

Black-Box Testing pada Sistem Evaluasi Capaian Pembelajaran Berdasarkan Outcome Based Education

Yulherniwati[#], Andre Febbrian Kasmar[#], Ronal Hadi[#], Humaira[#], Ervan Asri[#]

[#] Jurusan Teknologi Informasi Politeknik Negeri Padang, Indonesia
E-mail: yulherniwati[at]pnp.ac.id

ABSTRACTS

The implementation of outcome based education curricula requires an evaluation system capable of objectively and systematically measuring the achievement of graduation learning outcomes (GLO). Therefore, comprehensive testing of the GLO evaluation system is essentially to ensure its reliability. This study designs functional testing using a black box-testing testing approach, employing techniques such as equivalence class partitioning, boundary value analysis and state transition testing. The test case design process follows the software testing process according to ISTQB, focusing on the initial three stages: planning, analysis and design. The findings indicate that this approach is effective in verifying the quality of GLO evaluation system based on OBE. The resulting test case design can serve as a reference for subsequent testing stages, both manual and automated.

Manuscript received Sep 10, 2024;
revised Apr 16, 2025. accepted Jun
19, 2024 Date of publication Jun
30, 2025. International Journal,
JITSI : Jurnal Ilmiah Teknologi
Sistem Informasi licensed under a
Creative Commons Attribution-
Share Alike 4.0 International
License



ABSTRAK

Penerapan kurikulum berdasarkan Outcome Based Education (OBE) menuntut sistem evaluasi yang mampu mengukur capaian pembelajaran lulusan (CPL) secara objektif dan terstruktur. Oleh karena itu, diperlukan pengujian menyeluruh terhadap sistem evaluasi CPL untuk memastikan keandalannya. Penelitian ini merancang pengujian fungsional menggunakan pendekatan black-box testing dengan teknik, equivalence class partitioning, boundary value analysis dan state transition testing. Perancangan kasus uji mengikuti proses pengujian perangkat lunak menurut ISTQB dengan memfokuskan pada tiga tahap awal yaitu perencanaan, analisis dan perancangan pengujian. Temuan penelitian menunjukkan bahwa pendekatan ini efektif dalam memverifikasi kualitas evaluasi CPL berdasarkan OBE. Rancangan kasus uji yang dihasilkan dapat dijadikan acuan untuk tahap pengujian selanjutnya, baik secara manual maupun otomatis.

Keywords / Kata Kunci — *Outcome based Education (OBE); Graduate Learning Outcomes (CPL); Black Box Testing; Functional Testing; Test Case;*

CORRESPONDING AUTHOR

Yulherniwati
Jurusan Teknologi Informasi Politeknik Negeri Padang, Indonesia
Email: yulherniwati[at]pnp.ac.id

1. PENDAHULUAN

Outcome based education dalam sistem pendidikan tinggi menekankan bahwa hasil utama dari proses pembelajaran adalah pencapaian kemampuan mahasiswa setelah menyelesaikan studi [1],[2]. Kurikulum OBE disusun menggunakan indikator Capaian Pembelajaran Lulusan (CPL) yang harus dapat diukur secara sistematis dan terdokumentasi. Evaluasi CPL dilakukan dengan memetakan antara CPL dan Capaian Pembelajaran Mata

Kuliah (CPMK), menghitung CPL berdasarkan nilai mahasiswa yang diperoleh dalam perkuliahan, hingga membuat laporan evaluasi [3]. Pengolahan data yang masih manual atau menggunakan aplikasi *spreadsheet* / *word processor* untuk evaluasi ketercapaian CPL ini, masih dirasakan sulit dan memakan waktu karena melibatkan data yang banyak dan proses yang kompleks [4]. Oleh karena itu, diperlukan sistem informasi yang dapat memudahkan evaluasi ketercapaian CPL. Banyak institusi pendidikan tinggi mulai mengembangkan sistem informasi untuk evaluasi CPL yang terintegrasi dengan kurikulum OBE. Sistem Evaluasi CPL berdasarkan OBE ini perlu dipastikan tidak hanya kelengkapan fiturnya, tetapi juga keandalannya dalam menjalankan fungsi-fungsi utama tanpa kesalahan. Untuk itu, proses pengujian perangkat lunak (software testing) menjadi tahap penting dalam siklus pengembangan sistem.

Pengujian terhadap sistem perlu dilakukan secara sistematis, menggunakan dokumentasi kasus uji (*test case*) yang terstruktur. Menurut *International Testing Quality Board* (ISTQB) terdapat tahapan sistematis proses pengujian yang terdiri dari serangkaian aktivitas, termasuk di dalamnya adalah perancangan pengujian. [5], [6] Dalam tahap ini, terdapat teknik-teknik untuk mendapatkan kasus uji dengan pendekatan *black box testing* yang dapat mengevaluasi fungsionalitas perangkat lunak berdasarkan spesifikasi tanpa memperhatikan struktur internal kodenya [7]. Pendekatan ini cocok diterapkan dalam pengujian fungsional sistem yang interaksinya berbasis antarmuka pengguna. Terdapat sejumlah teknik-teknik dalam pendekatan *black-box testing*, seperti *equivalence partitioning*, *boundary value analysis*, *use case testing*, *decision table*, *state transition testing* [8].

Beberapa penelitian sebelumnya telah menerapkan *black box testing* dalam pengujian sistem informasi pendidikan [8, 9, 10], namun perancangan khusus untuk sistem evaluasi CPL berbasis OBE masih sangat terbatas. Selain itu perancangan kasus uji yang mengikuti tahapan proses pengujian menurut ISTQB belum banyak dijadikan fokus studi. Penelitian ini berangkat dari pertanyaan: “**Bagaimana merancang kasus uji yang efektif dan sistematis menggunakan pendekatan black-box testing pada sistem evaluasi CPL berbasis OBE?**” Tujuan dari penelitian ini adalah untuk: merancang kasus uji berdasarkan kebutuhan fungsional sistem evaluasi CPL, menerapkan pendekatan *black-box testing* dalam proses perancangan pengujian, dan mendokumentasikan proses pengujian berdasarkan tahapan menurut ISTQB

2. METODOLOGI PENELITIAN

2.1. Objek Penelitian

Objek dalam penelitian ini adalah sistem evaluasi CPL berdasarkan OBE yang dirancang untuk mendukung kurikulum OBE berdasarkan studi kasus di Jurusan Teknologi Informasi Politeknik Negeri Padang, untuk menghasilkan evaluasi CPL yang terdokumentasi dan dapat dianalisis oleh koordinator program studi. Proses evaluasi ketercapaian CPL memiliki fitur utama:

- Input dan pemetaan antara CPL dan CPMK serta penentuan bobot kontribusi CPMK terhadap CPL
- Input nilai mahasiswa per CPMK oleh dosen pengampu
- Penghitungan CPL berdasarkan bobot kontribusi CPMK dan nilai mahasiswa per CPMK
- Pembuatan laporan evaluasi CPL dalam bentuk tabel dan grafik

2.2. Pendekatan Pengujian

Pengujian dilakukan menggunakan pendekatan *black-box testing*, untuk menguji input-output sistem tanpa memperhatikan struktur kode internal. Pendekatan ini cocok digunakan untuk memverifikasi fungsionalitas sistem dari sudut pandang pengguna. Teknik perancangan kasus uji yang digunakan meliputi:

- Equivalence Partitioning*: mengelompokkan data uji menjadi kelas valid dan kelas tidak valid
- Boundary value analysis*: menguji nilai-nilai di atas batas bawah dan atas input.
- State transition testing*: memvalidasi perubahan status dalam sistem untuk memastikan struktur CPL-CPMK siap digunakan untuk evaluasi CPL

2.3. Siklus Hidup Pengujian

Pengujian fungsional dilakukan dengan mengikuti proses pengujian menurut ISTQB yang terdiri dari perencanaan, peninjauan dan pengendalian, analisis, perancangan, implementasi, pelaksanaan hingga penutupan pengujian. Pada penelitian ini, proses yang digunakan difokuskan pada tiga tahapan awal dalam proses pengujian, dimulai dari perencanaan, analisis hingga perancangan pengujian:

- Perencanaan pengujian ditujukan untuk menetapkan atau memperbarui rencana pengujian sistem evaluasi CPL berdasarkan OBE terkait dengan tujuan, aktivitas, pendekatan, teknik dan tugas pengujian, serta penjadwalan dan metrik pengujian.
- Analisis pengujian dilakukan dengan mengidentifikasi kondisi pengujian (*test condition*) berdasarkan basis pengujian (*test basis*). Analisis dilakukan terhadap desain sistem, kode dan dokumentasi seperti diagram UML serta daftar resiko untuk mengidentifikasi fitur yang akan diuji. Keterlacakan antara elemen sistem dan kondisi uji yang relevan juga didokumentasikan.

Perancangan pengujian mencakup penyusunan kasus uji (test case) logis yang memuat urutan langkah uji, input, dan output yang diharapkan sesuai dengan kondisi uji yang telah dianalisis. Data uji ditentukan secara menyeluruh mencakup input valid (positif) dan invalid (negatif) untuk menguji respon sistem terhadap berbagai scenario.

3. HASIL DAN PEMBAHASAN

3.1. Hasil

3.1.1. Perencanaan Pengujian

Pengujian dilakukan untuk memastikan bahwa sistem dapat menjalankan fungsionalitas utama seperti input CPL/CPMK, pemetaan CPMK dengan CPL, input nilai mahasiswa per CPMK, perhitungan CPL dan pelaporan hasil evaluasi. Aktivitas pengujian, mencakup analisis kebutuhan, identifikasi kondisi uji, perancangan kasus uji, serta dokumentasi pengujian. Pendekatan yang digunakan adalah pengujian fungsionalitas dengan penerapan teknik *black box testing* seperti *equivalence partitioning*, *boundary value analysis*, *decisin table* dan *state transition*. Tugas pengujian disesuaikan dengan jenis fitur yang diuji dan disusun secara sistematis, dan dijadwalkan. Untuk mengukur efektivitas pengujian ditetapkan metrik yaitu *traceability index* dan rasio antara *positive test case* dan *negative tet case*.

3.1.2. Analisis Pengujian

Analisis pengujian dilakukan dengan mengidentifikasi kondisi pengujian (*test condition*) berdasarkan basis pengujian (*test basis*). Basis pengujian yang digunakan adalah dokumen kebutuhan sistem, *use case diagram* dan dan rancangan sistem. Analisis pengujian menghasilkan identifikasi fitur yang akan diuji sebagaimana diperlihatkan dalam Tabel 1.

TABEL 1. Hasil Analisis pengujian

Modul/Use Case	Fitur yang Dianalisis	Kondisi Uji Utama
Input CPL/CPMK	Form entri data CPL/CPMK	Validasi format dan kelengkapan data
Pemetaan CPL-CPMK	Relasi antar data	Konsistensi pemetaan
Bobot kontribusi	Penyesuaian nilai kontribusi	Rentang nilai valid
Perhitungan CPL	Logika akumulasi capaian	Kesesuaian perhitungan akhir
Pelaporan hasil	Tabel dan grafik evaluasi	Akurasi tampilan hasil akhir

Daftar kondisi pengujian ini menjadi dasar penyusunan kasus uji (*test case*).

3.1.3. Perancangan Pengujian

Hasil perancangan pengujian mencakup:

- 48 kasus uji yang mencakup seluruh proses utama dalam evaluasi CPL, mulai dari input, pemetaan, perhitungan hingga pelaporan.
- Teknik yang digunakan, terdiri dari:
 - Equivalence Partitioning*: mengelompokkan data uji menjadi kelas valid dan kelas tidak valid
 - Boundary value analysis*: menguji nilai-nilai di atas batas bawah dan atas input.
 - State transition testing*: memvalidasi perubahan status sistem dalam menstrukturkan CPL-CPMK sebelum siap digunakan untuk evaluasi CPL.

3.1.4. Dokumentasi Kasus Uji berdasarkan Teknik Pengujian

a. State Transition Testing (STT)

Teknik pengujian *State Transition Testing* menggunakan definisi *State* untuk merancang *State Transition Table*. Tabel 2 memperlihatkan definisi *State* dalam sistem evaluasi CPL, khususnya dalam penstrukturkan CPL-CPMK yang menjadi dasar utama dalam evaluasi CPL pada kurikulum OBE.

TABEL 1. Defini State

Kode State	Nama State	Deskripsi
S0	Awal	Sistem siap menerima input
S1	CPL Diinput	Data CPL berhasil disimpan dengan format, panjang karakter, dan isi data yang valid.
S2	CPL Tervalidasi	Data CPL sudah divalidasi sehingga jumlah CPL yang tersimpan sudah memenuhi ketentuan jumlah minimal

			(misalnya 7) dan jumlah maksimal (misalnya 12) yang ditetapkan program studi.
S3	CPMK Diinput		Data CPMK berhasil disimpan dengan format, panjang karakter, dan isi data yang valid.
S4	CPMK Tervalidasi		Data CPMK sudah divalidasi sehingga minimal ada 1 CPMK per mata kuliah
S5	Pemetaan Diinput	CPMK-CPL	Pemetaan antara CPMK dan CPL berhasil disimpan
S6	Pemetaan Tervalidasi	CPMK-CPL	Pemetaan antara CPMK dan CPL sudah divalidasi sehingga minimal setiap CPL direlasikan dengan minimal 1 CPMK dan sebaliknya, tidak ada duplikasi pemetaan CPMK-CPL.
S7	Bobot Kontribusi Diinput		Penentuan bobot kontribusi CPMK terhadap CPL berhasil disimpan dengan format angka yang valid.
S8	Penentuan Bobot Kontribusi Tervalidasi		Penentuan bobot kontribusi CPMK terhadap CPL sudah divalidasi sehingga total bobot kontribusi per CPL = 100%.
SF	Siap Uji Evaluasi CPL		Semua elemen (CPL, CPMK, pemetaan, bobot) sudah lengkap dan tervalidasi

State Transition Table dijadikan landasan untuk menghasilkan kasus uji, seperti yang diperlihatkan pada Tabel 3.

TABLE 3. Tabel Kasus Uji berdasarkan State Transition Table

State Transition Table			Kasus Uji secara umum
State Awal	Event	State Akhir	
S0	Input CPL	S1	Validasi format, panjang karakter dan isi data CPL yang diinput
S1	Klik Validasi CPL	S2	Validasi jumlah CPL minimal dan maksimal yang ditetapkan program studi.
S2	Input CPMK	S3	Validasi format, panjang karakter dan isi data CPMK yang diinput
S3	Klik Validasi CPMK	S4	Validasi jumlah CPMK minimal satu per matakuliah
S4	Input pemetaan CPMK ke CPL	S5	Validasi relasi CPMK-CPL yang diinput
S5	Klik Validasi pemetaan CPMK-CPL	S6	Validasi relasi CPMK-CPL yang tersimpan sehingga semua CPL dan CPMK terhubung dan tidak ada duplikasi
S6	Input bobot kontribusi CPMK-CPL	S7	Validasi isi (>0,00 s.d 100,00) dan format angka bobot kontribusi CPMK-CPL yang diinput
S7	Klik Validasi bobot kontribusi CPMK-CPL	S8	Validasi total bobot kontribusi CPMK per CPL= 100%
S8	Klik Finalisasi	SF	Validasi nilai semua data yang sudah tersimpan sesuai ketentuan penstrukturan CPL – CPMK, sehingga data CPL-CPMK dikunci untuk perubahan.

- b. *Equivalence Class Partitioning* dan *Boundary Value Analysis*
Equivalence Class Partitioning (ECP) dan *Boundary Value Analysis* (BVA) digunakan dalam validasi input pada form penambahan pengubahan data serta untuk validasi pembacaan data pada pengolahan data,

TABLE 4. Kasus Uji berdasarkan teknik ECP dan BVA

No.	Kasus Uji	Deskripsi	Input	Output yang diharapkan	Teknik
1	TC01.A-1	Input CPL dengan semua field valid	Kode: CPL1, Deskripsi: 'Menguasai konsep rekayasa perangkat lunak'	Input diterima	ECP
2	TC01.A-2	Input CPL dengan kode kosong	Kode: ", Deskripsi: 'Menguasai konsep rekayasa perangkat lunak'	Input Ditolak Pesan: 'Kode CPL wajib diisi'	ECP
3	TC01.A-3	Input CPL dengan deskripsi kosong	Kode: CPL2, Deskripsi: "	Input Ditolak Pesan: 'Deskripsi CPL wajib diisi'	ECP
4	TC01.A-4	Input CPL dengan panjang kode maksimal (10 karakter)	Kode: 'CPLABCDE12'	Input diterima	BVA
5	TC01.A-5	Input CPL dengan panjang kode lebih dari batas (11 karakter)	Kode: 'CPLABCDEFGHI'	Input Ditolak Pesan: 'Kode maksimal 10 karakter'	BVA
6	TC01.B-1	Validasi jumlah CPL yang valid (8-12)	Jumlah CPL:8	Validasi diterima	ECP
7	TC01.B-2	Validasi jumlah CPL yang tidak valid (<8 atau >12)	Jumlah CPL:14	Validasi ditolak	ECP
8	TC02.A-1	Input CPMK dengan format valid	Kode: CPMK1, Deskripsi: 'Menjelaskan prinsip pengujian'	Input diterima	ECP
9	TC02.A-2	Input CPMK dengan kode kosong	Kode: ", Deskripsi: '...'	Input Ditolak Pesan: 'Kode CPMK wajib diisi'	ECP
10	TC02.A-3	Input CPMK dengan kode duplikat	Kode: CPMK1 (sudah ada), Deskripsi: '...'	Input Ditolak Pesan: 'Kode CPMK sudah digunakan'	ECP
11	TC02.A-4	Input CPMK dengan panjang deskripsi > 255 karakter	Kode: CPMK9, Deskripsi: [>255 char]	Input Ditolak Pesan: 'Deskripsi terlalu panjang'	BVA
12	TC02.B-1	Validasi jumlah CPMK yang valid (>=1)	Jumlah CPMK:2	Validasi diterima	ECP
13	TC02.B-2	Validasi jumlah CPMK yang tidak valid (<1)	Jumlah CPL:0	Validasi ditolak	ECP
14	TC03.A-1	Pemetaan CPMK yang valid ke CPL	CPMK1 → CPL1	Input diterima Relasi tersimpan	ECP
15	TC03.A-2	Pemetaan tanpa memilih CPMK	-, CPL1	Pesan: 'CPMK harus dipilih'	ECP

16	TC03.A-3	Pemetaan CPMK ke CPL ganda (duplikat)	CPMK1 → CPL1 (sudah ada)	Input Ditolak Pesan: 'Relasi sudah ada'	ECP
17	TC03.A-4	Pemetaan CPMK ke CPL dengan input tak terdaftar	CPMK9 → CPL1 (CPMK9 tidak ada)	Input Ditolak Pesan: 'CPMK tidak ditemukan'	ECP
18	TC03.B-1	Pemetaan CPMK ke CPL		Input Ditolak Pesan: 'CPMK tidak ditemukan'	ECP
19	TC03.B-2	Pemetaan CPL tanpa	-, CPL1	Validasi ditolak	ECP
20	TC03.B-3	Pemetaan CPMK ke CPL ganda (duplikat)	CPMK1 → CPL1 (sudah ada)	Validasi ditolak	ECP
21	TC04.A-1	Input bobot kontribusi valid (>0 dan ≤100)	CPL1 ← CPMK1: 30%	Input Diterima	ECP
22	TC04.A-2	Input bobot berupa string	CPL1 ← CPMK1: 'tiga puluh'	Input Ditolak	ECP
23	TC04.A-3	Input bobot = 0	CPL1 ← CPMK1: 0	Input Ditolak karena di bawah batas minimum	ECP
24	TC04.A-4	Input bobot negatif	CPL1 ← CPMK1: -20%	Input Ditolak Pesan: 'Bobot tidak boleh negatif'	ECP
25	TC04.A-5	Input bobot >100	CPL1 ← CPMK1: 110	Input Ditolak karena melebihi batas maksimum	ECP
26	TC04.A-6	Input bobot = 1 (batas bawah valid)	CPL1 ← CPMK1: 1	Input Diterima	BVA
27	TC04.A-7	Input bobot = 100 (batas atas valid)	CPL1 ← CPMK1: 100	Input Diterima	BVA
28	TC04.A-8	Input bobot = 101 (batas atas +1)	CPL1 ← CPMK1: 101	Input Ditolak	BVA
29	TC04.B-1	Total bobot semua CPMK = 100% ke satu CPL	CPMK1: 50%, CPMK2: 50%	Validasi Diterima	ECP
30	TC04.B-2	Total bobot semua CPMK > 100% ke satu CPL	CPMK1: 50%, CPMK2: 60%	Validasi Ditolak	ECP
31	TC04.B-3	Total bobot semua CPMK < 100% ke satu CPL	CPMK1: 50%, CPMK2: 40%	Validasi Ditolak	ECP
32	TC04.B-4	Semua CPL-CPMK terhubung, total bobot kontribusi = 100%, format angka valid, tidak ada perubahan data sejak validasi terakhir	CPL1 ← CPMK1 (100%): 80 CPL2 ← CPMK2 (60%): 60, CPMK3(40%):75 ... CPMK8 ← CPMK20 (55%): 60, CPMK21(45%):85	Finalisasi diterima	ECP

33	TC04.B-5	Ada satu CPL belum dipetakan dengan CPMK	-, CPL3	Finalisasi ditolak	ECP
34	TC04.B-6	Total bobot kontribusi hanya 90%	CPMK1: 50%, CPMK2: 40%	Finalisasi ditolak	ECP
35	TC05-1	Format angka bobot tidak valid (format desimal tidak sesuai regional)	Nilai CPMK1: 63.33 (regional Indonesia)	Finalisasi ditolak	ECP
36	TC05-2	Input nilai mahasiswa per CPMK kosong	Nilai CPMK1:	Input Ditolak	ECP
37	TC05-3	Input nilai mahasiswa per CPMK dengan bilangan negatif	Nilai CPMK1: -10	Input Ditolak	ECP
38	TC05-4	Input nilai mahasiswa per CPMK dengan bilangan desimal yang tidak sesuai format region	Nilai CPMK1: 70.00	Input Ditolak	ECP
39	TC05-5	Input nilai mahasiswa per CPMK dengan 0	Nilai CPMK1: 0	Input Diterima	BVA
40	TC05-6	Input nilai mahasiswa per CPMK dengan 100	Nilai CPMK1: 100	Input Diterima	BVA
41	TC06-1	Hitung CPL dengan bobot kontribusi valid	CPMK1: 30%, CPMK2: 70% (CPL1)	Nilai CPL1 dihitung dengan benar	ECP
42	TC06-2	Hitung CPL dengan nilai CPMK 0	CPMK1: 50% (0), CPMK2: 50% (80)	Nilai CPL = 40	ECP
43	TC06-3	Hitung CPL saat salah satu nilai CPMK tidak valid (belum dinilai/kosong)	CPMK1: 30% (nilai ada), CPMK2: 70% (nilai kosong)	Pesan: Nilai CPMK belum lengkap	ECP
44	TC06-4	Hitung CPL dengan nilai CPMK tidak valid (negatif)	CPMK1: 50% (-20), CPMK2: 50% (100)	Pesan: Nilai CPMK tidak valid (<0)	ECP
45	TC06-5	Hitung CPL dengan nilai CPMK tidak valid (> 100)	CPMK1: 50% (120), CPMK2: 50% (100)	Pesan: Nilai CPMK tidak valid (>100)	BVA
46	TC06-6	Hitung CPL dengan nilai CPMK tidak valid (format angka tidak sesuai region)	CPMK1: 50% (90.00), CPMK2: 50% (100)	Pesan: Nilai CPMK tidak valid	ECP
47	TC07-1	Laporan evaluasi CPL difilter berdasarkan mahasiswa, dengan data nilai yang sudah lengkap	Nama Mahasiswa: Hana, (Nilai CPMK lengkap)	Laporan tampil sesuai data CPL	ECP

48	TC07-2	Laporan evaluasi CPL difilter berdasarkan mahasiswa, dengan data nilai yang belum lengkap	Nama Mahasiswa: Erlan, (Nilai CPMK ada yang kosong)	Pesan: Data nilai belum tersedia	ECP
----	--------	---	---	----------------------------------	-----

c. Dokumentasi kasus uji gabungan teknik STT, ECP, BVA

Tabel 5 memperlihatkan Kasus Uji yang dihasilkan untuk setiap *use case* utama dari sistem, berdasarkan teknik *State Transition Testing*, *Equivalence Class Partitioning* dan *Boundary Value Analysis*.

TABLE 5. Dokumentasi Kasus Uji					
Use Case	State Awal	Event	State Akhir	Kasus Uji	Jenis Kasus Uji
Menginput CPL	S0	Input CPL	S1	TC01.A-1	Positif
	S0	Input CPL	Tetap S0	TC01.A-2, TC01.A-3, TC01.A-4, TC01.A-5	Negatif
	S1	Klik Validasi CPL	S2	TC01.B-1	Positif
	S1	Klik Validasi CPL	Tetap S1	TC01.B-2	Negatif
Menginput CPMK	S2	Input CPMK	S3	TC02.A-1	Positif
	S2	Input CPMK	Tetap S2	TC02.A-2, TC02.A-3, TC02.A-4, TC02.A-5	Negatif
	S3	Klik Validasi CPMK	S4	TC02.B-1	Positif
	S3	Klik Validasi CPMK	Tetap S3	TC02.B-2	Negatif
Memetakan CPMK dan CPL	S4	Input pemetaan CPMK ke CPL	S5	TC03.A-1	Positif
	S4	Input pemetaan CPMK ke CPL	Tetap S4	TC03.A-2, TC03.A-3, TC03.A-4	Negatif
	S5	Klik Validasi pemetaan CPMK-CPL	S6	TC03.B-1	Positif
	S5	Klik Validasi pemetaan CPMK-CPL	Tetap S5	TC03.B-2, TC03.B-3	Negatif
Menentukan bobot kontribusi CPMK ke CPL	S6	Input bobot kontribusi CPMK-CPL	S7	TC04.A-1	Positif
	S6	Input bobot kontribusi CPMK-CPL	Tetap S6	TC04.A-2 s.d TC04.A-8	Negatif
	S7	Klik Validasi bobot kontribusi CPMK-CPL	S8	TC04.B-1	Positif
	S7	Klik Validasi bobot kontribusi CPMK-CPL	Tetap S7	TC04.B-2, TC04.B-3	Negatif
	S8	Klik Finalisasi	SF	TC04.B-4	Positif
	S8	Klik Finalisasi	Tetap S8	TC04.B-5, TC04.B-6	Negatif
Menginput nilai mahasiswa per CPMK				TC05-1, TC05-5, TC05-6	Positif
				TC05-2, TC05-3, TC05-4	Negatif

Menghitung CPL		TC06-1	Positif
		TC06-2,	Negatif
		TC06-3,	
		TC06-4,	
		TC06-5,	
Membuat laporan evaluasi CPL		TC07-1	Positif
		TC07-2	Negatif

3.2. Pembahasan

Perancangan pengujian yang dilakukan telah mencakup semua fungsi utama berdasarkan dokumen kebutuhan sistem. Penggunaan teknik pengujian disesuaikan dengan karakteristik setiap modul (form input, kalkulasi numerik, transisi status dan alur kerja pengguna). Tahap analisis pengujian menghasilkan kondisi pengujian yang menjadi perantara antara kebutuhan sistem dan perancangan kasus uji. Walaupun pengujian belum dieksekusi, hasil perancangan ini menyediakan kerangka yang lengkap dan sistematis untuk digunakan pada tahap implementasi dan eksekusi berikutnya.

Untuk mengevaluasi efektivitas pengujian yang difokuskan hingga tahap perancangan kasus uji, digunakan beberapa metrik berikut:

- Tingkat keterlacakan anatara kebutuhan sistem dan kasus uji (*test case*) (Tracibility Index)
 Metrik ini membandingkan jumlah kebutuhan yang tercakup dalam test case dengan jumlah total kebutuhan fungsional. Keselarasan antara kebutuhan dan proses pengujian diperlukan, sehingga pengukuran tingkat keterlacakan (**traceability index**) penting dilakukan untuk memastikan semua prasyarat/kebutuhan (*requirement*) diuji [11]. Tabel 6 menampilkan penggunaan metrik traceability index pada sistem evaluasi CPL berdasarkan OBE.

TABLE 6. Traceability index

Kebutuhan Fungsional (<i>Functional Requirement</i>) Sistem	Use Case	Kasus Uji	Tracibility Index
FR1: Sistem dapat menerima input dan pemetaan antara CPL dan CPMK serta penentuan bobot kontribusi CPMK terhadap CPL	UC01: Menginput CPL UC02: Menginput CPMK UC03: Memetakan CPL dan CPMK UC04: Menentukan bobot kontribusi CPMK ke CPL	Seluruh Kasus uji yang diawali kode TC01, TC02, TC03, TC04	100%
FR2 : Sistem dapat menerima input nilai mahasiswa per CPMK oleh dosen pengampu	UC05: Menginput nilai mahasiswa per CPMK	Seluruh Kasus uji yang diawali kode TC05	100%
FR3: Sistem dapat menghitung CPL berdasarkan bobot kontribusi CPMK dan nilai mahasiswa per CPMK	UC06: Menghitung CPL	Seluruh Kasus uji yang diawali kode TC06	100%
FR4: Sistem dapat memfasilitasi pembuatan laporan evaluasi CPL	UC07: Membuat laporan evaluasi CPL	Seluruh Kasus uji yang diawali kode TC07	100%

- Rasio kasus uji positif (*positive test case*) dan kasus uji negatif (*Negative test case*):
 Metrik ini menggambarkan proporsi antara kasus uji yang menguji skenario benar dan skenario salah/kegagalan sistem, karena keseimbangan antara pengujian normal dan pengujian batas/*error* perlu diperhatikan [12], [13]. Pengujian positif bertujuan untuk menunjukkan bahwa perangkat lunak bekerja sebagaimana mestinya; sedangkan pengujian negatif bertujuan untuk menunjukkan bahwa perangkat lunak tidak gagal saat diberikan masukan yang tidak valid atau tidak terduga [14]. Tabel 7 menampilkan rasio kasus uji positif dan negatif pada sistem evaluasi CPL berdasarkan OBE.

TABLE 7. Rasio kasus uji positif dan negatif

Jenis Kasus Uji	Jumlah	Persentase
Positif	12	25%
Negatif	36	75%
Total	48	100%

4. KESIMPULAN

Penelitian ini menerapkan pengujian fungsional dengan pendekatan *black-box testing* terhadap sistem evaluasi CPL berdasarkan OBE yang memfokuskan pada tiga tahap awal dari proses pengujian menurut ISTQB yang terdiri dari perencanaan, analisis dan perancangan pengujian. Hasil perancangan menunjukkan bahwa pendekatan *black box testing* memungkinkan pengujian fungsional secara sistematis tanpa memahami kode program. Teknik yang digunakan yaitu *Equivalence Partitioning*, *Boundary Value Analysis*, dan *State Transition Testing* efektif dalam merancang skenario uji untuk berbagai fitur utama sistem, seperti validasi input nilai, pemetaan CPMK ke CPL, perhitungan capaian, dan pelaporan hasil. Dokumentasi kasus uji yang dihasilkan dapat digunakan sebagai dasar dalam proses pemrosesan pengujian manual maupun sebagai dasar pengembangan pengujian otomatis di tahap selanjutnya. Pendekatan ini dapat diadaptasi untuk sistem serupa yang mengimplementasikan kurikulum OBE.

Saran untuk penelitian berikutnya adalah melanjutkan proses hingga tahap eksekusi pengujian, mengevaluasi efektivitas kasus uji dalam menemukan cacat (*defects*), serta mengintegrasikan dengan alat pengujian otomatis untuk meningkatkan efisiensi dan memudahkan *regression testing* dalam pengembangan sistem.

UCAPAN TERIMA KASIH

Penelitian ini terlaksana atas koordinasi dari Lembaga Penelitian dan Pengabdian kepada Masyarakat (LP2M) dan dukungan dana dari DIPA Politeknik Negeri Padang. Penulis menyampaikan terima kasih kepada seluruh pihak yang telah memberikan kontribusi dalam pelaksanaan penelitian ini.

REFERENSI

- [1] Direktorat Jenderal Pendidikan Tinggi, Riset, dan Teknologi, Panduan Penyusunan Kurikulum Pendidikan Tinggi Era Industri 4.0 untuk Mendukung Merdeka Belajar Kampus Merdeka (MBKM), Kementerian Pendidikan dan Kebudayaan RI, 2020.
- [2] Biggs, J., & Tang, C. (2022). *Teaching for Quality Learning at University: What the Student Does* (5th ed.). Open University Press.
- [3] Mahajan, R., & Singh, T. (2020). Outcome-Based Education (OBE): A novel approach to quality education. *International Journal of Research in Medical Sciences*, 8(4), 1451–1457.
- [4] Agarwal, P., & Ahmed, T. (2021). Automation of Outcome-Based Education Process using Web-based Systems. *International Journal of Emerging Technologies in Learning (iJET)*, 16(9), 92–102.
- [5] ISTQB, Standard Glossary of Terms Used in Software Testing, Version 4.0, International Software Testing Qualifications Board, 2023.
- [6] Myers, G. J., Sandler, C., & Badgett, T. (2019). *The Art of Software Testing* (4th ed.). Wiley.
- [7] P. Leloudas, *Introduction to Software Testing: A Practical Guide to Testing, Design, Automation, and Execution*. Ilioupoli, Greece: Apress, 2023.
- [8] Uddin, M. F., & Rahman, A. (2020). Functional testing of educational management system using black-box testing technique. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 11(3), 530–537.
- [9] Mishra, D., & Dash, S. (2021). Black-box testing of university ERP system: An empirical evaluation. *Journal of Engineering Research and Reports*, 21(7), 77–87.
- [10] Sari, P. K., & Harimurti, S. (2020). Evaluation of student information system using black box testing approach. *International Journal of Recent Technology and Engineering (IJRTE)*, 9(1), 300–304.
- [11] R. A. Khan and S. U. Khan, "Software requirements traceability and testing: A systematic literature review," *Journal of Systems and Software*, vol. 173, p. 110871, Jan. 2021.
- [12] Patel, N., & Patel, D. (2020). Importance of Negative Testing in Software Quality Assurance. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 6(1), 245–249.
- [13] P. C. Jorgensen, *Software Testing: A Craftsman's Approach*, 4th ed., Boca Raton, FL, USA: CRC Press, 2013.
- [14] ISTQB®, Certified Tester Foundation Level (CTFL) Syllabus v4.0, ISTQB®, 2023