

TEKNIK PENGUJIAN BOUNDARY VALUE ANALYSIS PADA APLIKASI LEARNING MANAGEMENT SYSTEM POLINELA

Zuriati¹, Dewi Kania Widyawati², Imas Sukaesih Sitanggang³, Agus Buowo⁴

^{1,2} Program Studi Manajemen Informatika, Jurusan Ekonomi dan Bisnis, Politeknik Negeri Lampung

^{3,4} Departemen Ilmu Komputer, FMIPA, Institut Pertanian Bogor

^{1,2} Jln. Soekarno Hatta No. 10 Rajabasa, Bandar Lampung, 35144, Indonesia

^{3,4} Jln. Raya Dramaga, Babakan, Dramaga, Bogor, Jawa Barat, 16680, Indonesia

¹zuriati_mi@polinela.ac.id, ²dewi_mi@polinela.ac.id, ³imas.sitanggang@ipb.ac.id, ⁴pudasha@gmail.com

ABSTRAK

Penelitian ini membahas proses pengujian terhadap aplikasi e-learning Politeknik Negeri Lampung yang dinamakan *Learning Management Sistem* (LMS) Polinela. 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. Terdapat beberapa teknik yang digunakan untuk menguji perangkat lunak diantaranya adalah teknik *black box testing*. Pada penelitian ini menggunakan *black box testing* dengan teknik *boundary value analysis* untuk menguji aplikasi e-learning yang telah diterapkan pada program studi Manajemen Informatika Politeknik Negeri Lampung. Teknik *boundary value analysis* bekerja dengan cara menentukan nilai batas bawah dan batas atas dari sebuah data yang akan diuji, kemudian data tersebut diinputkan ke aplikasi LMS Polinela. Tahapan penelitian yang dilakukan dimulai dengan menentukan fungsionalitas yang akan diuji, merancang skenario pengujian, menentukan data yang akan diuji, menentukan nilai batas atas dan bawah sesuai dengan struktur basisdata yang telah dibuat, melakukan percobaan pengujian, mendokumentasikan hasil penelitian, dan penarikan kesimpulan. Hasil pengujian menunjukkan bahwa tidak terdapat kesalahan pada aplikasi LMS Polinela saat melakukan validasi data yang akan diproses.

Kata Kunci : Blackbox Testing, *Boundary Value Analysis*, E-learning, Learning Management System.

ABSTRACT

This study discusses the testing process of the Lampung State Polytechnic e-learning application called the Polinela Learning Management System (LMS). The testing phase is one of the stages in the development cycle of a software. This stage is a series of activities carried out to find errors or shortcomings of a software or computer application. There are several techniques used to test software including black box testing techniques. In this study using black box testing with a boundary value analysis technique to test e-learning applications that have been applied to the Informatics Management program of the Lampung State Polytechnic. The boundary value analysis technique works by determining the lower and upper limit values of the data to be tested, then the data is inputted into the LMS Polinela application. The stages of research carried out begin by determining the functionality to be tested, designing test scenarios, determining the data to be tested, determining the upper and lower limit values according to the database structure that has been created, conducting testing experiments, documenting the results of research, and drawing conclusions. The test results show that there were no errors in the LMS Polinela application when validating the data to be processed.

Key Words: Blackbox Testing, *Boundary Value Analysis*, Learning Management System.

1. Pendahuluan

1.1 Latar Belakang Penelitian

Aplikasi *Learning Management System* Politeknik Negeri Lampung (LMS Polinela) merupakan aplikasi *e-learning* yang dikembangkan sebagai media ajar alternatif yang berfungsi melengkapi media ajar yang sudah ada dengan tujuan untuk membantu mahasiswa belajar mandiri agar hasil pembelajaran lebih optimal. Melalui aplikasi ini diharapkan dosen dan mahasiswa memiliki kemampuan dalam penggunaan teknologi informasi. Aplikasi LMS Polinela terdiri dari beberapa

modul program diantaranya: Pengelolaan Data Program Studi, Pengelolaan Data Matakuliah, Pengelolaan Data Dosen, Pengelolaan Materi, Pengelolaan Mahasiswa, Pengelolaan Tugas, UTS, UAS, dan Pengelolaan Nilai.

Siklus pengembangan aplikasi LMS Polinela telah melewati tahap analisis kebutuhan pengembangan [1] dan tahap desain dan implementasi [2]. Tahap berikutnya yang dilakukan adalah tahap pengujian. Tahap pengujian adalah bagian yang penting dan tidak dapat dipisahkan dalam proses pengembangan sebuah aplikasi, ia merupakan rangkaian kegiatan yang disusun

secara rinci dan dilakukan secara sistematis dengan tujuan mendeteksi perbedaan antara keluaran aplikasi dengan kondisi yang diharapkan [3]. Pengujian merupakan suatu kegiatan yang bertujuan untuk menemukan kesalahan atau kekurangan pada pengembangan aplikasi komputer [4,5,6,7,8]. Pengujian bertujuan untuk memeriksa kesesuaian fungsionalitas yang telah dirumuskan pada tahap sebelumnya tercapai atau tidak [9].

Teknik pengujian aplikasi komputer yang banyak digunakan adalah *black box testing*. Pengujian *black box testing* berusaha untuk menemukan kesalahan dalam beberapa kategori, diantaranya: fungsi yang tidak benar atau hilang, kesalahan *interface*, kesalahan pada struktur data atau akses basisdata, kesalahan pada kinerja, kesalahan pada saat inisialisasi dan terminasi, kesensitifan sistem terhadap nilai input tertentu, dan batasan dari suatu data [10,11]. Penelitian terdahulu telah banyak memanfaatkan teknik *black box testing* untuk pengujian aplikasi. Teknik ini memiliki beberapa jenis diantaranya: *boundary value analysis*, *equivalency partitions* dan *test case* [3].

Penelitian ini menerapkan teknik pengujian *black box testing boundary value analysis* untuk pengujian aplikasi LMS Polinela. Teknik *boundary value analysis* bekerja dengan cara menentukan nilai batas atas dan batas bawah dari sebuah data yang akan diuji, kemudian data tersebut diinputkan ke aplikasi LMS Polinela. Hasil keluaran aplikasi akan diamati dan dicatat apakah sudah sesuai dengan yang diinginkan. Hasil pengujian pada LMS Polinela menunjukkan tidak terdapat kesalahan saat melakukan validasi data, sehingga data yang disimpan pada basisdata sesuai dengan data yang diharapkan.

1.2 Tujuan Penelitian

Tujuan penelitian ini adalah:

1. Melakukan pengujian terhadap aplikasi *e-learning* LMS Polinela untuk menemukan kesalahan atau kekurangan pada aplikasi menggunakan teknik pengujian *blackbox testing boundary value analysis*
2. Hasil pengujian akan digunakan untuk perbaikan aplikasi LMS Polinela agar kinerja *e-learning* semakin baik.

II. Tinjauan Pustaka/Landasan Teori

2.1 Pengujian Blackbox Testing

Untuk meningkatkan kualitas dari sebuah perangkat lunak perlu dilakukan pengujian. Pengujian akan dilakukan semakin terukur dan sistematis jika perangkat lunak yang diuji semakin besar dan lengkap. Pengujian adalah satu kegiatan yang direncanakan secara teratur dan terencana untuk menguji atau mengevaluasi kebenaran yang diinginkan. Pengujian *blackbox* merupakan strategi pengujian yang memperhatikan spesifikasi perangkat lunak dan faktor fungsionalitas [10], hanya mengidentifikasi jenis kesalahan antar muka,

kesalahan fungsi, kesalahan dalam pemodelan data dan kesalahan dalam akses ke sumber data eksternal [10, 11].

2.2 Boundary Value Analysis

Boundary value analysis merupakan salah satu jenis teknik pengujian *blackbox* yang melakukan pengujian pada batas atas dan batas bawah dari suatu nilai yang diinput kedalam aplikasi. Ketentuan utama dari *boundary value analysis* adalah:

1. *Boundary value analysis* merupakan pelengkap teknik *equivalence class testing* yang hanya memperhatikan nilai input, sedangkan *boundary value analysis* juga memperhatikan nilai output.
2. *Boundary value analysis* menguji input pada batas atas maupun batas bawah sebuah nilai yang valid.

2.3. Penelitian Terdahulu

Penelitian terdahulu telah banyak memanfaatkan teknik *black box testing* untuk pengujian aplikasi karena dalam teknik ini Penguji hanya perlu memberikan masukan pada aplikasi dan mengamati keluaran yang dihasilkan apakah telah sesuai dengan hasil yang diharapkan atau belum. Teknik pengujian *black box testing* sangat banyak variasinya, diantaranya adalah: *boundary value analysis*, *equivalency partitions*, dan *test case*.

Boundary value analysis bekerja dengan cara menentukan batas atas dan batas bawah dari nilai yang akan diinputkan kepada aplikasi, hasil keluaran diamati apakah sesuai dengan yang diharapkan [3]. Teknik *boundary value analysis* telah diterapkan untuk menguji aplikasi prediksi kelulusan snmptn, hasil pengujian menemukan kekurangan pada aplikasi saat melakukan validasi data sehingga dapat mengurangi kesalahan dalam penyimpanan data [12]. Penerapan teknik pengujian *boundary value analysis* pada pengujian aplikasi sistem informasi manajemen konseling tugas akhir menunjukkan bahwa teknik *boundary value analysis* merupakan teknik yang efektif untuk menemukan kesalahan dan kekurangan didalam sistem. [13]. Demikian juga penerapan *boundary value analysis* pada aplikasi kantor digital Politeknik Negeri Lampung menunjukkan aplikasi mampu menangani kesalahan data hingga 91.67% [14].

Equivalency partitions merupakan metode yang bekerja berdasarkan ekuivalensi *class* untuk kondisi input yang merepresentasikan sekumpulan keadaan valid dan invalid dari data yang diujikan [3]. Teknik ini telah diterapkan untuk menguji sistem informasi sekolah [15], dan untuk menguji aplikasi desain test case secara otomatis [4].

Test case bekerja berdasarkan asumsi bahwa kasus uji yang dirancang untuk satu kelas mewakili semua nilai input yang lain dalam kelas yang sama [3]. *Test case* telah diterapkan untuk menguji aplikasi game first person shooter 3D Alien Hunter [16].

Kajian-kajian mendasar juga telah banyak

dilakukan untuk membandingkan kinerja dari berbagai teknik *black box testing*, diantaranya membandingkan kinerja teknik *boundary value analysis* dengan *equivalency partitioning* [17], selain itu secara umum juga terdapat kajian yang membandingkan berbagai teknik pada *black box testing* untuk menemukan kesalahan-kesalahan aplikasi [18].

III. Metode Penelitian

Tahapan penelitian sebagai berikut:

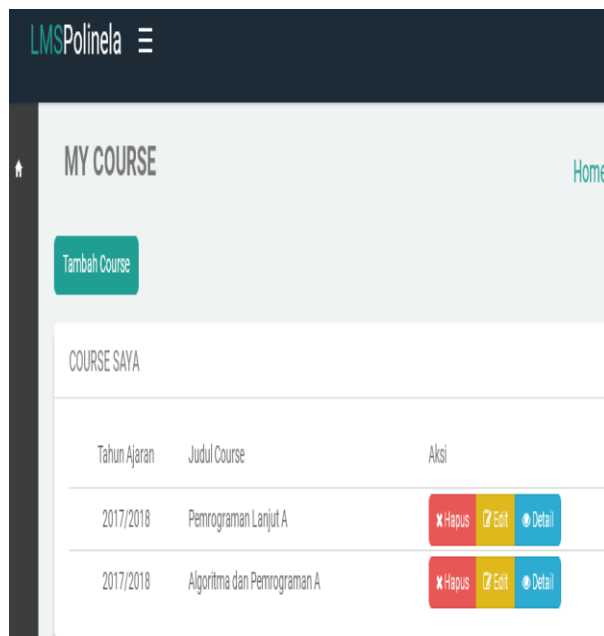
1. Menentukan fungsionalitas yang akan diuji.
2. Merancang skenario pengujian.
3. Menentukan data yang akan diuji.
4. Menentukan nilai batas atas dan bawah sesuai dengans struktur basisdata yang telah dibuat.
5. Melakukan Pengujian
6. Dokumentasi Hasil
7. Penarikan Kesimpulan

IV. Hasil dan Pembahasan

Fungsionalitas yang akan diuji pada pengujian aplikasi LMS Polinela adalah: fungsionalitas tambah course, fungsionalitas tambah materi, fungsionalitas tambah tugas, dan fungsionalitas soal UTS dan UAS. Berikut adalah penjelasan proses pengujian untuk setiap fungsionalitas yang akan diuji, penjelasan dilengkapi dengan proses pengujian, hasil pengujian, dan skenario pengujian untuk setiap field.

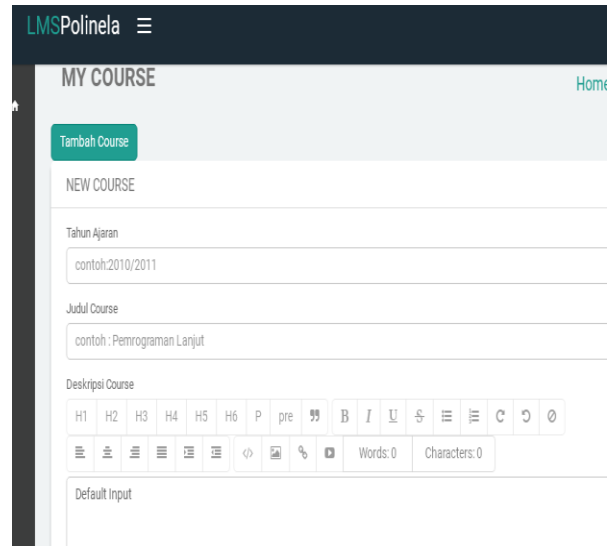
4.1 Pengujian Fungsionalitas Tambah Course.

Pada Gambar 1 disajikan form Tambah Course.



Gambar 1. Form Tambah Course

Pada Gambar 2 disajikan Detil Form Tambah Course.



Gambar 2. Detail Form Tambah Course

Ketentuan proses pengujian fungsionalitas yang dilakukan untuk form Tambah Course disajikan pada Tabel 1.

Tabel 1. Pengujian Fungsionalitas Form Tambah Course

No	Fungsi	Hasil yang diharapkan	Hasil percobaan	Simpulan
1.	Tombol tambah course	Dapat menambah course	Benar	Berhasil
2.	Masukkan tahun ajaran	Dapat mengisi tahun ajaran	Benar	Berhasil
3.	Masukkan judul course	Dapat mengisi judul course	Benar	Berhasil
4.	Masukkan deskripsi course	Dapat mengisi deskripsi course	Benar	Berhasil

Hasil pengujian untuk fungsionalitas Tambah Course, Masukkan Tahun Ajaran, Masukkan Judul Course, dan Masukan Deskripsi Course adalah berhasil.

Selanjutnya akan dilakukan pengujian pada field yang dipakai pada form Tambah Course. Tabel 2 menyajikan field yang ada pada form Tambah Course.

Tabel 2. Field Tambah Course

No	Nama field	Tipe data
1.	Tahun ajaran	String (9)
2.	Judul course	String (50)
3.	Deskripsi	String (100)

Skenario Pengujian Form Tambah Course

1. Pengujian field Tahun ajaran

Tipe data string tetapi yang diinputkan adalah rangkaian angka sebanyak 8 digit dan diantaranya ada tanda “/” contoh: 2017/2018. Hasil uji coba disajikan pada Tabel 3.

Tabel 3. Hasil Uji Coba Field Tahun Ajaran

Data uji	Skenario	Hasil	Kesimpulan
2017/2018/2	Jumlah karakter lebih banyak	Dapat diinputkan	Gagal
2017/18	Jumlah karakter lebih sedikit	Dapat diinputkan	Gagal
2017/2018	Jumlah karakter sesuai ketentuan	Dapat diinputkan	Sukses
	Kosong	Pesan Kesalahan	Sukses

Hasil pengujian untuk field tahun ajaran pada setiap batas nilai atas dan bawah dengan skenario jumlah karakter lebih banyak, jumlah karakter lebih sedikit, jumlah karakter sesuai ketentuan dan jumlah karakter kosong adalah berhasil.

2. Pengujian field Judul Course

Tipe data string sebanyak 50 karakter dengan ketentuan dapat mengandung spasi, contoh: Algoritma dan pemrograman. Hasil uji coba disajikan pada Tabel 4.

Tabel 4. Hasil Ujicoba Field Judul Course

Data uji	Skenario	Hasil	Kesimpulan
Analisa dan perancangan sistem informasi, Algoritma dan Bahasa Pemrograman	Jumlah karakter lebih banyak	Dapat diinputkan	Sukses
Analisa dan perancangan sistem informasi, Algoritma			
Algoritma	Jumlah karakter lebih sedikit	Dapat diinputkan	Sukses
	Kosong	Pesan Kesalahan	Sukses

Hasil pengujian untuk field judul course pada setiap batas nilai atas dan bawah dengan skenario jumlah karakter lebih banyak dari 50, jumlah karakter lebih sedikit dari 50, jumlah karakter 50 dan jumlah karakter kosong adalah berhasil.

3. Pengujian field Deskripsi

Tipe data string berjumlah 100 karakter dengan ketentuan dapat mengandung spasi. Hasil uji coba disajikan pada Tabel 5.

Tabel 5. Hasil Ujicoba Field Deskripsi

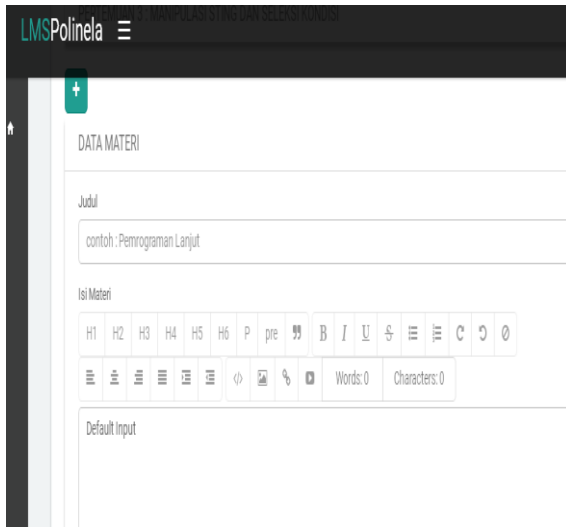
Data uji	Skenario	Hasil	Kesimpulan
Deskripsi matakuliah: Bahasa pemrograman yang digunakan adalah C Sharp (C#). Bahasa pemrograman C# berbasis visual sehingga cocok digunakan untuk pengembangan aplikasi yang memerlukan tampilan GUI yang menarik.	Jumlah karakter lebih banyak	Dapat diinputkan	Sukses
Capaian Pembelajaran 1. Memahami ruang lingkup C# 2. Bekerja dengan menu dan mampu menerapkannya dalam program C# 3. Mampu melakukan manipulasi string dan menerapkannya dalam program C# 4. Mampu menerapkan event dalam program C# 5. Mampu bekerja dengan basis data dan menerapkannya dalam C#			
Algoritma	Jumlah karakter lebih sedikit	Dapat diinputkan	Sukses
	Kosong	Pesan Kesalahan	Sukses

Hasil pengujian untuk field deskripsi pada setiap batas nilai atas dan bawah dengan skenario jumlah karakter lebih banyak dari 100, jumlah karakter lebih

sedikit dari 100, dan jumlah karakter kosong adalah berhasil.

4.2 Pengujian Fungsionalitas Tambah Materi.

Pada Gambar 3 disajikan Form Tambah Materi.



Gambar 3. Form Tambah Materi

Ketentuan proses pengujian fungsionalitas yang dilakukan untuk form Tambah Materi disajikan pada Tabel 6.

Tabel 6. Pengujian Fungsionalitas Form Tambah Materi

No.	Proses Uji	Hasil	Kesimpulan
1.	Klik tombol Tambah Materi	Dapat menambah materi	Sukses
2.	Masukkan judul materi	Dapat menambah judul materi	Sukses
3.	Masukkan isi materi	Dapat menambah isi materi	Sukses
4.	Klik tombol Submit Materi	Dapat menyimpan materi	Sukses

Hasil pengujian untuk fungsionalitas Tambah Materi, Masukkan Judul Materi, Masukkan Isi Materi, dan Tombol Submit Materi adalah berhasil.

Selanjutnya akan dilakukan pengujian pada field yang dipakai pada form Tambah Materi. Tabel 7 menyajikan field yang ada pada form Tambah Materi.

Tabel 7. Field Tambah Materi

No	Nama field	Tipe data
1.	Judul materi	String (30)
2.	Isi dan deskripsi materi	String (500)

No	Nama field	Tipe data
1.	Judul materi	String (30)
2.	Isi dan deskripsi materi	String (500)

Skenario Pengujian Form Tambah Materi

1. Pengujian field Judul Materi.

Tipe data string sebanyak 30 karakter dengan ketentuan dapat mengandung spasi. Hasil uji coba disajikan pada Tabel 8.

Tabel 8. Hasil Uji Coba Field Judul Materi

Data uji	Skenario	Hasil	Kesimpulan
Seleksi Kondisi, Percabangan, Looping, dan Array	Jumlah karakter lebih banyak	Dapat diinputkan	Sukses
Seleksi Kondisi	Jumlah karakter lebih sedikit	Dapat diinputkan	Sukses
	Kosong	Pesan Kesalahan	Sukses

Hasil pengujian untuk field judul materi pada setiap batas nilai atas dan bawah dengan skenario jumlah karakter lebih banyak dari 30, jumlah karakter lebih sedikit dari 30, dan jumlah karakter kosong adalah berhasil.

2. Pengujian field Isi Materi

Untuk field Isi Materi tersedia dua (2) tipe data string sebanyak 100 karakter dengan ketentuan dapat mengandung spasi. Hasil uji coba disajikan pada Tabel 9.

Tabel 9. Hasil Ujicoba Isi Materi

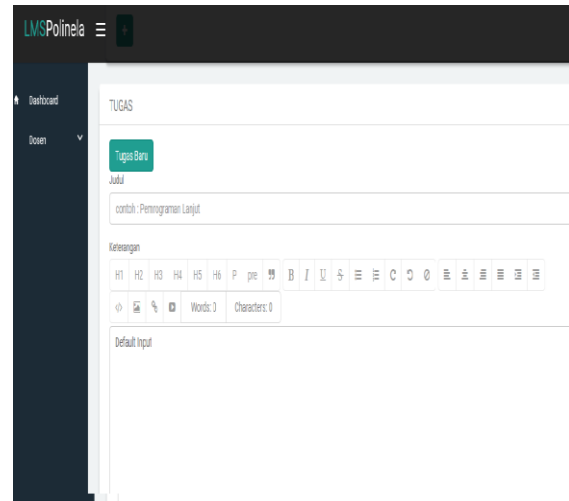
Data uji	Skenario	Hasil	Kesimpulan
C# merupakan sebuah bahasa pemrograman yang menganut paradigma Object Oriented Programming. Semua variabel dan method yang ada dalam C# harus berada dalam sebuah kelas, jadi harus	Jumlah karakter lebih banyak	Dapat diinputkan	Sukses

ada 1 buah kelas dalam sebuah script C#. Pada contoh latihan tersebut kelas bernama Program Contoh. Pada sebuah kelas ada properti dan method. Properti merupakan variabel dalam sebuah kelas sedangkan method merupakan fungsi yang ada dalam kelas. Pada contoh script diatas tidak ada properti yang dimiliki oleh kelas namun kelas ProgramContoh memiliki sebuah method yang bernama Main.			
Properti merupakan variabel dalam sebuah kelas sedangkan method merupakan fungsi yang ada dalam kelas.	Jumlah karakter lebih sedikit	Dapat diinputkan	Sukses
	Kosong	Dapat diinputkan	Sukses

Hasil pengujian untuk fungsionalitas Tambah Isi Materi, berupa artikel atau dokumen yang mengandung lebih dari 100 karakter, kurang dari 100 karakter dan kosong karakter adalah berhasil. Pada penginputan isi materi dengan data kosong karakter berhasil membentuk file Materi walaupun artikelnya kosong.

4.3 Pengujian Fungsionalitas Tambah Tugas.

Pada Gambar 4 disajikan Form Tambah Tugas untuk halaman Tugas.



Gambar 4. Form Tambah Tugas

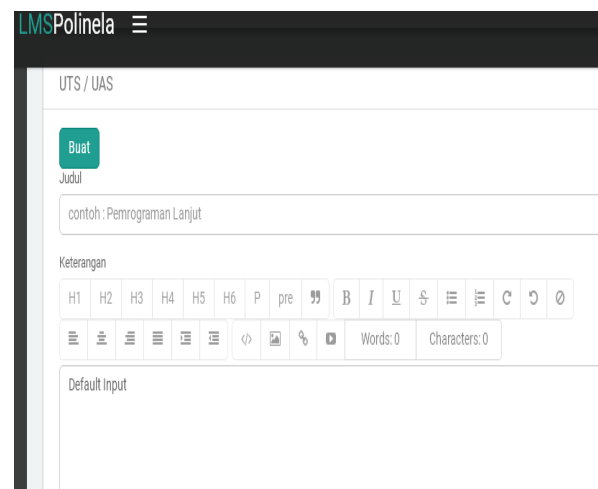
Tabel 10. Hasil Uji Coba Form Tambah Tugas

No.	Proses Uji	Hasil	Kesimpulan
1.	Klik tombol Tambah Tugas	Dapat menambah Tugas	Sukses

Hasil pengujian untuk fungsionalitas form tambah tugas adalah berhasil.

4.4 Pengujian Fungsionalitas Soal UTS dan UAS

Form Tambah Soal UTS/UAS disajikan pada Gambar 5.



Gambar 5. Form Tambah Soal UTS/UAS

Tabel 11. Hasil Uji Coba Form Soal Uts/Uas

No.	Proses Uji	Hasil	Kesimpulan
1.	Klik tombol UTS	Dapat menambah	Sukses

		UTS	
2.	Klik tombol UAS	Dapat menambah UAS	Sukses

Hasil pengujian untuk fungsionalitas tambah soal UTS dan soal UAS adalah berhasil.

V. PENUTUP

5.1. Kesimpulan

Setelah melakukan pengujian pada aplikasi LMS Polinela menggunakan teknik pengujian *blackbox testing boundary value analysis* dapat ditarik kesimpulan bahwa:

1. Teknik pengujian *blackbox testing boundary value analysis* merupakan teknik pengujian yang mudah digunakan dan diterapkan pada aplikasi LMS Polinela dengan hanya menentukan nilai batas bawah dan batas atas dari struktur data yang telah dirancang.
2. Hasil pengujian dapat diamati dan didokumentasikan dengan mudah.
3. Hasil pengujian untuk fungsionalitas tambah course, fungsionalitas tambah materi, fungsionalitas tambah tugas, dan fungsionalitas soal UTS dan UAS adalah sukses.

5.2. Saran

Penelitian dapat dilanjutkan dengan menerapkan teknik pengujian *blackbox* yang lainnya seperti: *equivalency partitioning* dan *test case*.

Daftar Pustaka

- [1] Zuriati, D.K. Widyawati, I. M. Sitanggang, A.Buono, "Analisis Kebutuhan Pengembangan E-Learning Politeknik Negeri Lampung", Seminar Nasional Pengembangan Teknologi Pertanian VI, Politeknik Negeri Lampung (SEMNAS TEKTAN VI), pp:287-295, 7-September-2017.
- [2] Zuriati, D.K. Widyawati, I.M. Sitanggang, A. Buono, "Design And Implementation Learning Management System (LMS) Polinela". The 1st International Conference on Social, Technological Innovation, Economics, and Management (ICSTIEM 2017). 27-December-2017.
- [3] M. Shi, "Software Functional Testing from the Perspective of Business Practice". Computer and Information Science, www.ccsenet.org/cis. Vol.3, No.4, November 2010.
- [4] Arochman, A. S. Darmawan, Fx. S. Santoso, "Otomatisasi Desain Test Case Pengujian Perangkat Lunak Metode Black-Box Testing Dengan Teknik Equivalence Partitioning Menggunakan Algoritma Genetika". Journal Ilmiah ICTech. Vol.9. No.2. pp 47-55, 25-Okttober-2014.
- [5] C. Berard, "Issues in the Testing of Object-Oriented

Software", 1994.

- [6] G.J Myers, "The Art of Software Testing (2nd ed.)", Hoboken, NJ: John Wiley & Sons, Inc. 2004.
- [7] W. Wibisono, F. Baskoro, "Pengujian Perangkat Lunak Dengan Menggunakan Model Behaviour UML". JUTI: Jurnal Ilmiah Teknologi Informasi Vol.1. No.1. Pp 43-50. Juli 2002
- [8] R. S. Pressman, "Software Engineering – A Practitioner’s Approach", New York: McGraw-Hill Inc. 6th edition, 2005.
- [9] O. Pastor, J.C. Molina, "Model Driven Architecture in Practice: A Software Production Environment Based on Conceptual Modeling", Springer Science & Business Media. 2007
- [10] A. S. Rosa, M. Shalahuddin, "Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek", Bandung: Informatika. 2013.
- [11] B.B. Agarwal, S.P. Tayal, M. Gupta, "Software Engineering & Testing", Sudbury, Massachusetts: Johannes and Bartlett Publishers. 2010
- [12] M.S. Mustaqbal, R. F. Firdaus, H. Rahmadi, "Pengujian Aplikasi Menggunakan Black Box Testing Boundary Value Analysis (Studi Kasus: Aplikasi Prediksi Kelulusan SNMPTN)". Jurnal Ilmiah Teknologi Informasi Terapan, vol. 1, no. 3, pp.31-36. 2015.
- [13] D. Andriansyah, "Pengujian Kotak Hitam Boundary Value Analysis Pada Sistem Informasi Manajemen Konseling Tugas Akhir". Indonesian Journal on Networking and Security - Volume 7 No 1 – 2018.
- [14] T. S . Jaya, "Pengujian Aplikasi dengan Metode Blackbox Testing Boundary Value Analysis (Studi Kasus: Kantor Digital Politeknik Negeri Lampung)", Jurnal Informatika: Jurnal Pengembangan IT, vol.3, no.1, pp 45-48. September 2017.
- [15] M. Z. Komarudin, "Pengujian Perangkat Lunak Metode Black-Box Berbasis Equivalence Partitions Pada Aplikasi Sistem Informasi Sekolah", Jurnal Mikrotik, Vol: 06 No: 03. Februari 2016
- [16] R.Meisadri, N.Indriani, "Pembangunan Game First Person Shooter 3d Alien Hunter". Jurnal Ilmiah Komputer dan Informatika (KOMPUTA) Vol. 2, No. 1, Maret 2013, ISSN : 2089-9033.2013.
- [17] A. Bhat, S.M. K. Quadri, "Equivalence Class Partitioning and Boundary Value Analysis - A review", 2nd International Conference on Computing for Sustainable Global Development (INDIACom). pp 1557-1562. 2015.
- [18] M. E. Khan, "Different Approach to Blackbox Testing Technique for Finding Error", International Journal of Software Engineering & Applications (IJSEA), Vol.2, No.4, October 2011.