**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**INSTITUT TEKNLOGI SEPULUH NOPEMBER**

**USULAN TUGAS AKHIR**

# IDENTITAS PENGUSUL

**NAMA : FERNANDES P. SINAGA**

**NRP : 5110100190**

**DOSEN WALI : Prof. Dr. Ir. Joko Lianto Buliali, M.Sc**

**DOSEN PEMBIMBING : 1. Prof. Ir. Riyanarto Sarno, SE, MSc, PhD  
 2. Abdul Munif, S.Kom, M.Sc.**

# JUDUL TUGAS AKHIR

“Rancang Bangun *Fuzzy Association Rule Miner* untuk Mendeteksi *Fraud* pada Proses Bisnis *Enterprise Resource Planning (ERP)*.”

# LATAR BELAKANG

Saat ini organisasi dunia bisnis sangat bergantung pada sistem informasi digital untuk memberikan mereka pemahaman mengenai proses bisnis yang berjalan. Contoh nyatanya adalah penggunaan *Workflow Management* (WFM) *systems* yang bertujuan untuk mengotomatisasikan proses bisnis [1]. WFM merupakan sebuah sistem yang mengawasi proses pertukaran informasi proses, aktifitas, kegiatan atau responbilitas dari 1 mesin atau pekerja ke pekerja lainnya [2].

Dari suatu *Workflow,* dapat dilakukan simulasi dari suatu proses. Hasil dari simulasi tersebut berupa *event log* yang berisi ­*case-case* yang dijalankan dalam simulasi tersebut. Dengan *event log* tersebut, dapat dilakukan penggalian proses yang berjalan yang dikenal sebagai *Process Mining*. *Process Mining* dilakukan untuk mengekstraksi informasi proses menjadi suatu model proses. Model proses yang dihasilkan hendaknya menggambarkan semua contoh proses yang terdapat di dalam *event log*. Melalui informasi proses yang telah didapatkan dari penggalian proses, dapat dilakukan peningkatan proses yang dapat memberikan dampak positif bagi perusahaan atau organisasi yang menggunakan proses bisnis tersebut.

Pada tugas akhir ini akan diimplementasikan penggalian proses menggunakan pendekatan *Fuzzy Association Rule Mining* untuk menangkap aturan asosiasi yang terdapat pada aktifitas-aktifitas di dalam *event log*. Aturan asosiasi tersebut dapat digunakan untuk mengetahui kebiasaan dan alur proses bisnis berjalan yang nantinya dapat digunakan untuk pengembangan proses lebih lanjut atau untuk mendeteksi kecurangan(*Fraud*) yang terdapat di dalam proses. Pendekatan *Fuzzy* digunakan untuk membantu proses pencarian aturan asosiasi dari data kuantitatif yaitu data yang tidak bersifat biner (1 atau 0) [3]. Hal ini dapat digunakan untuk mendeteksi daerah “abu-abu” yang dapat diidentifikasikan sebagai sebuah *Fraud* dalam proses bisnis tersebut. Dengan pendekatan ini, proses bisnis dapat ditingkatkan untuk menghasilkan model proses yang lebih efektif.

# RUMUSAN MASALAH

Rumusan masalah yang diangkat dalam Tugas Akhir ini dapat dipaparkan sebagai berikut:

1. Bagaimana menganalisa dan mengolah *event log* menjadi himpunan data *fuzzy*.
2. Bagaimana cara memetakan himpunan data *fuzzy* ke dalam fungsi keanggotaan
3. Bagaimana menghasilkan aturan asosiasi dengan pendekatan *Association Rule Mining.*
4. Bagaimana dapat mendeteksi *Fraud* dari aturan asosiasi yang dihasilkan.

# BATASAN MASALAH

Permasalahan yang dibahas dalam tugas akhir ini memiliki beberapa batasan, di antaranya sebagai berikut:

1. Bahasa pemrograman menggunakan bahasa Java
2. Data uji yang akan digunakan berupa *event log* proses bisnis ERP.
3. *Plugin* yang dihasilkan khusus untuk perangkat lunak ProM.

# TUJUAN PEMBUATAN TUGAS AKHIR

Tujuan dari Tugas Akhir ini adalah sebagai berikut:

1. Membuat program untuk menganalisa dan mengolah *event log* menjadi himpunan data *fuzzy.*
2. Membuat program untuk memetakan himpunan data *fuzzy* ke dalam fungsi keanggotaan.
3. Membuat program untuk menghasilkan aturan asosiasi dengan pendekatan *Association Rule Mining.*
4. Membuat program untuk mendeteksi *Fraud* dari aturan asosiasi yang dihasilkan.

# MANFAAT TUGAS AKHIR

Manfaat dari Tugas Akhir ini adalah sebagai berikut:

1. Mempermudah dalam menganalisa dan mengolah *event log* menjadi himpunan data *fuzzy*.
2. Mempermudah dalam memetakan himpunan data *fuzzy* ke dalam fungsi keanggotaan.
3. Mempermudah dalam menghasilkan aturan asosiasi dengan pendekatan *Association Rule Mining*.
4. Mempermudah untuk mendeteksi *Fraud* dari aturan asosiasi yang dihasilkan.
5. Mempermudah menganalisa dan mendeteksi *Fraud* yang bersifat “abu-abu”.

# TINJAUAN PUSTAKA

1. **Plugin ProM**

ProM merupakan kerangka kerja umum yang *open-source* untuk mengimplementasikan penggalian proses. Plug-in (atau plugin,ekstensi) adalah komponen yang menambahkan fitur spesifik ke dalam aplikasi perangkat lunak yang ada. Dalam tugas akhir ini, akan dibuat plugin khusus untuk perangkat lunak ProM yang bergerak dalam penggalian proses.

1. ***Association Rule***

*Association Rule* (aturan asosiasi) menggambarkan keterkaitan dari sebuah set dari hubungan-hubungan nilai atribut, yang biasa dipanggil *items*. *Association Rule* merupakan asosiasi antara 2 atau lebih item/elemen/tugas di dalam sebuah database. *Market Basket Analysis* (MBA) merupakan aplikasi terbesar untuk algoritma yang menemukan aturan asosiasi ini. Hal ini merupakan teknik pemodelan yang berdasar pada teori dimana jika seorang membeli kumpulan item tertentu, maka dia juga akan cenderung membeli kumpulan item lainnya [4]. Contoh aturan asosiasi adalah sebagai berikut :

Bread, Milk => Butter | 90%

Item yang berada pada sebelah kiri dari aturan asosiasi di atas disebut *antecedent* (yang mendahului) dan item yang berada pada sebelah kanan disebut *consequents*. Sebuah aturan asosiasi dapat memiliki banyak *antecedent* dan *consequent*. Bilangan 90% di atas mengindikasikan bahwa 90% pembeli yang membeli roti dan susu juga membeli mentega. Persentase ini mengindikasikan kepastian (*confidence*) dari aturan asosiasi ini.

Kepastian/*Confidence* merupakan satu dari unsur pengukuran tingkat ketertarikan dari sebuah aturan asosiasi. Pengukuran lainnya adalah *support*. *Support* mengindikasikan kegunaan dari sebuah aturan asosiasi. Sebagai contoh, jika aturan di atas memiliki nilai *support* 5%, hal itu berarti 5% dari semua transaksi yang telah dianalisa menunjukkan bahwa roti, susu dan mentega dibeli secara bersama. Ketika aturan asosiasi ini diaplikasikan terhadap *event log*, akan diambil asosiasi dan pola yang sering ada diantara berbagai *event* di dalam *event log*.

**Definisi:**

1. ***Association Rule***

Aturan asosiasi secara formal didefinisikan sebagai pernyataan X=>Y, di mana X dan Y adalah *itemset*, dan Y bukan merupakan *itemset* kosong. X dan Y adalah kumpulan item dari data transaksi. Aturan asosiasi menyarankan sebuah relasi kejadian yang kuat antara item pada *antecedent* dan *consequent* pada aturannya.

1. ***Support***

Aturan asosiasi memerlukan kumpulan item yang sering muncul di database atau catatan transaksi. Hal ini menandakan bahwa hanya item dengan nilai *support* yang tinggi yang digunakan. Nilai *support* merupakan jumlah atau banyaknya suatu item muncul di dalam catatan transaksi. Di dalam aturan asosiasi, terdapat penentuan nilai ambang batas minimum untuk nilai *support*. Jika sebuah kumpulan item melebihi ambang batas minimum *support* tersebut, maka kumpulan item tersebut dapat diarahkan sebagai frekuen *itemset*. Nilai *support* dapat dihitung dengan menggunakan persamaan berikut:

*Support*(a,b) =

1. ***Confidence***

*Confidence* dari sebuah aturan asosiasi X=>Y adalah probabilitas untuk menemukan Y pada kumpulan transaksi. Dengan kata lain, *confidence* mengindikasikan sebagaimana sering munculnya item di Y pada transaksi yang mengandung X. Nilai *confidence* dapat dihitung dengan menggunakan persamaan berikut:

*Confidence*(a,b) =

**Algoritma untuk *Association Rule*:**

Semua algoritma untuk penggalian aturan asosiasi meliputi 2 tahap:

1. **Cari semua frekuen *itemset***.

Merupakan *itemset* yang paling tidak muncul sebanyak batas *support* minimum yang telah ditentukan.

1. **Hasilkan aturan asosiasi yang kuat dari frekuen *itemset*.**

Pada tahap ini, aturan asosiasi yang kuat berasal dari frekuen *itemset* yang dihasilkan pada tahap pertama. Aturan yang kuat adalah aturan yang memenuhi batas ambang minimum *support* dan *confidence*.

Terdapat beberapa algoritma untuk pencarian aturan asosiasi seperti *Apriori, AprioriTid, PredictiveApriori, Tertius*, dan sebagainya [4].

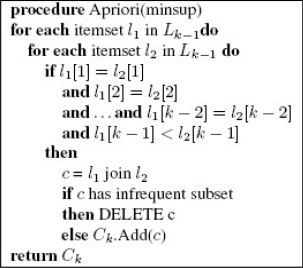
**Algoritma Apriori**

Algoritma apriori menggunakan pencarian berdasarkan level untuk menghasilkan frekuen *itemset* yang melintas dari frekuen 1-*itemset* hingga ke ukuran maksimum dari frekuen *itemset*. Pencarian iterative ini berlanjut hingga tidak terdapat frekuen *itemset* baru yang dihasilkan. Langkah-langkahnya adalah sebagai berikut:

1. Untuk batas ambang nilai *support* s, item-item yang lolos nilai minimum *support* dinotasikan dengan L1.
2. Pasangan dari L1 adalah kandidat untuk *itemset* ukuran 2. Kandidat ini dinotasikan dengan C2. Pasangan frekuen *itemset* yang memenuhi nilai *support* s menjadi frekuen *itemset* ukuran 2 dan dinotasikan dengan L2.
3. Lakukan langkah kedua berikut sampai tidak ada frekuen *itemset* yang dihasilkan.
4. Setiap kali frekuen *itemset* diperoleh, untuk tiap frekuen *itemset* 1, hasilkan semua subset yang tidak kosong dari 1.
5. Untuk setiap subset s dari 1 yang tidak kosong, didapatkan aturan “s=>(1-s)” jika perbandingan dari *support*(1) ke *support*(s) lebih besar dari atau sama dengan nilai ambang batas minimum *confidence*.

**Apriori’s pseudocode**

*Pseudocode* dari algoritma Apriori dapat dilihat pada gambar 1.



Gambar 1Algoritma Apriori

1. ***Fuzzy Set***

Himpunan *Fuzzy* merupakan himpunan umum yang memungkinkan untuk adanya tingkat keanggotaan untuk elemen-elemennya. Biasanya interval satuan riil [0; 1] dipilih sebagai struktur derajat keanggotaan [5]. *Fuzzy set* telah terbukti sangat berguna di banyak daerah di mana bahasa sehari-hari berpengaruh. Diberikan X sebagai himpunan global. µA merupakan fungsi keanggotaan yang mendefinisikan himpunan A.

Secara formal :

µA:X ->[0,1]

**Konsep *Fuzzy Set***

* *Support*: nilai *support* untuk sebuah himpunan *fuzzy* diperoleh dari himpunan *crisp* yang mengandung semua elemen yang memiliki derajat keanggotaan di himpunan A tidak 0.

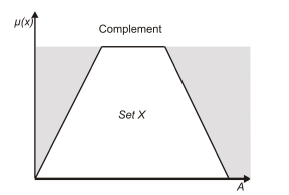
*Supp(A)={xЄX|µA(x)>0}*

* *Height:* tinggi dari sebuah himpunan *fuzzy* didefinisikan oleh nilai keanggotaan terbesar yang dicapai oleh unsur dari himpunan. Fuzzy set disebut normal jika setidaknya satu dari unsur-unsurnya mencapai keanggotaan kelas tertinggi.
* *α-cut:* didefinisikan oleh sebuah himpunan *crisp* Aα mengandung semua elemen yang memiliki keanggotaan kelas ke himpunan *fuzzy* yang lebih besar dari 0.
* *Scalar cardinality:* Penjumlahan dari nilai keanggotaan dari semua elemen dalam himpunan fuzzy

**Operasi *Fuzzy***

* ***Complement***

Pelengkap *fuzzy* didefinisikan sebagai : µA(x)=1- µA(x). Nilai pelengkap dari sebuah kelas keanggotaan 0.4 akan menjadi 0.6 (lihat gambar 2).

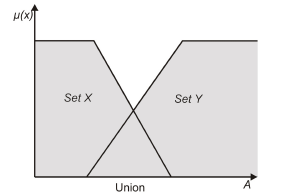


Gambar 2 Fuzzy Complement

* ***Union***

Perpaduan dari 2 himpunan *fuzzy* dapat secara umum dispesifikasi oleh sebuah fungsi : u:[0,1]x[0,1]->[0,1]. Untuk menghitung sebuah perpaduan *fuzzy*, dapat menggunakan persamaan :

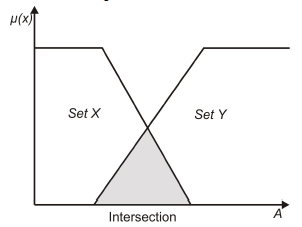
Jika 2 kelas keanggotaan 3.0 dan 0.6 melakukan perpaduan akan menjadi 0.6 (lihat gambar 3)



Gambar 3 Fuzzy Union

* ***Intersection***

Perpotongan pada *fuzzy* hampir sama dengan perpaduan pada *fuzzy*. Untuk menghitung nilai keanggotaan dengan perpotongan *fuzzy*, minimum dari kelas keanggotaan yang digunakan (lihat gambar 4) :



Gambar 4 Fuzzy Intersection

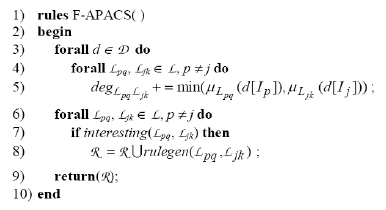
1. ***Fuzzy Association Rule***

*Fuzzy Association Rule* merupakan suatu solusi untuk menggali aturan asosiasi dari data kuantitatif yaitu data yang bukan bernilai biner, 0 dan 1. Terdapat 3 pendekatan yang dapat digunakan untuk melakukan *fuzzy ARL* ini yaitu:

1. Pendekatan kuantitatif
2. Struktur Taksonomi *Fuzzy*
3. Pendekatan *Item Set* perkiraan

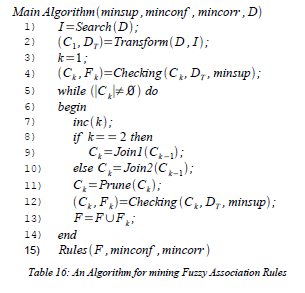
**Algoritma**

1. Algoritma untuk menggali *fuzzy association rule* pada basis data kuantitatif telah diusulkan yaitu F-APACS, yang mengerjakan istilah linguistic untuk mengambarkan keteraturan dan pengecualian tersembunyi daripada memisahkan atribut kuantitatif ke dalam himpunan *fuzzy* [6]. Algoritma ini dapat dilihat pada gambar 5.



Gambar 5 F-APACS Algorithm

1. Algoritma lainnya seperti Apriori juga dapat digunakan sebagai dasar tetapi menggabungkan himpunan *fuzzy* untuk menggali nilai kuantitatif dari sebuah database. Algoritma ini pertama mengubah tiap atribut kuantitatif menjadi himpunan *fuzzy* dan memetakan item-item tersebut melalui fungsi keanggotaan. Kemudian algoritma apriori menghasilkan aturan asosiasi menggunakan jumlah *fuzzy* yang dikumpulkan sebelumnya [7].
2. Algoritma lain dengan pendekatan Apriori yang membagi proses penggalian menjadi 2 tahap yaitu pencarian frekuen *itemset* dan pembentukan aturan asosiasi dari basis data kuantitatif. Untuk *pseudocode* nya dapat dilihat pada gambar 6 [8].



Gambar 6 Algorithm for mining Fuzzy Association Rules

Untuk notasi yang terdapat pada algoritma di atas dpat dilihat pada tabel 1.

Tabel 1 Notation

|  |  |
| --- | --- |
| D | *The database* |
| DT | *The transformed database* |
| FK | *set of frequent k -itemsets (having k items)* |
| CK | *set of candidate k -itemsets (having k items)* |
| I | *complete itemset* |
| *Minsup* | *support threshold* |
| *Minconf* | *confidence threshold* |
| *mincorr* | *correlation threshold* |

1. ***Fraud Detection***

*Fraud* (Penipuan) merupakan ketidakjujuran direncanakan menyebabkan cedera lain dengan membuat pernyataan palsu, menyembunyikan, atau mengabaikan fakta material [9]. *Fraud* juga dapat diartikan sebagai penyalahgunaan sistem organisasi tanpa memimpin dasarnya konsekuensi hukum [10].

Proses penipuan bisnis adalah penipuan yang disebabkan oleh salah mengartikan urutan proses. Sebuah kasus penipuan mungkin berisi ketidak cocokan dibandingkan dengan kasus hukum

***Process Mining for Fraud Detection***

Dalam kasus deteksi penipuan, *Process Mining* memberikan kontribusi beberapa keuntungan. Manfaat memeriksa kesesuaian (*Conformance Checking*) untuk membandingkan data aktual untuk model standar.

Manfaat lain yang dihasilkan dari penyelidikan menggunakan proses pertambangan adalah analisis kontrol aliran. Hal ini bisa mendeteksi aktivitas yang dilewati, aktivitas yang dimasukkan, dan urutan yang salah. Dalam konteks deteksi penipuan, bagian menyimpang tersebut dipandang sebagai tipuan mencurigakan [9].

**ARL diaplikasikan untuk mendeteksi penipuan**

Dalam konteks deteksi penipuan, aturan berdasarkan data historis harus diperhitungkan. Aturan yang telah ditetapkan dapat meningkatkan sistem dalam kasus mendeteksi aktivitas penipuan sama [9]. Karakteristik dari sebuah penipuan dijadikan sebagai *antecedent* dan status penipuan sebagai *consequent*.

Pada tabel 2, dapat dilihat asosiasi antara karakteristik dari suatu penipuan dan status penipuan.

Tabel 2 Aturan asosiasi dari karakteristik dan status penipuan

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Case ID** |  | | | | |  | **Min Support** | **Confidence** |
|
|  |  |  |  |  | **Fraud** |
|
| 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0.05 | 0.66 |
| 2 | 0 | 0 | 1 | 0 | 0 | 0 | 0.05 | 0.20 |
| 3 | 0 | 1 | 0 | 1 | 0 | 1 | 0.05 | 0.92 |
| 4 | 1 | 1 | 1 | 1 | 1 | 1 | 0.05 | 0.70 |

Definisi umum dari aturan asosiasi dari data di atas yaitu:

* C1-Cn merupakan karakteristik penipuan.
* X adalah *itemset* dari karakteristik penipuan.
* Y adalah status penipuan
* X => Y adalah aturan jika *itemset* X terjadi maka Y dipertimbangkan
* *Supp(X)* adalah proporsi dari transaksi pada data set yang berisi *item set*.
* *Conf(X => Y)* adalah nilai *confidence* dari aturan X=>Y

(2.1)

(2.2)

Pada persamaan (2.1) dijelaskan bahwa nilai *confidence* dari aturan X=>Y diperoleh dari frekuensi, ketika X dan Y muncul, dibandingkan dengan frekuensi ketika hanya Y yang muncul pada *event log*. Pada persamaan (2.2), nilai ambang batas menetukan premis mana yang mempengaruhi *fraud*. Jika sebuah kasus terdiri dari beberapa premis dan tingkat *confidence* dari beberapa premis tersebut lebih tinggi dari nilai ambang batas, maka kasus tersebut dapat dianggap sebagai sebuah penipuan/*fraud*.

# RINGKASAN ISI TUGAS AKHIR

Perangkat lunak yang akan dibangun dapat menghasilkan aturan asosiasi yang berasal dari *event log* proses bisnis ERP yang terlebih dahulu diolah menjadi himpunan data *fuzzy* yang kemudian akan dilakukan pencarian aturan asosiasi berdasarkan *fuzzy itemset* yang selanjutnya akan digunakan untuk proses pendeteksi *fraud* pada proses bisnis ERP tersebut. Berikut alur pemrosesan dan bentuk arsitektur perangkat secara sederhana (gambar 7) :



Gambar 7 Arsitektur Sistem

Tahap pertama merupakan tahap pemberian input data set berupa *event log* dari proses bisnis ERP yang telah berjalan. Langkah selanjutnya adalah menganalisa dan mengolah data tersebut menjadi *fuzzy set*. Langkah selanjutnya adalah memetakan *fuzzy itemset* tersebut ke dalam fungsi keanggotaan. Tahap berikutnya adalah melakukan penggalian aturan asosiasi dengan menggunakan pendekatan *Fuzzy Association Rule*. Dari tahap ini telah diperoleh aturan asosiasi dari proses bisnis tersebut yang akan digunakan untuk proses pendeteksian *fraud* yang terdapat pada proses tersebut. Hasil yang diharapkan adalah terbentuknya aturan asosiasi yang kuat dan memperoleh data mengenai *fraud* pada proses bisnis yang digali.

Selanjutnya kasus penggunaan yang terdapat di dalam *plugin* ini dijelaskan pada gambar 8.



Gambar 8 Diagram Kasus Penggunaan

Dalam diagram kasus penggunaan di atas, seorang pengguna akan menjalankan 4 buah kasus penggunaan dengan penjelasan sebagai berikut :

1. Membuat *Fuzzy Set* dari *Event log.*

*Event log* yang diperoleh sebagai masukan dianalisa dan diolah menjadi *fuzzy set*.

1. Memetakan *Fuzzy Set* ke fungsi keanggotaan.

*Fuzzy set* yang dihasilkan kemudian dipetakan ke dalam fungsi keanggotaan.

1. Menghasilkan *Association Rule* dari *Fuzzy Set*.

Dari *Fuzzy set* yang telah dipetakan tersebut, akan dihasilkan aturan asosiasinya.

1. Menghasilkan laporan Fraud

Dari aturan asosiasi yang diperoleh, akan diidentifikasi case-case yang mengalami *fraud* sehingga akan menghasilkan laporan *fraud*.

# METODOLOGI

## Penyusunan proposal tugas akhir

Proposal tugas akhir ini berisi mengenai rencana pembangunan perangkat lunak berupa *plugin* untuk menggali aturan asosiasi dari *event log* proses bisnis ERP dengan menggunakan pendekatan *Fuzzy Association Rule Miner*. Pada latar belakang dijelaskan mengenai isu yang berkembang mengenai penggalian proses saat ini. *Workflow Management System* (WMF) berguna untuk mengoptimalisasi proses bisnis yang dilakukan dengan menggali proses bisnis tersebut. Kemudian dilanjutkan dengan rumusan serta batasan masalah yang ingin diselesaikan. Setelah itu dilanjutkan dengan menjelaskan tujuan dan manfaat yang ingin dicapai dalam tugas akhir ini.

Pada bagian tinjauan pustaka dijelaskan mengenai framework, metode serta algoritma yang digunakan sebagai penunjang dalam pengerjaan tugas akhir ini. Algoritma yang dibahas yaitu mengenai konsep *Association Rule, Fuzzy set, Fuzzy Association Rule,* dan *Fraud Detection.* Kemudian dilanjutkan dengan ringkasan tugas akhir yang menjelaskan secara singkat mengenai arsitektur perangkat lunak beserta fitur-fitur yang akan dikembangkan. Selanjutnya pada bagian metodologi dijelaskan tahap-tahap pengerjaan tugas akhir seperti penyusunan proposal, studi literature, analisis dan desain perangkat lunak, implementasi perangkat lunak, pengujian dan evaluasi, serta penyusunan buku tugas akhir. Bagian terakhir dari proposal ini adalah pembuatan jadwal pengerjaan tugas akhir dan daftar pustaka yang digunakan dalam pembuatan proposal ini.

## Studi literatur

Dalam pembuatan tugas akhir ini, literature yang akan dipelajari sebagai ilmu penunjang penyelesaiannya adalah:

1. Mengolah *event log* menjadi data yang bersifat *fuzzy.*
2. Membentuk fungsi keanggotaan dari data *fuzzy* tersebut agar dapat mencari aturan asosiasi.
3. Menggali aturan asosiasi dengan menggunakan algoritma yang terdapat dalam *Association Rule Mining*.
4. Menggali informasi mengenai *fraud* di dalam suatu proses.
5. Dengan aturan asosiasi yang ada dilakukan pendeteksian *fraud* yang terdapat di dalam proses.

Selain literature di atas juga akan ditambahkan beberapa literature lainnya yang dapat menunjang penyelesaian tugas akhir ini.

## Analisis dan desain perangkat lunak

Aktor yang menjadi pelaku adalah pengguna ProM yang memanfaatkan *plugin* yang dihasilkan dari tugas akhir ini. Adapun kebutuhan-kebutuhan fungsional dari *plugin* ini adalah sebagai berikut:

1. Melakukan proses analisa dan kelola *event log* ke *fuzzy set*.
2. Melakukan proses pemetaan *fuzzy set* ke dalam fungsi keanggotaan.
3. Melakukan proses penggalian aturan asosiasi.
4. Melakukan proses pendeteksian *fraud* menggunakan aturan asosiasi yang dihasilkan.

## Implementasi perangkat lunak

Dalam tugas akhir ini nantinya akan menghasilkan plugin pada ProM 6.2. Masukan data berupa *event log* yang diambil dari proses ERP. Hasil perangkat lunak berupa aturan asosiasi yang dapat digunakan untuk mendeteksi *fraud* pada proses. Berikut beberapa hal yang diperlukan dalam implementasi :

1. IDE menggunakan *Eclipse Kepler*.
2. *Java Development Kit 7.0.*
3. *Java Runtime Environment*.
4. Perangkat yang dikembangkan merupakan *plugin ProM*
5. *Library ProM*

## Pengujian dan evaluasi

Proses pengujian yang dilakukan adalah dengan metode pengujian *blackbox testing*. Dataset diperoleh dari *event log* suatu ERP yang telah berjalan. Dari data tersebut akan dihasilkan aturan asosiasi yang akan diperiksa apakah telah menghasilkan aturan asosiasi yang benar dan dapat mendeteksi *fraud* dari proses bisnis ERP tersebut.

## Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku tugas akhir secara garis besar antara lain:

1. Pendahuluan
   1. Latar Belakang
   2. Rumusan Masalah
   3. Batasan Tugas Akhir
   4. Tujuan
   5. Metodologi
   6. Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Pengujian dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka

# JADWAL KEGIATAN

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tahapan | Bulan (Tahun 2013) | | | | | | | | | | | | |
| Oktober | | | | Nopember | | | | Desember | | | |
| Analisa kebutuhan dan studi literatur |  |  |  |  |  |  |  |  |  |  |  |  |
| Perancangan system |  |  |  |  |  |  |  |  |  |  |  |  |
| Implementasi |  |  |  |  |  |  |  |  |  |  |  |  |
| Uji coba dan evaluasi |  |  |  |  |  |  |  |  |  |  |  |  |
| Penyusunan buku |  |  |  |  |  |  |  |  |  |  |  |  |

# DAFTAR PUSTAKA

|  |  |
| --- | --- |
| [1] | J. Stoop, "A case study on the theoretical and practical value of using process mining for the detection of fraudulent behavior in the procurement process," in *Process Mining and Fraud Detection*, Netherlands, Twente University, 2012. |
| [2] | N. Khajehzadeh, Data and Process Mining Applications on a Multi-Cell Factory Automation Testbed Master of Science Thesis, 2012. |
| [3] | L. Helm, Fuzzy Association Rules An Implementation in R, Vienna: Vienna Univesity of Economics and Business Administration, 2007. |
| [4] | Gupta, S, W.M.P van der Aalst, A.J.M.M.Weijters, and A.K.Alves de Medeiros, Workflow and Process Mining in Healthcare, Eindhoven, 2007. |
| [5] | S. Gottwald, Universes of Fuzzy Sets and Axiomatizations of Fuzzy Set Theory, Springer: Studia Logica, 2006. |
| [6] | Chan, Keith C.C, Au, Wai-Ho, "An Effective Algorithm for Discovering Fuzzy Rules in Relational Database," *IEEE World Congress on Computational Intelligence,* 1998. |
| [7] | Hen, Tzung-Pei, Kuo, Chan-Sheng, Chi, Sheng-Chai, "A Fuzzy Data Mining Algorithm for Quantitative Values," *Knowledge-Based Intelligent Information Engineering Systems,* 1999. |
| [8] | A. Gyenesei, "A Fuzzy Approach for Mining Quantitative Association Rules," *Turku Centre for Computer Science Technical Reports,* 2000. |
| [9] | Dewandono, R. D., Sarno, R., Process Sequence Mining For Fraud Detection Using Complex Event Processing, Surabaya, 2013. |
| [10] | Lundin, E., Kvarnstr, H. and Jonsson, E., A Synthetic Fraud Data, 2002. |