

# Analisis Proses E-Learning untuk Menentukan Pola Belajar Mahasiswa Menggunakan Pendekatan *Process Mining* Studi Kasus Universitas Telkom

1<sup>st</sup> Hidayatul Aji Adika Putra

Fakultas Rekayasa Industri  
Universitas Telkom

Bandung, Indonesia

hidayatulaji@student.telkomuniversity.  
ac.id

2<sup>nd</sup> Rachmadita Andreswari

Fakultas Rekayasa Industri  
Universitas Telkom

Bandung, Indonesia

andreswari@telkomuniversity.ac.id

3<sup>rd</sup> Deden Witarsyah Jacob

Fakultas Rekayasa Industri  
Universitas Telkom

Bandung, Indonesia

dedenw@telkomuniversity.ac.id

**Abstrak**— Untuk mendukung pembelajaran jarak jauh, pembelajaran daring, dan blended learning, dibutuhkan suatu *Learning Management System (LMS)*. Dalam pengembangannya terdapat tiga komponen utama dalam LMS yang dibuat menggunakan model yaitu: pengguna, mata kuliah, dan *enrollment*. Proses pembelajaran yang tepat dapat memastikan berjalannya LMS secara normal. Untuk mengetahui pola belajar mahasiswa, dibutuhkan sebuah *event log* yang didapatkan melalui LMS. Log ini berisi informasi mengenai berbagai proses pembelajaran. Teknik *Process Mining* diterapkan untuk menganalisis pembelajaran mandiri selama perkuliahan, dengan menganalisis pembelajaran secara mandiri yang digambarkan dalam sebuah model proses, bertujuan agar dosen dapat mengetahui perkembangan mahasiswa, dan kedepannya dapat menentukan metode belajar apa yang cocok untuk mahasiswa – mahasiswa yang diajarkan oleh dosen bersangkutan. *Process Mining* juga dapat membantu dalam mengidentifikasi efektivitas pembelajaran dan kurikulum dengan cara memonitoring dan mengevaluasi kinerja Mahasiswa. Algoritma *heuristic mining* dapat memodelkan pola belajar mahasiswa pada mata kuliah Pemrograman Berorientasi Objek dan Struktur Data pada LMS dengan baik berdasarkan nilai *conformance* yang dihasilkan.

**Kata kunci**— *event log*, *process mining*, *algoritma heuristic mining*

## I. PENDAHULUAN

Melalui kementerian pendidikan dan kebudayaan (kemendikbud), pemerintah telah menerapkan kebijakan untuk melakukan aktivitas belajar dari rumah atau *Learning from Home* sebagai salah satu upaya agar mencegah penyebaran Covid-19 [1]. Untuk mendukung proses pembelajaran jarak jauh perlu adanya *platform* dalam mendukung pembelajaran tersebut, dengan demikian perlu adanya *Learning Management System (LMS)* untuk mendukung pembelajaran jarak jauh pembelajaran *daring*, dan *blended learning*. Saat ini, Telkom University menggunakan *Learning Management System* atau LMS merupakan teknologi yang memfasilitasi pembelajaran jarak jauh secara luas. LMS dapat didefinisikan sebagai platform perangkat lunak berbasis web yang menyediakan lingkungan

pembelajaran online interaktif dan mengotomatisasi administrasi, organisasi, pengiriman, dan pelaporan konten pendidikan dan hasil belajar [2].

Penelitian ini didasari karena belum adanya penelitian tentang pola belajar mahasiswa pada masing-masing kelompok *programming* saat perkuliahan online selama pandemi Covid-19. Data yang digunakan adalah *event log* pada mata kuliah Pemrograman Berorientasi Objek dan Struktur Data. Pada penelitian ini mencoba menunjukkan apakah terdapat korelasi antara tingkat aktivitas mahasiswa dan nilai akhir yang didapatkan, untuk itu metode *process mining* dapat digunakan untuk memproses data *event log* yang didapatkan dari LMS. *Process Mining* adalah *event log* yang dihasilkan dalam sistem informasi. Setiap catatan *log* terkait dengan peristiwa dan setiap peristiwa terkait dengan aktivitas yang dilakukan sedemikian rupa sehingga *event log* harus menyajikan hubungan berurutan antara peristiwa [3].

Peningkatan data merupakan tantangan yang harus dihadapi semua institusi. *Event Log* adalah catatan historis yang berisi serangkaian aktivitas pengguna yang direkam pada sistem [4]. Untuk itu peneliti menggunakan teknik *process mining* untuk melakukan pengolahan data dan menggunakan *event log* dari sistem manajemen pembelajaran Telkom University untuk menemukan pola belajar mahasiswa dalam penggunaan *e-learning*. Pada penelitian ini salah satu metode yang dapat diadopsi adalah *process mining*, yaitu menganalisis data *event log* berbasis proses. Analisis data berbasis proses ini bertujuan untuk memodelkan proses yang terjadi dan terekam dalam LMS, mengecek kesesuaian pelaksanaan proses dengan prosedur, dan mengusulkan pengembangan proses di masa mendatang [5].

Dalam penelitian ini akan dilakukan *process mining* dengan menerapkan algoritma *heuristic mining*. *Heuristic mining* dapat menangani *noise*, dan dapat digunakan untuk mengekspresikan perilaku utama (yaitu tidak semua detail dan pengecualian) yang terdaftar dalam *event log* [6]. *Heuristics miner* menggali perspektif *control-flow* dari model proses [6]. Menggunakan teknik pemodelan *heuristic mining* berguna untuk membuat model proses dengan menggunakan aplikasi Petri Net, untuk melakukan analisis ini memerlukan

tools process mining yaitu menggunakan tools Disco dan ProM untuk menemukan model proses dan mengevaluasinya.

## II. KAJIAN TEORI

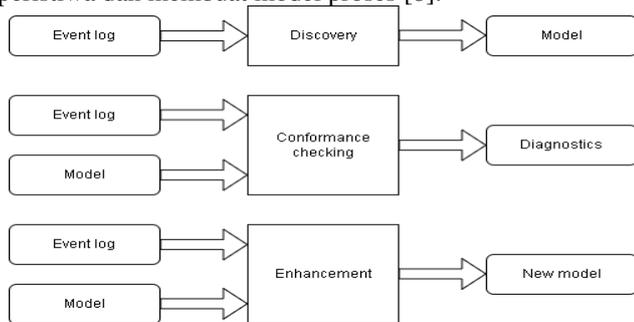
### A. Learning Management System

Learning Management System LMS adalah sistem aplikasi kooperatif yang didukung komputer, yang mendukung pembelajaran kolaboratif di antara peserta didik dari tempat yang berbeda [7]. LMS juga menyediakan tempat kerja kolaboratif bagi mahasiswa untuk bekerja satu sama lain [7].

Learning Management System (LMS) adalah perangkat lunak yang digunakan untuk mengelola satu atau lebih kursus kepada satu atau lebih mahasiswa, LMS biasanya merupakan sistem berbasis web yang memungkinkan mahasiswa untuk meng-otentikasi diri mereka sendiri, mendaftar untuk kursus, menyelesaikan kursus dan mengambil penilaian [3].

### B. Process Mining

Process mining adalah istilah yang biasa digunakan untuk algoritma yang mampu mengekstrak pengetahuan dari log peristiwa dan membuat model proses [8].



GAMBAR 1  
PROCESS MINING

Berdasarkan gambar 1 *Process mining* adalah sebuah pendekatan manajemen proses untuk analisis proses bisnis berdasarkan *event log*. *Event log* dihasilkan secara otomatis dalam sistem informasi berdasarkan akses pengguna secara *real-time*, sehingga *event log* dapat dikatakan menggambarkan pelaksanaan proses yang sebenarnya dalam organisasi. Data log peristiwa dianalisis menggunakan penambangan proses untuk mengidentifikasi pola dan tren selama eksekusi proses [9]. *Process mining* dapat menjelaskan bagaimana proses bisnis sebelumnya berdampak terhadap proses bisnis di masa depan, kapan dan mengapa proses bisnis diimplementasikan berbeda dari prosedur, serta bagaimana mengontrol proses operasi, dan bagaimana mendesain ulang proses tersebut untuk meningkatkan kinerjanya [10]. Berikut adalah teknik *process mining* [11], diantaranya.

#### 1. Process Discovery

*Process discovery* merupakan proses menemukan model proses dari *event log* [11]. *Process discovery* harus memberikan model dengan kualitas, di mana diharapkan dapat dipahami dengan menghindari kompleksitas yang tidak perlu, dan memberikan akurasi yang dapat

diterima, menyeimbangkan *recall*, presisi, dan generalisasi [8].

#### 2. Conformance Checking

Teknik *conformance checking* menyediakan cara untuk membandingkan perilaku dalam log peristiwa dengan perilaku yang dijelaskan oleh model proses, yang dapat ditemukan atau dibuat secara manual [11]

#### 3. Model Enhancement

*Model enhancement* memperoleh informasi tambahan seperti data kinerja menggunakan model proses dan log peristiwa yang selaras [12].

### C. Petri net

Petri net adalah grafik yang terdiri dari tempat-tempat yang dapat berisi token, keberadaan token menentukan keadaan *net*, serta transisi yang mengubah keadaan *net* dengan mengkonsumsi dan memproduksi token dari tempat ke tempat yang terhubung dan memancarkan aktivitas terkait dengan *firing* [11]. Dalam *workflow (Petri) net* ada satu tempat tanpa *incoming transition (source)* dan satu tempat *outgoing transition (sink)* dan setiap tempat dan transisi berada di jalur yang diarahkan antara dua tempat ini [11]. Pada *workflow net* semua transisinya dapat diaktifkan dan setiap status dapat dijangkau, status akhir dapat dicapai [11].

### D. Perbandingan Algoritma Process Mining

Dalam *process mining* terdapat beberapa algoritma yang digunakan seperti *Alpha Miner*, *Heuristic Miner*, *Fuzzy Miner*, *Inductive Visual Miner* dan *Genetic Miner*, setiap algoritma memiliki karakteristik masing-masing. Dengan demikian perlu mengetahui perbedaan dari masing-masing algoritma. Berikut merupakan penjelasan dari perbandingan algoritma *process mining*.

TABEL 1  
PERBANDINGAN ALGORITMA POCESS MINING

Algorithm	Primary Problem	Predictors	Power	Raw Implementation	Interpretability	Regression also	Normalization
Alpha Miner	Multiclass or Binary	Numeric or Categorical	Medium	Medium	Medium	No	Usually
Heuristic Miner	Binary	Numeric or Categorical	High	Easy	Good	Yes	No
Fuzzy Miner	Binary	Numeric or Categorical	High	Easy	Good	Yes	No
Inductive Visual Miner	Binary	Numeric or Categorical	High	Medium	Good	Yes	Usually
Genetic Miner	Binary	Numeric or Categorical	High	Easy	Good	Yes	No

#### E. Heuristic Miner

*Heuristics Miner* adalah algoritma penambangan praktis yang dapat menangani *noise*, dan dapat digunakan untuk mengekspresikan perilaku utama (yaitu tidak semua *detail* dan pengecualian) yang terdaftar dalam *event log* [6]. *Heuristic miner* dapat digunakan untuk menentukan proses utama dari ribuan log dan mendeteksi perilaku yang tidak umum dalam suatu proses [13].

F. Event Log

Event log adalah kumpulan jejak, masing-masing terdiri dari langkah-langkah proses yang akan dieksekusi [11]. Langkah-langkah tersebut sebagai aktivitas dan pelaksanaannya dalam jejak sebagai instance aktivitas, setiap instance aktivitas mungkin melintasi *lifecycle* tertentu [11].

G. Tools Disco dan ProM

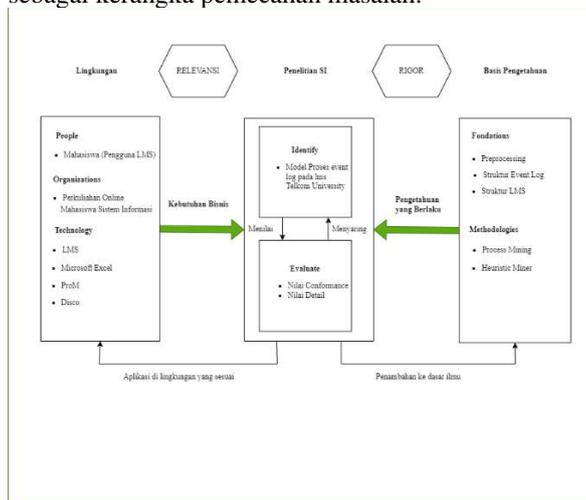
Berikut merupakan *tools* yang digunakan dalam penelitian.

- 1) *Disco*  
Disco merupakan toolkit *process mining* lengkap dari Fluxion yang membuat *process mining* cepat dan mudah [14].
- 2) *ProM*  
ProM merupakan alat mendukung penemuan proses, pemeriksaan kesesuaian, analisis jaringan sosial, penambangan organisasi, penambangan keputusan, dan prediksi [15].

III. METODE

A. Model Konseptual

Dalam pembuatan model konseptual mengimplementasikan kerangka *design science research* sebagai kerangka pemecahan masalah.



GAMBAR 2 MODEL KONSEPTUAL

Berdasarkan gambar 2 menjelaskan bahwa lingkungan mendefinisikan ruang masalah. Untuk penelitian SI, terdiri dari orang, organisasi, dan teknologi yang ada atau yang direncanakan. Di dalamnya terdapat tujuan, tugas, masalah, dan peluang yang mendefinisikan kebutuhan bisnis sebagaimana adanya dirasakan oleh orang-orang dalam organisasi.

B. Pengumpulan Data

Pada penelitian ini menggunakan data sekunder sebagai bahan untuk analisis dan perancangan artifak. Data primer mencakup semua dokumen atau laporan yang dihasilkan dalam aktivitas akademik di studi kasus dimintakan sampling berupa *softcopy* sebagai acuan dalam perancangan data. Peneliti menggunakan metode deskriptif dengan mengumpulkan data-data yang dibutuhkan pada penelitian ini dari objek penelitian yaitu Telkom University.

TABEL 2 PENGAMBILAN DATA

No	Nama Mata Kuliah	Program Studi	Jumlah Log (rows)
1	Pemrograman Berorientasi Objek	S1 Sistem Informasi	113.693
2	Struktur Data	S1 Sistem Informasi	231.620

Berdasarkan tabel 2 *event log* mata kuliah tersebut merupakan data mentah yang akan diolah sehingga mendapatkan data yang dilihat berdasarkan waktu, individu dan aktivitas yang dilakukan oleh pengguna *e-learning* Telkom University.

C. Pengolahan Data

Data yang sudah diperoleh seperti *event log* berguna untuk melihat perbedaan pola mahasiswa terhadap penggunaan *e-learning* Telkom University pada dua mata kuliah berbeda yaitu Pemrograman Berorientasi Objek dan Struktur Data S1 Sistem Informasi, *event log* tersebut akan dilakukan proses penyesuaian kebutuhan data. *Event log* mata kuliah tersebut masih merupakan data mentah, data tersebut akan diolah sehingga mendapatkan data yang dapat dilihat berdasarkan waktu, aktor, dan aktivitas yang telah dilakukan oleh pengguna terhadap *e-learning* Telkom University. Perancangan perangkat lunak menggunakan *tools* Disco dan ProM dengan menggunakan algoritma *heuristic miner*.

1. Preprocessing Data

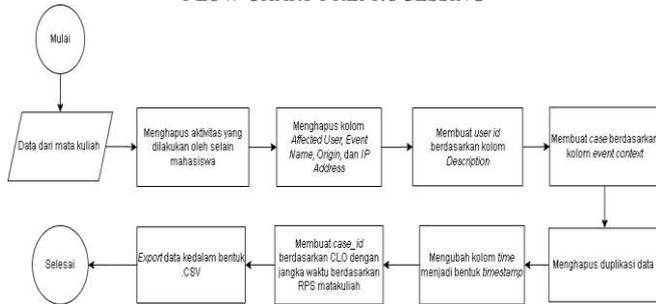
*Preprocessing* pada *event log* mata kuliah Pemrograman Berorientasi Objek dan Struktur Daya, hal yang dilakukan pertama pada proses data cleansing dengan menggunakan MS.Excel, dimana atribut yang tidak digunakan sebagai input dari process mining dibuang.

TABEL 3 CONTOH EVENT LOG

No	Kolom atribut data mentah	Contoh dari Mata Kuliah Pemrograman Berorientasi Objek	Contoh dari Mata Kuliah Struktur Data
1	<i>Timestamp</i>	2020/09/30, 14:28	2020/09/20, 21:47
2	<i>User</i>	DARRYL BASTIAN	SYIFANADA SAFIA DJAUHARI
3	<i>Case</i>	<i>View course</i>	<i>Collect Assignment</i>
4	<i>Component</i>	<i>System</i>	<i>Assignment</i>
5	<i>User_ID</i>	18106	21843
6	<i>CLO</i>	CLO1	CLO1
7	<i>Case_ID</i>	1	5

Berdasarkan tabel 3 merupakan contoh atribut data mata kuliah Pemrograman Berorientasi Objek dan Struktur Data. Berikut merupakan *flowchart* dari *preprocessing*, seperti pada gambar IV.1

TABEL 4  
FLOW CHART PREPROCESSING



Berdasarkan gambar 4 pada tahap *preprocessing*. *preprocessing* pada penelitian pada penelitian dilakukan dengan bantuan aplikasi MS.Excel. *Case attribute* yang telah ditentukan seperti pada Tabel 5.

TABEL 5  
EVENT CONTEXT ATTRIBUTE

No	Case Attribute	Keterangan
1.	View Course	Diakses oleh mahasiswa untuk mengikuti perkuliahan
2.	Collect Assignment	Berisi tugas-tugas yang diberikan oleh dosen dan digunakan untuk mengumpulkan tugas
3.	Label	Sumber pembelajaran eksternal
4.	File	Materi pembelajaran dari dosen
5.	View URL	Link eksternal seperti website, youtube dan lain-lain
6.	Attempt Quiz	Pembelajaran aktif yang disediakan oleh dosen untuk menguji pemahaman mahasiswa
7.	Open Folder	Membuka folder
8.	Fill in Forum	Digunakan untuk diskusi antar mahasiswa dan dosen
9.	Interactive Content	Berisi video pembelajaran dari dosen
10.	Other	Aktivitas lainnya
11.	Access lesson	Mengakses materi pembelajaran

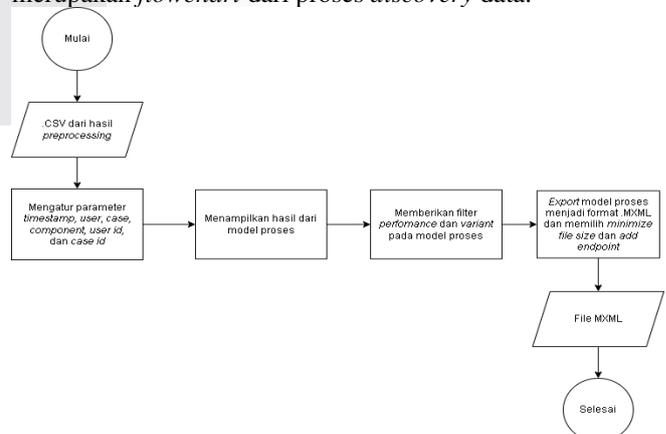
Setelah melakukan *cleansing data*, selanjutnya mengubah format *timestamp* menjadi DD/ MM/ YY untuk tanggal dan HH : MM untuk waktu, *Case ID* ditentukan berdasarkan *timestamp* dan user yang mengakses LMS Telkom University, dengan menggunakan asumsi akses dengan *delay* akses yang dikelompokkan berdasarkan *Course Learning Outcome* (CLO) yang sesuai dengan Rencana Pembelajaran Semester (RPS) dari mata kuliah Pemrograman Berorientasi Objek dan Struktur Data. Kemudian *event log* dalam bentuk hasil CSV dari *preprocessing* dengan menggunakan MS.Excel yang digunakan sebagai *input* dari proses *discovery* dengan menggunakan bantuan *tools* yaitu Disco. Berikut merupakan data mentah pada penelitian ini.

Tabel 6  
(Data Mentah)

NO	Timestamp	User ID	User	Case	Case ID	Component	CLO
1.	2020/09/30 14:28	18106	DARRYL BASTIAN	View Course	1	System	1
2.	2020/10/02 23:11	21843	SYFANADA SAFIA DJAUHARI	View Course	4	System	1
3.	2020/09/28 14:12	21847	EDBERT SALIM	Collect Assignment	7	Assignment	1
4.	2020/10/03 15:31	21853	AHMAD DHAFIN FAZA	View Course	10	System	1
5.	2020/10/04 11:07	21854	MOHAMMAD WASIK AMIN	View Course	13	System	1
6.	2020/09/30 16:19	21862	ILHAM AULIYA RAHMAN	Collect Assignment	16	Assignment	1
7.	2020/10/03 00:11	21864	BAIQ VIDISASCA SUHITA	View Course	19	System	1
8.	2020/10/04 22:36	21867	NETA AULYA KURNIA NINGRUM	View Course	22	System	1
9.	2020/10/02 10:24	21876	KANZA JILAN FATIN	View Course	25	System	1
10.	2020/10/03 20:58	21881	MUHAMMAD DAFFI DAULAY	View Course	28	System	1
11.	2020/10/03 23:11	21887	GILDAS AGATHON CATO	View Course	31	System	1
12.	2020/10/01 15:52	21888	SOYA KARERRA	View Course	34	System	1
13.	2020/10/03 11:11	21891	AINI RAHMADANA	View Course	37	System	1
14.	2020/10/03 16:09	21894	MEGA FITRI YANI	View Course	40	System	1
15.	2020/10/04 15:51	21896	FIRENZE SUKMANING DIEFTA	Fill in Forum	43	System	1
16.	2020/10/03 20:11	21898	RAMIZ QUDAMAH	Collect Assignment	46	Assignment	1
17.	2020/10/04 23:11	21899	MUHAMMAD RISQAN ADITAMA	View Course	49	System	1
18.	2020/10/03 20:48	21901	TIARA HATI GIWANGKARA	Collect Assignment	52	Assignment	1
19.	2020/09/30 07:45	21928	AULIA RAMADHANI JOHANIS PARANSA	Collect Assignment	55	Assignment	1
20.	2020/09/28 13:01	21938	ANANDA VIAMIANNI	View Course	58	System	1

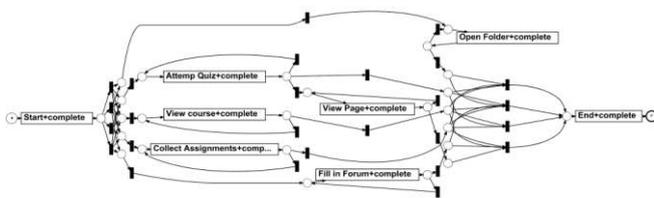
2. Discovery Data

Pada tahap *discovery data*, dilakukan pemrosesan terhadap *event log* untuk menghasilkan *file* MXML. Dataset yang telah melewati *preprocessing* akan dimasukan ke dalam *tools* Disco untuk mencari pemodelan. Seperti pada gambar 3 merupakan *flowchart* dari proses *discovery data*:



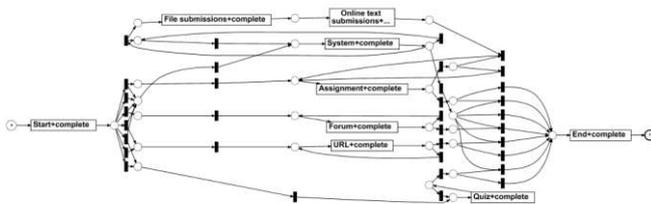
GAMBAR 3  
FLOWCHART PROCESS DISCOVERY





GAMBAR 9

PETRI NET PADA MODEL HEURISTIC MINER PEMROGRAMAN BERORIENTASI OBJEK TANPA SKENARIO



GAMBAR 10

PETRI NET PADA MODEL HEURISTIC STRUKTUR DATA TANPA SKENARIO

Pada Gambar 9 dan Gambar 10 terdapat banyak panah yang menghubungkan antar aktivitas yang terjadi pada event log. Sehingga sulit untuk melakukan analisis lebih lanjut terhadap kecocokan model proses dengan event log. Oleh karena itu, dibutuhkan penerapan skenario untuk melakukan analisis terhadap event log pada kedua mata kuliah, dengan tidak mengurangi nilai-nilai representatif dari kedua model proses.

#### IV. HASIL DAN PEMBAHASAN

##### A. Process Mining dengan Skenario Pengujian

Tujuan dilakukannya pengujian skenario pada tugas akhir ini adalah:

1. Mengetahui pengaruh parameter *heuristic miner*, *dependency threshold*, *relative to best threshold*, dan *positive observation threshold* pada model proses dan mendapatkan model proses terbaik berdasarkan hasil pengujian.
2. Mengetahui performa dari algoritma *heuristic miner* dalam melakukan pemodelan model proses dari *event log* LMS Universitas Telkom.
3. Melakukan pengujian dan analisis performansi untuk menemukan pola belajar mahasiswa.

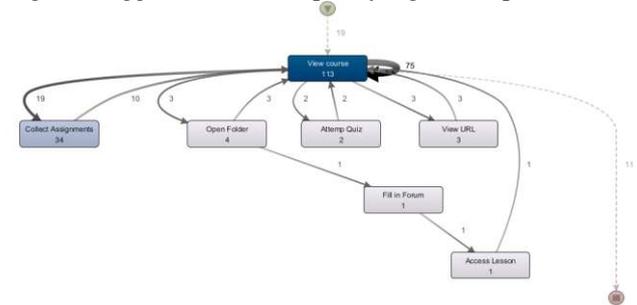
##### a. Skenario Satu

Pada pengujian yang dilakukan skenario ini dilakukan pengujian terhadap dua mata kuliah yang memiliki *event log* setelah *preprocessing* dengan jumlah 113.693 pada mata kuliah Pemrograman Berorientasi Objek dan 231.620 pada mata kuliah Struktur Data. *Event log* dengan jumlah tersebut merupakan *event log* yang tidak bebas dari *low case-variants* atau variasi aktivitas dengan jumlah *case* yang rendah, oleh karena itu, pada skenario ini peneliti menerapkan *filter variant* setidaknya tiga *cases*. Banyaknya jumlah variasi dapat menyebabkan jalur antar *case* dan panah yang tergambar pada model proses menjadi sangat banyak sehingga menghambat penemuan model proses utama pada *event log* tersebut. *Event* yang terjadi dibatasi menjadi minimal dua *event* setiap kali *user* melakukan akses, artinya aktivitas *user* dibatasi menjadi minimal dua *event* pada pengujian skenario satu, seperti pada tabel 7.

TABEL 7  
KRITERIA FILTER PADA SKENARIO

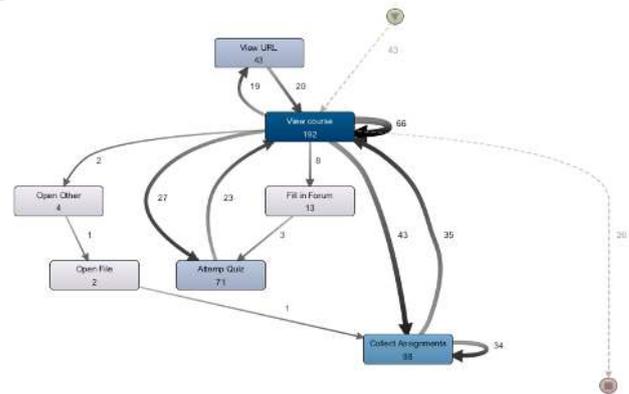
Filter	Keterangan	Keterangan
<i>Variation</i>	Setidaknya memiliki 3 <i>cases</i>	Bertujuan untuk mendapatkan pola generik dari model proses sebagai representatif.
<i>Performance</i>	Setidaknya memiliki 2 <i>event</i>	Agar adanya interaksi yang dilakukan oleh <i>user</i> pada sistem.

Pada Gambar 11 dan 12 merupakan *initial Process model* dengan menggunakan filter seperti yang tertera pada Tabel 7



GAMBAR 11

INITIAL PROCESS MODEL PADA MATA KULIAH PEMROGRAMAN BERORIENTASI OBJEK DENGAN SKENARIO



GAMBAR 12

INITIAL PROCESS MODEL PADA MATA KULIAH STRUKTUR DATA DENGAN SKENARIO

Model proses tersebut merupakan hasil identifikasi dengan menggunakan aplikasi Disco, yang selanjutnya output dengan format MXML tersebut akan digunakan untuk analisis lebih lanjut menggunakan aplikasi ProM dengan berbagai kombinasi dari parameter *dependency threshold*, *relative to-the-best threshold*, dan *positive observation threshold*. Kemudian hasil dari kombinasi tersebut ditampilkan dalam tabel dan diurutkan dalam bentuk analisis pengaruh ketiga parameter tersebut model proses yang dihasilkan.

##### b. Skenario Dua

Pada skenario ini model proses terbaik yang didapatkan dari pengujian skenario satu akan dianalisis menggunakan *Conformance analysis with Petri Net*. Hasil berupa nilai *fitness*, *precision*, dan *structure* yang diuraikan melalui analisis lebih lanjut.

##### c. Skenario Tiga

Pada skenario ini model proses terbaik yang didapatkan dari pengujian skenario satu akan dianalisis menggunakan *performance analysis with Petri Net* untuk mengidentifikasi adanya *bottleneck* dan juga analisis dari sudut pandang performansi lainnya untuk mengetahui pola belajar mahasiswa

B. Hasil Pengujian dan Analisis

Pada pengujian skenario satu, *event log* yang diproses dikelompokkan menjadi tiga tabel yang menunjukkan parameter yang berbeda. *Relative-to-best Threshold* (RT), *Positive observations Threshold* (PT), dan *Dependency Threshold* (DT). Analisis perbandingan *fitness* terhadap parameter *Relative-to-best Threshold* (RT), *Positive observations Threshold* (PT), dan *Dependency Threshold* (DT) akan dilakukan dengan menerapkan algoritma *Heuristic Miner* menggunakan tools ProM 5.2 dan *Conformance Checker* untuk mendapatkan nilai *fitness*. Nilai tersebut diuji dengan perubahan yang diberikan kepada RT, PT, dan DT.

1. Pengujian dan Analisis Mata Kuliah Pemrograman Berorientasi Objek

Pada tahap ini dilakukan pengujian dan analisis terhadap mata kuliah Pemrograman Berorientasi Objek untuk mengetahui pengaruh parameter heuristic miner, yaitu *Dependency Threshold* (DT), *Relative-to-best Threshold* (RT), dan *Positive observation Threshold* (PT) pada model proses dan mendapatkan model proses terbaik berdasarkan hasil pengujian.

a. Analisis Terhadap Parameter *Relative-to-best Threshold* Mata Kuliah Pemrograman Berorientasi Objek

Berdasarkan Berdasarkan hasil pengujian terhadap parameter *Relative-to-best Threshold* (RT) yang ditunjukkan pada Tabel 8, dapat dilihat bahwa perubahan terhadap nilai parameter tersebut tidak memiliki pengaruh terhadap nilai *fitness*. Nilai *Relative-to-best Threshold* (RT) 0.05 memiliki nilai *fitness* yang sama dengan nilai *Relative-to-best Threshold* (RT) 1.0. Begitu juga dengan nilai *fitness* yang sama dengan nilai *Relative-to-best Threshold* (RT) 0.25, 0.50, dan 0.75 memiliki nilai *fitness* yang sama, yaitu 0.357.

TABEL 8

PENGUJIAN PADA NILAI RELATIVE-TO-BEST THRESHOLD (RT)

Perubahan pada Nilai <i>Relative-to-best Threshold</i> (RT) mata kuliah Pemrograman Berorientasi Objek			
RT	PT	DT	<i>Fitness</i>
0.05	50	0.9	0.357
0.25	50	0.9	0.357
0.50	50	0.9	0.357
0.75	50	0.9	0.357
1.00	50	0.9	0.357

b. Analisis Terhadap Parameter *Relative-to-best Threshold* Mata Kuliah Pemrograman Berorientasi Objek

Perubahan parameter *Dependency Threshold* (DT) pada hasil pengujian seperti yang ditunjukkan pada Tabel 9 di bawah menghasilkan model proses yang tidak terlalu signifikan, sama seperti sebelumnya pada pengujian pada parameter *Relative-to-best Threshold* (RT). Nilai *fitness* yang dihasilkan juga serupa ketika melakukan pengujian terhadap nilai parameter *Dependency Threshold* (DT) yang memiliki nilai *fitness* 0.357.

TABEL 9

PENGUJIAN PADA NILAI DEPENDENCY THRESHOLD (DT)

Perubahan pada Nilai <i>Dependency Threshold</i> (DT) mata kuliah Pemrograman Berorientasi Objek			
RT	PT	DT	<i>Fitness</i>
0.05	50	0.9	0.357
0.05	50	0.7	0.357
0.05	50	0.5	0.357
0.05	50	0.3	0.357
0.05	50	0.1	0.357

c. Analisis Terhadap Parameter *Relative-to-best Threshold* Mata Kuliah Pemrograman Berorientasi Objek

Berdasarkan hasil pengujian seperti ditunjukkan pada Tabel 10, nilai parameter *Positive Observation Threshold* (PT) yang diubah memiliki pengaruh terhadap nilai *fitness*. Sehingga nilai *fitness* yang mendekati 1 adalah dengan parameter *Positive Observation Threshold* (PT) bernilai 50. Dengan nilai *fitness* terbesar yaitu 0.357. Nilai default pada parameter *Positive Observation Threshold* (PT) adalah 10. menyebabkan seluruh aktivitas yang memiliki frekuensi rendah dilibatkan dalam model proses. Selain itu nilai *fitness* yang dihasilkan menjadi rendah. Semakin kecil nilai dari *threshold*, maka semakin banyak *low-frequent activity* yang ikut serta pada model proses. Pada nilai RT = 0.05, PT = 50, dan DT = 0.9 memiliki hasil *fitness* yang mendekati 1. dengan nilai *fitness* 0.357

TABEL 10

PENGUJIAN PADA NILAI POSITIVE THRESHOLD (PT)

Perubahan pada Nilai <i>Positive Observation Threshold</i> (PT) mata kuliah Pemrograman Berorientasi Objek			
RT	PT	DT	<i>Fitness</i>
0.05	10	0.9	0.307
0.05	15	0.9	0.338
0.05	20	0.9	0.357
0.05	25	0.9	0.357
0.05	50	0.9	0.357

2. Pengujian dan Analisis pada Mata Kuliah Struktur Data

a. Analisis Terhadap Parameter *Relative-to-best Threshold* Mata kuliah Struktur Data

Berdasarkan hasil pengujian terhadap parameter *Relative-to-best Threshold* (RT) 0.05 memiliki nilai *fitness*. Nilai *Relative-to-best Threshold* (RT) 1.00. Begitu juga dengan nilai *Relative-to-best Threshold* (RT) 0.25, 0.50, dan 0.75 memiliki nilai *fitness* yang sama yaitu 0.753.

Perubahan pada Nilai <i>Relative-to-best Threshold</i> (RT) mata kuliah Struktur Data			
RT	PT	DT	<i>Fitness</i>
0.05	50	0.9	0.753
0.25	50	0.9	0.753
0.50	50	0.9	0.753
0.75	50	0.9	0.753
1.00	50	0.9	0.753

b. Analisis Terhadap Parameter *Relative-to-best Threshold* Mata kuliah Struktur Data

Perubahan parameter *Dependency threshold* (DT) pada hasil pengujian seperti yang ditunjukkan pada Tabel 11 menghasilkan model proses yang tidak berubah, sama seperti sebelumnya pada pengujian parameter *Relative-to-best Threshold* (RT). Pada nilai *fitness* yang dihasilkan juga sama serupa ketika dilakukan pengujian terhadap parameter *Dependency Threshold* (DT). Nilai *fitness* tidak terpengaruh terhadap nilai parameter *Dependency Threshold* (DT).

TABEL 11

HASIL PENGUJIAN TERHADAP PARAMETER DEPENDENCY THRESHOLD (DT)

Perubahan pada Nilai <i>Dependency Threshold</i> (DT) mata kuliah Struktur Data			
RT	PT	DT	<i>Fitness</i>
0.05	50	0.1	0.753

0.05	50	0.3	0.753
0.05	50	0.5	0.753
0.05	50	0.7	0.753
0.05	50	0.9	0.753

c. Analisis Terhadap Parameter *Positive Observation Threshold* Mata kuliah Struktur Data

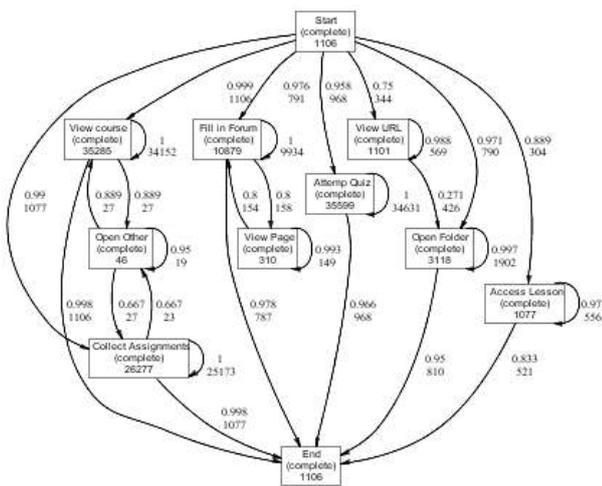
Berdasarkan hasil pengujian seperti yang di tunjukan pada Tabel 12, nilai parameter *Positive Observation Threshold* (PT) yang diubah mempunyai pengaruh terhadap nilai *fitness*. Sehingga nilai *fitness* yang mendekati 1 adalah parameter *Positive Observation Threshold* (PT) bernilai 15. Dengan nilai *fitness* terbesar yaitu 0.753. Nilai *default* pada parameter *positive Observation Threshold* (PT) adalah 10, mempengaruhi semua aktivitas yang memiliki frekuensi rendah dilibatkan dalam model proses. Selain itu nilai *fitness* yang dihasilkan menjadi rendah. Semakin kecil nilai dari *threshold*, maka semakin banyak *low frequent activity* yang ikut serta pada model proses. Pada nilai RT = 0.05, PT = 15, dan DT = 0.9 memiliki hasil *fitness* yang mendekati 1, dengan nilai *fitness* 0.753

TABEL 12  
HASIL PENGUJIAN TERHADAP PARAMETER POSITIVE OBSERVATION THRESHOLD (PT)

Perubahan pada Nilai <i>Positive Threshold</i> (PT) mata kuliah Struktur Data			
RT	PT	DT	<i>Fitness</i>
0.05	10	0.9	0.612
0.05	15	0.9	0.509
0.05	20	0.9	0.301
0.05	25	0.9	0.372
0.05	50	0.9	0.753

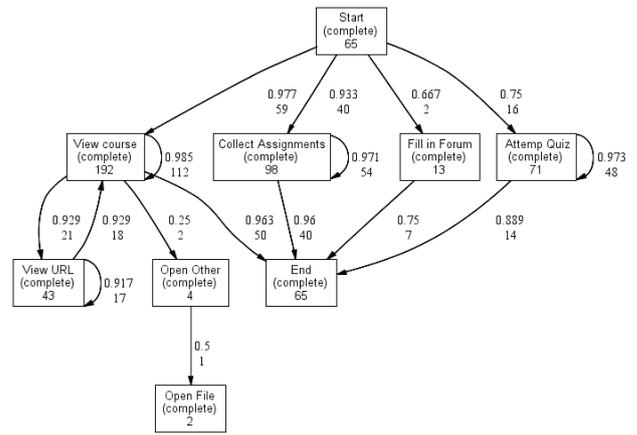
3. Hasil Pengujian dan Analisis Skenario Dua

Pada pengujian skenario dua, model proses yang digunakan adalah seperti pada Gambar 13 dan 14.



GAMBAR 13

Model Proses Hasil Pengujian Skenario pada Mata Kuliah Pemrograman Berorientasi Objek



GAMBAR 14

MODEL PROSES HASIL PENGUJIAN SKENARIO PADA MATA KULIAH STRUKTUR DATA

Model proses dari *process mining* ini merupakan hasil dari algoritma *heuristic miner*, menggunakan setelan parameter yang digunakan untuk mata kuliah Pemrograman Berorientasi Objek, *Relative-to-best Threshold* (RT) 0.05, *Positive Observations Threshold* (PT) bernilai 50 dan nilai *Dependency Threshold* (DT) adalah 0.9. Sedangkan untuk mata kuliah Struktur Data, *Relative-to-best Threshold* (RT) 0.05, *Positive Observations Threshold* (PT) bernilai 50 dan nilai *Dependency Threshold* (DT) adalah 0.9. Model proses tersebut merupakan model proses terbaik yang dihasilkan dari pengujian skenario satu.

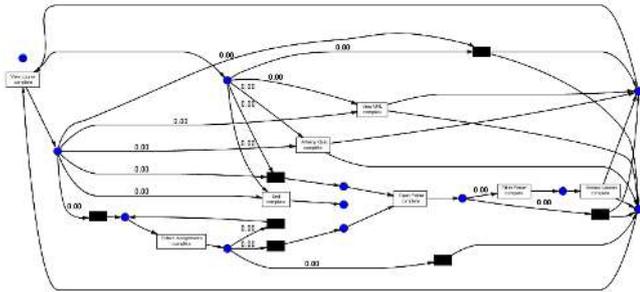
Hasil *Conformance Checker* terdiri dari tiga bagian yaitu *fitness*, *precision*, dan *structure*. Seperti pada Tabel 13 berikut.

TABEL 13  
HASIL CONFORMANCE CHECKING

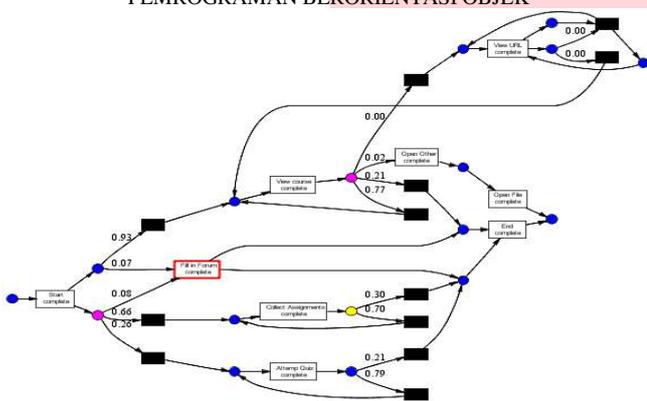
Nilai	Mata Kuliah Pemrograman Berorientasi Objek	Mata Kuliah Struktur Data
<i>Fitness</i>	0.726	0.915
<i>Advanced Behavioral Appropriateness</i>	0.589	0.771
<i>Degree of Model Flexibility</i>	0.374	0.416
<i>Structural</i>	1	1

Pada bagian *fitness*, mata kuliah Pemrograman Berorientasi Objek menampilkan nilai 0.726 Sedangkan, mata kuliah Struktur Data menampilkan nilai 0.915. Hal ini menunjukkan bahwa proses model dapat memodelkan *event log* dengan cukup baik, karena nilai sempurna yang dihasilkan jika keseluruhan *log di replay* secara benar dan sempurna adalah 1. Nilai *precision* pada mata kuliah Pemrograman Berorientasi Objek menampilkan nilai 0.589 pada *advanced behavioral appropriateness*, dan 0.374 pada *degree of model flexibility*. Sedangkan nilai *precision* pada mata kuliah Struktur Data menampilkan nilai 0.771 pada *advanced behavioral appropriateness*, dan 0.416 pada *degree of model flexibility*. Hal ini menunjukkan bahwa model proses yang kurang fleksibel sehingga memungkinkan hanya sedikit variasi task yang diperbolehkan pada log. Tampak pada model proses yang memiliki cukup banyak jalur untuk merepresentasikan urutan task yang berlangsung sesuai dengan *event log*. Nilai struktur yang dihasilkan dari *advanced appropriateness* adalah 1.0 pada kedua mata kuliah dalam penelitian ini. Nilai ini menunjukkan bahwa tidak ada duplikasi task pada model proses di kedua mata kuliah serta tidak ada task yang berulang.

4. Hasil Pengujian dan Analisis Skenario Tiga  
 Pada pengujian skenario tiga, model proses berupa petri net dianalisis menggunakan *plugin performance analysis with petri net*. Seperti pada Gambar 15 dan Gambar 16 di bawah



GAMBAR 15  
 HASIL PERFORMANCE ANALYSIS MATA KULIAH PEMROGRAMAN BERORIENTASI OBJEK



GAMBAR 16  
 HASIL PERFORMANCE ANALYSIS MATA KULIAH STRUKTUR DATA

Seperti yang ditunjukkan pada Gambar 15 dan 16 di atas adalah analisis dari *time perspective* berupa detail waktu antar proses dimana token yang menunjukkan *waiting time* rendah berwarna biru, *waiting time* sedang berwarna kuning, dan *waiting time* tinggi berwarna ungu. *Waiting time* tinggi menunjukkan bahwa pada titik tersebut terjadi *event log* atau pemampatan jalannya proses. Proses identifikasi *event log* antar kedua mata kuliah digambarkan pada Tabel 14 dan Tabel 15.

TABEL 14  
 IDENTIFIKASI EVENT LOG ANTAR PROSES MATA KULIAH PEMROGRAMAN BERORIENTASI OBJEK

Relation\Waiting Time (Hours)	Average	Min.	Max.
Quiz - End	87.33	0.0	200.10
Assignment - End	62.56	0.0	412.09
Forum - End	0.0	0.0	0.0
View Course - End	119.00	0.0	771.00
View URL - End	0.0	0.0	0.0
Open File - End	0.0	0.0	0.0
Open Other - End	0.0	0.0	0.0

TABEL 15  
 IDENTIFIKASI EVENT LOG ANTAR PROSES MATA KULIAH STRUKTUR DATA

Relation\Waiting Time (Hours)	Average	Min.	Max.
Quiz - End	90.75	0.0	310.13
Assignment - End	0.0	0.0	0.0
Forum - End	0.0	0.0	0.0
View Course - End	108.37	0.0	646.37
View URL - End	0.0	0.0	0.0

Open File - End	0.0	0.0	0.0
Open Other - End	0.0	0.0	0.0

Penyebab terjadinya *waiting time* yang tinggi antar proses seperti pada mata kuliah Pemrograman Berorientasi Objek dan mata kuliah Struktur Data adalah dikarenakan frekuensi *user* yang berbeda - beda dalam melakukan akses. Contohnya seperti mahasiswa yang mengakses *quiz* diluar waktu jam perkuliahan yang menyebabkan mahasiswa memiliki jarak waktu dalam mengakses aktivitas. Oleh karena itu setiap aktivitas memiliki frekuensi *user* yang berbeda - beda dalam melakukan akses sehingga menyebabkan *waiting time* yang tinggi. terdapat paling banyak 18 frekuensi akses dan paling sedikit 1 frekuensi akses untuk mata kuliah Pemrograman Berorientasi Objek, sedangkan pada mata kuliah Struktur Data memiliki paling banyak 28 frekuensi akses dan paling sedikit 1 frekuensi akses.

### V. KESIMPULAN

Berdasarkan hasil penelitian Tugas Akhir ini, ada beberapa kesimpulan yang didapatkan oleh peneliti, diantaranya sebagai berikut:

- A. *Heuristic Miner* dapat memodel kan *event log* ke dalam model proses dengan baik, nilai *fitness* untuk mata kuliah Pemrograman adalah 0,357 dan mata kuliah Struktur Data adalah 0,753. Terdapat tiga parameter yang paling berpengaruh pada nilai *fitness* mata kuliah Pemrograman Berorientasi Objek dan mata kuliah Struktur Data. Parameter yang paling berpengaruh yaitu *Positive Observation Threshold*, karena frekuensi kedua mata kuliah berbeda.
- B. Model proses terbaik diperoleh dengan menggunakan pengaturan parameter untuk Pemrograman Berorientasi Objek, *Positive Observation Objek* (PT) adalah 50, *Relative-to-best Threshold* (RT) adalah 0,05, dan *Dependency Threshold* (DT) adalah 0,9. Sedangkan untuk Struktur Data, *Positive Observation Threshold* (PT) adalah 15, *Relative-to-best Threshold* (RT) adalah 0,05, dan *Dependency Threshold* (DT) adalah 0,9. Perubahan pada parameter *Positive Observation Threshold* (PT) mempengaruhi nilai *fitness*.

### REFERENSI

- [1] Kementerian Pendidikan dan Kebudayaan, "Surat Edaran No 15 Tahun 2020 Tentang Pedoman Pelaksanaan Belajar Dari Rumah Selama Darurat Bencana COVID-19 di Indonesia," *Sekr. Nas. SPAB (Satuan Pendidik. Aman Bencana)*, no. 15, pp. 1–16, 2020, [Online]. Available: <https://www.kemdikbud.go.id/main/blog/2020/05/ke-mendikbud-terbitkan-pedoman-penyelenggaraan-belajar-dari-rumah>
- [2] D. Adidrana, D. Haryadi, and S. A. Rozano, "Integrasi Learning Management System dan Database Eksternal Menggunakan Oracle Studi Kasus: IT Telkom Jakarta," *J. Informatics Commun. Technol.*, vol. 3, no. 2, pp. 22–32, 2021, doi: 10.52661/j\_ict.v3i2.80.
- [3] P. Berking and S. Gallagher, "Choosing a Learning

- Management System Advanced Distributed Learning (ADL) Initiative,” 2016. [Online]. Available: <http://creativecommons.org/licenses/by-nc-sa/4.0/>
- [4] R. K. Ravikiran and K. R. Anil Kumar, “Experimental performance analysis of confidence-based online assessment portal in e-learning using data mining,” *Mater. Today Proc.*, vol. 47, pp. 5912–5917, 2021, doi: 10.1016/j.matpr.2021.04.456.
- [5] A. P. Kurniati, G. Agung, and A. Wisudiawan, “ANALISIS KESIAPAN PENERAPAN PROCESS MINING PADA SISTEM MANAJEMEN PEMBELAJARAN UNIVERSITAS TELKOM,” vol. 8, no. 6, pp. 1227–1236, 2021, doi: 10.25126/jtiik.202183637.
- [6] A. J. M. M. Weijters, W. M. P. van der Aalst, and A. K. A. de Medeiros, “Process Mining with the HeuristicsMiner Algorithm,” *Beta Work. Pap.*, 2006.
- [7] Y. Li, “A remodeling method of automatic learning process based on LMS in E-learning,” in *2009 International Conference on Web Information Systems and Mining, WISM 2009*, 2009, pp. 565–569. doi: 10.1109/WISM.2009.120.
- [8] C. dos S. Garcia *et al.*, “Process mining techniques and applications – A systematic mapping study,” *Expert Systems with Applications*, vol. 133. Elsevier Ltd, pp. 260–295, Nov. 01, 2019. doi: 10.1016/j.eswa.2019.05.003.
- [9] P. Chandrasekar and K. Qian, “The Impact of Data Preprocessing on the Performance of a Naive Bayes Classifier,” in *Proceedings - International Computer Software and Applications Conference*, Aug. 2016, vol. 2, pp. 618–619. doi: 10.1109/COMPSAC.2016.205.
- [10] M. Bozkaya, J. Gabriels, and J. M. Van Der Werf, “Process diagnostics: A method based on process mining,” in *Proceedings - International Conference on Information, Process, and Knowledge Management, eKNOW 2009*, 2009, pp. 22–27. doi: 10.1109/eKNOW.2009.29.
- [11] S. J. J. Leemans, E. Poppe, and M. T. Wynn, “Directly follows-based process mining: Exploration & a case study,” in *Proceedings - 2019 International Conference on Process Mining, ICPM 2019*, Jun. 2019, pp. 25–32. doi: 10.1109/ICPM.2019.00015.
- [12] R. Sarno and Y. A. Effendi, “Hierarchy process mining from multi-source logs,” *Telkonnika (Telecommunication Comput. Electron. Control.*, vol. 15, no. 4, pp. 1960–1975, Dec. 2017, doi: 10.12928/TELKOMNIKA.v15i4.6326.
- [13] D. Rahmawati, M. Ainul Yaqin, and R. Sarno, “Fraud detection on event logs of goods and services procurement business process using Heuristics Miner algorithm,” *Proc. 2016 Int. Conf. Inf. Commun. Technol. Syst. ICTS 2016*, no. November 2017, pp. 249–254, 2017, doi: 10.1109/ICTS.2016.7910307.
- [14] N. M. Lohmann, “BPM 2012 Demonstration Track,” *Proceedings*, no. September, 2012.
- [15] W. Van Der Aalst, “Process mining: Overview and opportunities,” *ACM Trans. Manag. Inf. Syst.*, vol. 3, no. 2, pp. 1–17, 2012, doi: 10.1145/2229156.2229157.