

Guilherme Gustavo Henschel

Manual de Usuário e Desenvolvimento

Valinor – Ferramenta de Descoberta de Anomalias baseada em
Mineração de Dados

Guilherme Henschel
25/10/2018

Sumário

1. Introdução	2
1.1. Sobre.....	2
1.2. Acronimos, Abreviações e definições	3
2. Manual de Usuário.....	5
2.1. Fluxicon Disco Instalação	5
2.2. Fluxicon Disco Exportação do Modelo de Processo	5
2.3. Utilizando o Valinor	6
2.3.1. Janela Principal	7
2.3.2. Janela do Algoritmo de Conformidade	8
2.3.3. Janela de Algoritmo de Classificação	10

1. Introdução

1.1. Sobre

Este software se propõe a combinar o potencial de dois processos de mineração, a saber: de Processos e de Dados. Esta combinação implica que a Mineração de Processos crie um novo atributo, referente a conformidade ou não. A partir deste novo atributo agregado ao conjunto de dados, a Mineração de Dados permitirá descobrir os fatores associados a esta anomalia (não conformidade).

A nível exemplo, a ferramenta usaria o Modelo de Processo (e.g. Figura 1) e um log de eventos (e. g. Tabela 1) afim de gerar um novo atributo a ser adicionado a um conjunto de dados (e.g. Tabela 2) gerando um novo conjunto composto do conjunto anterior mais o atributo de conformidade (e.g. Tabela 3).

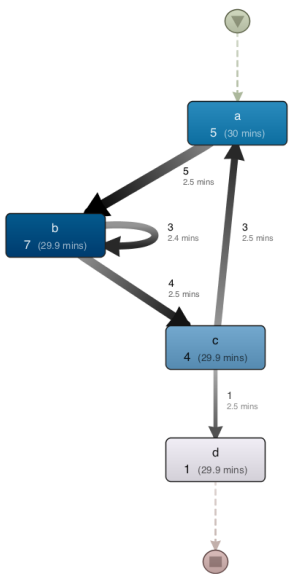


Figura 1 - Exemplo de modelo de processo minerado a partir de log de eventos. Fonte: O Autor.

ID Caso	Atividade	Horário de Inicio	Horário de Termino (Opcional)
10	a	2018-04-06T02:39:37.583302	2018-04-06T03:17:23.768729
10	b	2018-04-06T03:18:17.563624	2018-04-06T03:50:21.139901
10	c	2018-04-06T03:51:18.497461	2018-04-06T04:22:49.252083
10	d	2018-04-06T04:27:46.609520	2018-04-06T04:50:02.818474

...
40	a	2018-04-06T04:34:20.563110	2018-04-06T04:37:34.083913
40	b	2018-04-06T04:38:20.138525	2018-04-06T04:54:44.951611
40	c	2018-04-06T04:58:45.436672	2018-04-06T05:37:05.828534
40	d	2018-04-06T05:41:18.405130	2018-04-06T05:51:51.533523
...

Tabela 1 - Exemplo de Log de Eventos. Fonte: O Autor.

ID Caso	Atributo 1	Atributo 2	Atributo 3	Atributo 4
10	True	6	0.20230633250841934	s3
40	False	4	0.010772960285866606	s5
56	False	5	0.5127887155117519	s1
81	True	7	0.2517624250099463	s5
118	True	1	0.5535074106934649	s2

Tabela 2 - Exemplo de Log de Dados para Mineração de Dados. Fonte: O Autor.

ID Caso	Atributo 1	Atributo 2	Atributo 3	Atributo 4	Conformidade
10	True	6	0.20230633250841934	s3	Conforme
40	False	4	0.010772960285866606	s5	Conforme
56	False	5	0.5127887155117519	s1	Não Conforme
81	True	7	0.2517624250099463	s5	Conforme
118	True	1	0.5535074106934649	s2	Conforme

Tabela 3 - Exemplo de Log Pós Processado Utilizando a Ferramenta. Fonte: O Autor.

Utilizando linguagem de programação de alto nível (Python 3.6) desenvolve-se o Valinor, uma ferramenta que encapsula todos o algoritmo de avaliação de Conformidade a partir de um modelo processado pelo Fluxicon Disco e os logs de eventos, e a mineração de dados tradicional.

Valinor sempre pode ser encontrado atualizado com todos os arquivos fontes no seguinte endereço: <https://github.com/ggHenschel/Valinor>

1.2. Acronomimos, Abreviações e definições

A seguir são apresentadas definições de termos e softwares que são citados ao longo do documento:

1. Disco – Fluxicon Disco, Ferramenta Mineração de Processos com foco em mercado;
2. Log de Eventos – Conjuntos de dados compostos de Atividade e uma *Timestamp*;
3. Mineração Dados – Metodologia para processamento de grande quantidade de dados afim descobrir padrões consistentes;
4. Mineração de Processos – Metodologia para descoberta de modelos de processos a partir de log de eventos;
5. SciPy – Biblioteca Científica para Python;
6. *Timestamp* – Tipo de dado que denomina uma data e hora específica;

2. Manual de Usuário

Antes de usar o Valinor o usuário deve estar ciente de alguns passos a serem cumpridos. A ferramenta não é *Stand Alone*, ou seja, ela depende de outros softwares afim de funcionar corretamente. Para tal o usuário deve primeiro Instalar e estar acostumado com o uso do software Disco.

2.1. Fluxicon Disco Instalação

Para Instalar o Disco o usuário deve adquiri-lo no *website* <https://fluxicon.com/disco/> . Ele possui um tempo de demonstração, e oferece licença acadêmica. A compra do software pode ser realizada após uma avaliação e como forma de consultoria.



Figura 2- Website do Disco. Aperte no botão Azul escrito Download para iniciar a instalação. Fonte: Fluxicon Disco Website.

O site possui uma grande gama de informações sobre o software.

2.2. Fluxicon Disco Exportação do Modelo de Processo

Após o usuário importar o Log de Eventos para o Disco o usuário deve exportar o modelo gerado. Para tal ele deverá apertar no na imagem de uma flecha no canto inferior direito do software, onde está escrito *Export* (figura 3). E selecionar Xml como formato, apertando o botão *Export XML File* (figura 4).

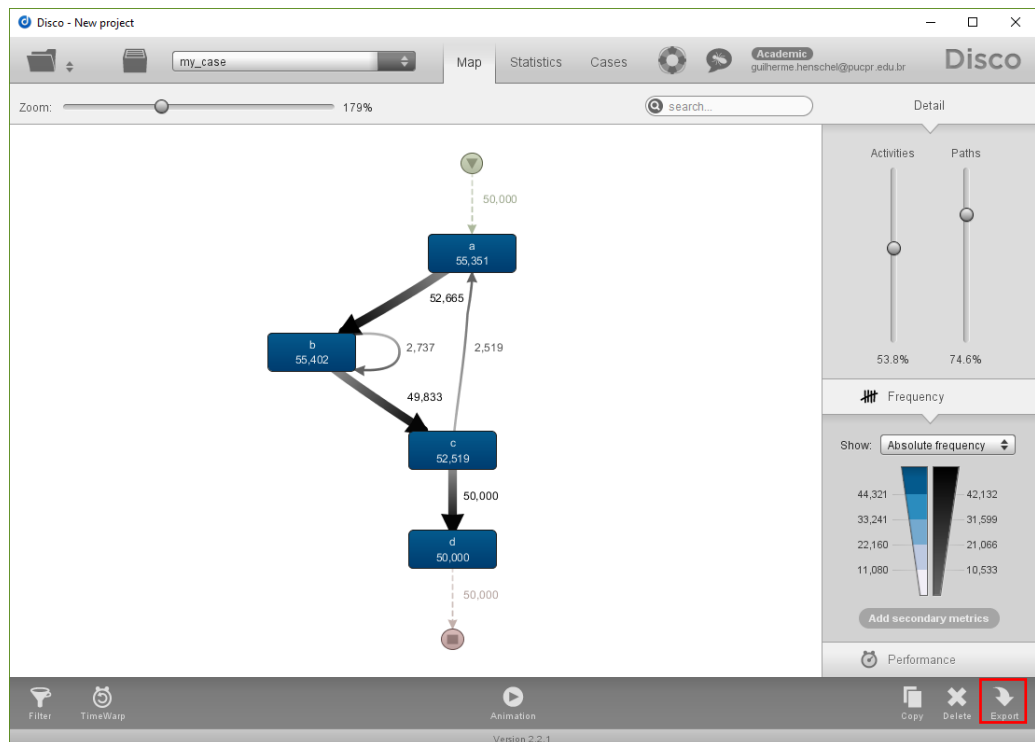


Figura 3 - Janela Disco com botão de Exportação destacado. Fonte: O Autor

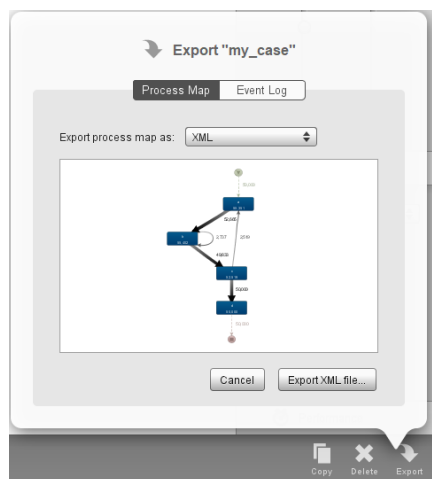


Figura 4 - Janela de Opções de Exportação do Fluxicon Disco.

O arquivo pode ser salvo em qualquer local.

2.3. Utilizando o Valinor

Valinor possui duas formas de utilização, Diretamente através dos seus arquivos em Python e seus arquivos executáveis. Esse documento irá focar no executável. No próximo capítulo haverá uma seção sobre os requisitos de desenvolvimento e eles cobrem como utilizar o software como um arquivo python.

2.3.1. Janela Principal

Ao abrir o Valinor o usuário estará na janela principal (figura 5) do software, nessa janela há apenas informações e os botões de importação dos dados.

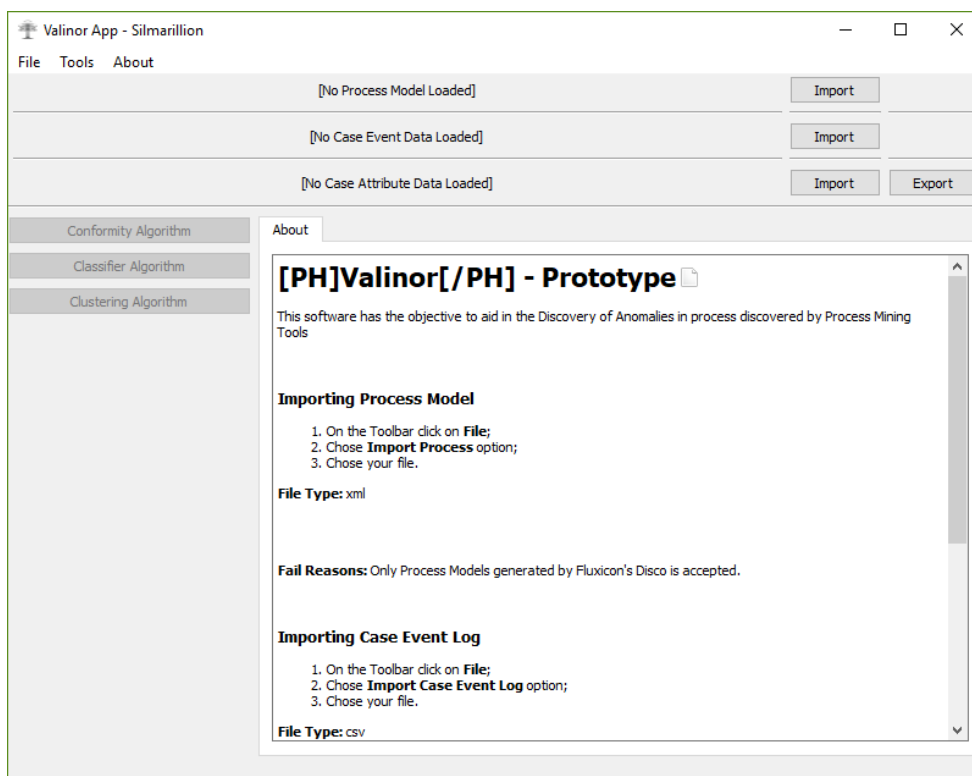


Figura 5 - Janela Principal sem dados Importados

Após importar seus dados as informações básicas dos mesmos serão disponibilizados em seus respectivos campos.

Na primeira linha do cabeçalho estarão os dados do Modelo de Processo. Na segunda linha os dados do Log de Evento. E na Terceira linha os dados do log de atributo (figura 6). Como os dados do log de atributo serão alterados esse log pode ser exportado apertando o botão Export.

O log de atributo completo sempre será exibido na aba de Case Attribute da Janela, que é atualizada sempre que o Log for atualizado.

Os algoritmos podem ser acessados apertando seus respectivos atributos, cada um será explicado em sua respectiva seção.

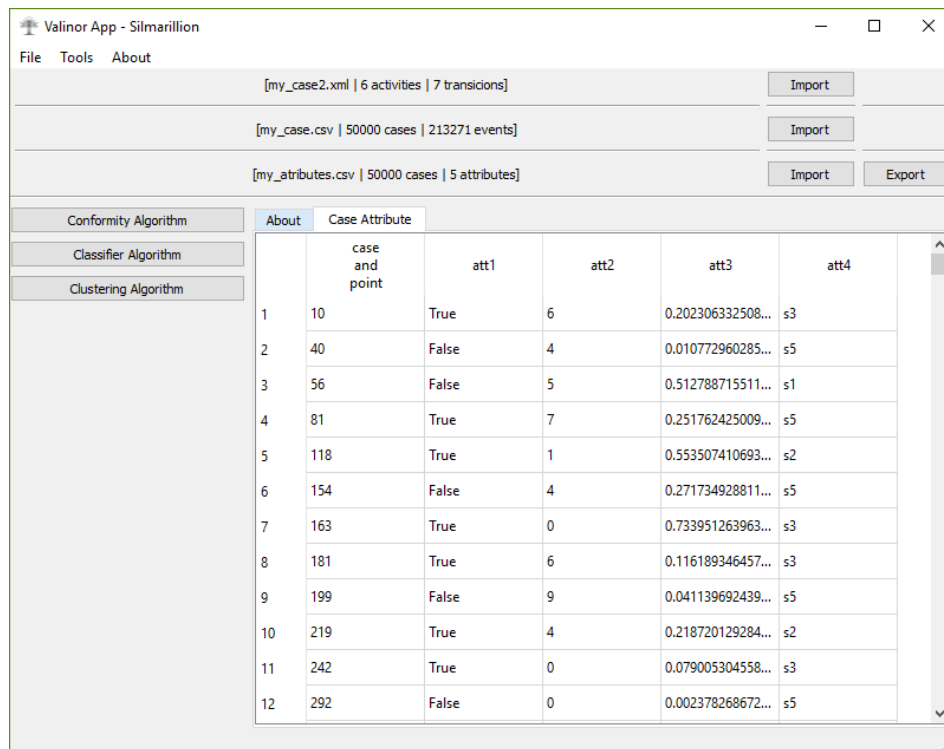


Figura 6 - Janela Principal com os Dados Importados.

2.3.2. Janela do Algoritmo de Conformidade

A janela de Conformidade é janela onde será definido as opções do algoritmo de conformidade dos eventos.

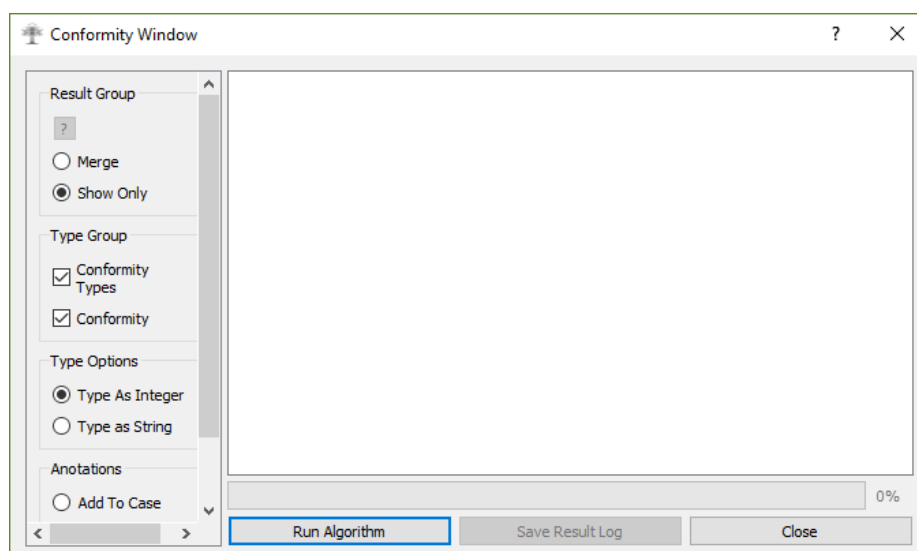


Figura 7 - Janela de Conformidade

2.3.2.1. Opções e Parâmetros

O algoritmo possui os seguintes grupos de parâmetros e opções:

- **Result Group** – Define o que será feito com o resultado, por padrão ele sempre será mostrado na tela.
 - **Merge** – O(s) atributo(s) de conformidade será(ão) adicionados aos casos no fim do log de atributos ao final de cada caso.
 - **Show Only** - Os resultados serão apenas apresentados em tela
- **Type Group** – Define quais serão os atributos de conformidade, ambos podem ser incluídos.
 - **Conformity Types** – Atributo de conformidade é um tipo relacionado ao motivo de não conformidade ou conformidade.
 - **Conformity** – Atributo de conformidade é um booleano (Falso ou Verdadeiro) que indica se o Caso é conforme (Verdadeiro) ou não conforme (Falso).
- **Type Options** – Apenas disponível se o usuário seleciona Conformity Types, e diz respeito ao formato de saída do atributo.
 - **Type as Integer** – O atributo de Tipo terá saída um inteiro com domínio [0,7].
 - **Type as String** – O atributo de Tipo terá saída um descritor em texto, que reflete uma das 8 possibilidades. (Explicações na próxima seção)
- **Anotations** – Reflete o que o algoritmo irá fazer com as anotações de motivo de não conformidade de cada caso.
 - **Add to Case** – As anotações serão adicionadas como um atributo extra ao final de cada caso do log de atributos.
 - **Show Only** – As anotações serão apenas mostradas em tela.
 - **Ignore** – As anotações serão descartadas pelo algoritmo.

2.3.2.2. *Tipos de Não Conformidade*

O software define 8 tipos de não conformidade.

- Conform, ou 0 – O caso é conforme com o processo minerado.
- Non Conform: Inexistent Activity , ou 1 – O caso possui uma atividade que não foi descrita no modelo de processo.
- Non Conform: Inexistent Transition, ou 2 – O caso possui uma transição entre duas atividades que não foi descrita no modelo de processo.

- Non Conform: Inexistent Activity and Inexistent Transition, ou 3 – O caso é uma combinação dos dois casos anteriores, mas a transição que falhou não é a que gerou a transição inexistente.
- Non Conform: Transition with Innappropriate Duration, ou 4 – O caso apresenta uma transição com tempo inadequado que extrapola os limites definidos pela mineração de processo.
- Non Conform: Transition with Inappropriate Duration and Inexistent Activity, ou 5 – O caso é uma combinação do Tipo 4 e Tipo 1.
- Non Conform: Transition with Innappropriate Duration and Inexistent Transition, ou 6 – O caso é uma combinação do Tipo 4 e Tipo 2.
- Non Conform: Transition with Innappropriate Duration, Inexistent Activity and Inexistent Transition, ou 7 – O caso é uma combinação do Tipo 1, Tipo 2 e Tipo 4.

Anotações – As anotações possuem as atividades e transições que falharam, e o motivo de falha.

2.3.3. Janela de Algoritmo de Classificação

O software implementou um algoritmo de classificação para análises iniciais que podem auxiliar o usuário a tomar decisões.

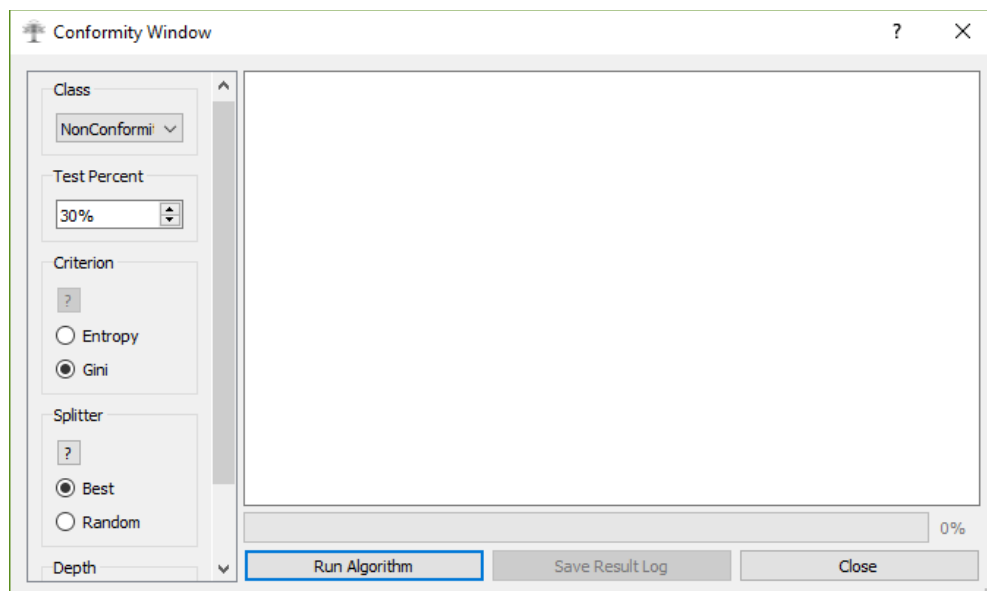


Figura 8 - Janela de Classificação

Os principais parâmetros são:

- **Class** – Determina qual atributo será considerado a classe a ser descoberta.

- **Test Percent** - Quantos por cento dos dados serão usados para calcular a acurácia do classificador, algoritmo utiliza Cross Validation.
- **Criterion** – Define o critério de divisão para o classificador de árvore.
- **Depth** – Define a profundidade máxima da árvore de decisão

3. Manual de Desenvolvimento

3.1. Linguagem de Programação e Bibliotecas

Valinor foi desenvolvido em Python 3.6, mas é funciona para Python 3.5+.

Para as partes de algoritmo as Principais bibliotecas utilizadas foram:

- Networkx – pode ser instalada utilizando pip install networkx.
- ScyKitLearn – pode ser instalada utilizando pip install sklearn.
- Numpy – pode ser instalada utilizando pip install numpy.

Para integrar os algoritmos e criar a interface gráfica foi utilizado Qt5, implementado para python na biblioteca PyQt5.

3.2. Diagrama de Classe

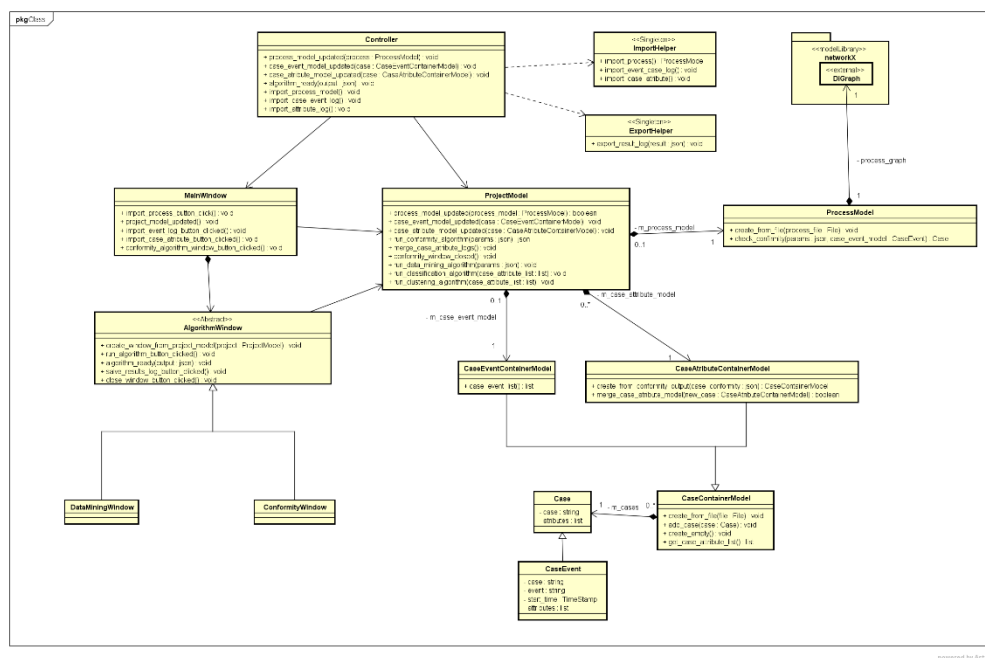


Figura 9 - Diagrama de Classe

3.3. dev_utils.py

Esse arquivo possui funções que auxiliam o desenvolvedor para desenvolver o programa.

A função `resource_path` auxilia na localização dos objetos, facilitando no desenvolvimento multiplataforma e na criação de um executável único.

3.4. Executável único

Para criar um executável único o desenvolvedor deve instalar o pacote PyInstaller e executar o comando:

```
$python -m PyInstaller main.py --clean -F -n Valinor --specpath "\\Valinor" --additional-hooks-dir="\\Valinor\\hooks\\" --add-data "<path-to>/Valinor/*.ui;" --add-data "<path-to>/Valinor/resources/*;resources" -w
```