci diklik, maka muncul pesan pertanyaan berupa kalimat “Apakah yakin ingin mengunci nilai ini? Catatan : Nilai akan terkunci dan tidak bisa dirubah oleh orang lain kecuali jika anda membuka kunci penilaian kembali”. Dan setelah dilakukan klik tombol kunci nilai terbukti nilai yang sudah dimasukan sudah terkunci dan tidak dapat diubah, kecuali pengguna dalam hal ini dosen melakukan pembukaan kunci untuk melakukan perubahan nilai. Hal ini sudah sesuai dengan kebutuhan awal sistem yang membutuhkan keamanan dalam hal privasi nilai, agar tidak sembarang pengguna dapat mengubah nilai yang sudah dimasukan oleh dosen yang berwenang atas matakuliah tersebut.

Tabel 4. Pengujian

No	Data Uji	Tipe Uji	Output	Hasil
TC10	Input Nilai (D001)	iv	Nilai huruf untuk angka -0 2 belum di konfigurasi secara sesuai, harap menghubungi bagian admin atau puskom	Berhasil
TC11	Input Nilai (D002)	v	Tidak ada pesan	Berhasil
TC12	Input Nilai (D003)	iv	Nilai terinput dalam form nilai	Berhasil
TC13	Kunci Nilai (D004)	v	Nilai yang anda dapat dimasukkan adalah antara 0-100, silahkan perbaiki kembali. Nilai yang lebih dari 100 akan mengganggu akurasi perhitungan	Berhasil
TC13	Kunci Nilai (D004)	v	Apakah yakin ingin mengunci nilai ini? Catatan : Nilai akan	Berhasil

No	Data Uji	Tipe Uji	Output	Hasil
			terkunci dan tidak bisa dirubah oleh orang lain kecuali jika anda membuka kunci penilaian kembali	
			Nilai terkunci dan tidak dapat ubah nilai	
TC14	Kunci Nilai (D005)	iv	Nilai tidak terkunci dan masih dapat diubah	Berhasil
TC15	Cetak Nilai (D006)	v	Tercetak dalam bentuk pdf	Berhasil
TC16	Cetak Nilai (D007)	iv	Tidak tercetak	Berhasil

Sumber : Hasil Penelitian (2020)

### Evaluasi Hasil Pengujian

Setelah melakukan pengujian, maka langkah terakhir adalah dilakukan evaluasi hasil pengujian. Berdasarkan hasil pengujian menghasilkan hasil yang positif, itu dapat diartikan bahwa pengujian yang dilakukan berhasil berdasarkan kasus uji yang ditentukan. Namun ada beberapa hal yang perlu diperbaiki dari segi keluaran yang dihasilkan oleh aplikasi.

Tabel 5. Evaluasi Hasil

No	No Uji	Output Hasil Uji	Rekomendasi
EV01	TC10	Nilai huruf untuk angka -0 2 belum di konfigurasi secara sesuai, harap menghubungi bagian admin atau puskom	<b>Perbaikan pesan :</b> Mohon maaf nilai yang anda masukan belum sesuai, silahkan masukan nilai dalam rentang nilai 0-100. Terimakasih
EV02	TC11	Tidak ada pesan Nilai terinput dalam form nilai	<b>Muncul pesan :</b> Nilai berhasil diinput
EV03	TC12	Nilai yang anda dapat dimasukkan adalah antara 0-100, silahkan perbaiki kembali. Nilai yang lebih dari 100 akan mengganggu akurasi perhitungan	<b>Perbaikan pesan :</b> Mohon maaf nilai yang anda masukan belum sesuai, silahkan masukan nilai dalam rentang nilai 0-100. Terimakasih

Sumber : Hasil Penelitian (2020)

Pertama keluaran yang dihasilkan ketika melakukan input nilai kurang dari nol. Saat ini keluaran yang dihasilkan yaitu : "Nilai huruf untuk angka -0 2 belum di konfigurasi secara sesuai, harap menghubungi bagian admin atau puskom", sebaiknya pesan peringatan yang dikeluarkan bisa berupa pesan yang lebih dipahami oleh pengguna. Contohnya dapat diperbaiki menjadi "Mohon maaf nilai yang anda masukan belum sesuai, silahkan masukan nilai dalam rentang nilai 0-100. Terimakasih".

Kedua keluaran yang dihasilkan ketika melakukan input nilai antara nilai nol dan seratus. Saat ini tidak ada keluaran yang dihasilkan. Sebaiknya ditambahkan pesan notifikasi yang dikeluarkan. Contohnya dapat ditambahkan pesan : "Nilai berhasil diinput".

Ketiga keluaran yang dihasilkan ketika melakukan input nilai lebih dari seratus. Saat ini keluaran yang dihasilkan yaitu : "Nilai yang anda dapat dimasukkan adalah antara 0-100, silahkan perbaiki kembali. Nilai yang lebih dari 100 akan mengganggu akurasi perhitungan", sebaiknya pesan peringatan yang dikeluarkan bisa berupa pesan yang lebih dipahami oleh pengguna. Contohnya dapat diperbaiki menjadi "Mohon maaf nilai yang anda masukan belum sesuai, silahkan masukan nilai dalam rentang nilai 0-100. Terimakasih".

Dengan dihasilkannya rekomendasi berdasarakan evaluasi hasil pengujian, maka dapat dikatakan bahwa aplikasi ecampus sudah melakukan fungsi yang sesuai kebutuhan namun ada beberapa rekomendasi untuk perbaikan kedepannya agar aplikasi ecampus dapat terus melakukan *continous improvement*.



#### 4. Kesimpulan

Pengujian telah berhasil dilakukan menggunakan tujuh tahapan yang ada pada teknik pengujian *equivalence partitioning*, dimulai dari menentukan *use case*, menentukan kriteria dari setiap *use case*, membuat partisi berdasarkan kriteria, membuat partisi, menentukan data uji, menentukan kasus uji, melakukan pengujian berdasarkan kasus uji, dan evaluasi. Dari hasil pengujian dapat terlihat bahwa aplikasi ecampus memiliki fungsi yang sesuai harapan, namun masih ditemukannya beberapa temuan yang ada pada tahap evaluasi yaitu mengenai pesan peringatan yang muncul ketika melakukan pengujian input nilai. Dari hasil evaluasi dihasilkan beberapa rekomendasi untuk mengganti dan menambahkan kalimat pesan peringatan dengan tujuan agar pengguna lebih memahami isi pesan peringatan yang muncul mengenai ketidaksesuaian yang dihasilkan pada keluaran aplikasi.

Namun dari segi penggunaan teknik pengujian, teknik *equivalence partitioning* dapat digunakan dengan mudah dalam melakukan pengujian dengan berdasarkan data uji dan kasus uji yang mendetail sesuai dengan *use case* dan kriteria yang ditentukan sebelumnya. Dengan adanya penelitian ini diharapkan untuk penelitian selanjutnya dapat dilakukan pengujian aplikasi menggunakan teknik pengujian *black box* yang lain atau bisa melakukan pada aplikasi lainnya.

#### Referensi

- [1] D. Febiharsa, I. M. Sudana, and N. Hudallah, "Uji Fungsionalitas (BlackBox Testing) Sistem Informasi Lembaga Sertifikasi Profesi (SILSP) Batik Dengan AppPerfect Web Test Dan Uji Pengguna," *JOINED J.*, vol. 1, no. 2, pp. 117–126, 2018.
- [2] N. Safitri and R. Pramudita, "Pengujian Black Box Menggunakan Metode Cause Effect Relationship Testing," *Inf. Syst. Educ. Prof.*, vol. 3, no. 1, pp. 101 – 110, 2018.
- [3] T. S. Jaya, "Pengujian Aplikasi dengan Metode Blackbox Testing Boundary Value Analysis (Studi Kasus: Kantor Digital Politeknik Negeri Lampung)," *J. Inform. Pengemb. IT*, vol. 3, no. 2, pp. 45–46, 2018.
- [4] M. A. Setiawan, K. Muludi, and A. R. Irawati, "Pengembangan aplikasi kkn berbasis android (studi kasus universitas lampung)," vol. 1, no. 1, pp. 1–7, 2019.
- [5] Zuriati, D. K. Widyawati, I. S. Sitanggang, and A. Buowo, "Teknik Pengujian Boundary Value Analysis Pada Aplikasi Learning Management System Polinela," *TAM (Technology Accept. Model.)*, vol. 9, no. 2, pp. 86–92, 2018.
- [6] I. G. S. Aryandana, A. E. Permanasari, and T. B. Adji, "Comparing method equivalence class partitioning and boundary value analysis with study case add medicine module," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 732, no. 1, 2020.
- [7] M. S. Mustaqbal, R. F. Firdaus, and H. Rahmadi, "Pengujian Aplikasi Menggunakan Black Box Testing Boundary Value Analysis," *Penguji. Apl. Menggunakan Black Box Test. Bound. Value Anal. (Studi Kasus Apl. Prediksi Kelulusan SNMPTN)*, vol. 1, no. 3, p. 34, 2015.
- [8] W. N. Cholifah, Y. Yulianingsih, and S. M. Sagita, "Pengujian Black Box Testing pada Aplikasi Action & Strategy Berbasis Android dengan Teknologi Phonegap," *STRING (Satuan Tulisan Ris. dan Inov. Teknol.)*, vol. 3, no. 2, p. 206, 2018.
- [9] I. P. A. Prayudha, R. S. Hartati, and Y. Divayana, "on Learning Management System Applications," vol. 4, no. 2, pp. 25–28, 2019.
- [10] S. Ikhlashi, "Komparasi Dua Teknik Pengujian Perangkat Lunak Equivalence Partitioning Dan Boundary Value Analysis," vol. 5, no. 1, pp. 978–979, 2016.
- [11] A. Krismadi, A. F. Lestari, A. Pitriyah, I. W. P. A. Mardangga, M. Astuti, and A. Saifudin, "Pengujian Black Box berbasis Equivalence Partitions pada Aplikasi Seleksi Promosi Kenaikan Jabatan," *J. Teknol. Sist. Inf. dan Apl.*, vol. 2, no. 4, p. 155, 2019.
- [12] F. C. Ningrum, D. Suherman, S. Aryanti, H. A. Prasetya, and A. Saifudin, "Pengujian Black Box pada Aplikasi Sistem Seleksi Sales Terbaik Menggunakan Teknik Equivalence Partitions," *J. Inform. Univ. Pamulang*, vol. 4, no. 4, p. 125, 2019.
- [13] T. Hidayat and M. Muttaqin, "Pengujian Sistem Informasi Pendaftaran dan Pembayaran Wisuda Online menggunakan Black Box Testing dengan Metode Equivalence Partitioning dan Boundary Value Analysis," *J. Tek. Inform. UNIS JUTIS*, vol. 6, no. 1, pp. 2252–5351, 2018.
- [14] M. Komarudin, "Pengujian perangkat Lunak metode Black box berbasis partitions pada aplikasi sistem informasi di sekolah," *J. Mikrotik*, vol. o6, no. 3, pp. 02-16, 2016.

- [15] V. Akshatha and V. Illango, "A Comparative Analysis On Equivalence class partitioning And Boundary value analysis," *Int. J. Recent Trends Eng. Res.*, vol. 4, no. 3, pp. 542–554, 2018.
- [16] Y. Irawan, S. Muzid, N. Susanti, and R. Setiawan, "System Testing using Black Box Testing Equivalence Partitioning (Case Study at Garbage Bank Management Information System on Karya Sentosa)," *ICCSET*, pp. 1–7, 2018.
- [17] D. Bhagat, "Effective testing techniques," *Int. J. Multidiscip. Res. Dev.*, vol. 3, no. 2, pp. 21–23, 2016.