**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ**

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ**

**«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук

Департамент программной инженерии

|  |  |  |
| --- | --- | --- |
| СОГЛАСОВАНО  Старший преподаватель департамента программной инженерии факультета компьютерных наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ С.А. Шершаков  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2017 г. |  | УТВЕРЖДАЮ  Академический руководитель образовательной программы «Программная инженерия»  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В. Шилов  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2017 г. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | *Подп. и дата* |  | | *Инв. № дубл.* |  | | *Взам. инв. №* |  | | *Подп. и дата* |  | | *Инв. № подл* |  | | **МОДУЛЬ РАСШИРЕНИЯ ПРИЛОЖЕНИЯ VTMINE FOR VISIO ДЛЯ  ВИЗУАЛИЗАЦИИ ГРАФОВЫХ МОДЕЛЕЙ PROCESS MINING**  **Пояснительная записка**  **ЛИСТ УТВЕРЖДЕНИЯ**  **RU.17701729.509000-01 81 01-1 ЛУ** | | |
|  |  | |
| Исполнитель: студент группы БПИ151  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /М.А. Роговец/  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2017 г. | |
|  | | |
|  | |  |

**2017**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| УТВЕРЖДЕН  RU.17701729.509000-01 81 01-1 ЛУ |  | |  | |
| |  |  | | --- | --- | | *Подп. и дата* |  | | *Инв. № дубл.* |  | | *Взам. инв. №* |  | | *Подп. и дата* |  | | *Инв. № подл* |  | | **МОДУЛЬ РАСШИРЕНИЯ ПРИЛОЖЕНИЯ VTMINE FOR VISIO ДЛЯ  ВИЗУАЛИЗАЦИИ ГРАФОВЫХ МОДЕЛЕЙ PROCESS MINING**  **Пояснительная записка**  **RU.17701729.509000-01 81 01-1**  **Листов 15** | | | | | |
|  | |  | | | |
|  | | | |
|  | | | | | |
|  | | | |  | |

**2017**

Оглавление

[1. ВВЕДЕНИЕ 3](#_Toc482344242)

[**1.1.** **Наименование программы** 3](#_Toc482344243)

[**1.2.** **Документы, на основании которых ведется разработка** 3](#_Toc482344244)

[2. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ 4](#_Toc482344245)

[**2.1.** **Назначение программы** 4](#_Toc482344246)

[**2.2.** **Краткая характеристика области применения** 4](#_Toc482344247)

[3. ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ 5](#_Toc482344248)

[**3.1.** **Постановка задачи на разработку программы** 5](#_Toc482344249)

[**3.2.** **Описание алгоритма программы и функционирования программы** 5](#_Toc482344250)

[**3.3.** **Описание функционирования программы** 7](#_Toc482344251)

[**3.4.** **Обоснование выбора алгоритма решения задачи** 7](#_Toc482344252)

[**3.5.** **Входные и выходные данные** 7](#_Toc482344253)

[**3.6.** **Состав технических и программных средств** 8](#_Toc482344254)

[4. ОЖИДАЕМЫЕ ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ 9](#_Toc482344255)

[ПРИЛОЖЕНИЕ 1 10](#_Toc482344256)

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 10](#_Toc482344257)

[ПРИЛОЖЕНИЕ 2 11](#_Toc482344258)

[ОПИСАНИЕ БИБЛИОТЕКИ MGRAPH 11](#_Toc482344259)

[ПРИЛОЖЕНИЕ 3 12](#_Toc482344260)

[ОПИСАНИЕ БИБЛИОТЕКИ MGRAPHLAYOUT 12](#_Toc482344261)

[ПРИЛОЖЕНИЕ 4 13](#_Toc482344262)

[СХЕМА КЛАССОВ MGRAPH 13](#_Toc482344263)

[ПРИЛОЖЕНИЕ 5 14](#_Toc482344264)

[СХЕМА КЛАССОВ MGRAPHLAYOUT 14](#_Toc482344265)

[ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ 15](#_Toc482344266)

1. **ВВЕДЕНИЕ**
   1. **Наименование программы**

Наименование программы: «Модуль расширения приложения VTMine for Visio для

визуализации графовых моделей Process Mining».

Наименование программы на английском языке: «VTMine for Visio Plug-in for Graph Process Mining Models Visualization».

Краткое наименование программы – «PMVisualizer».

* 1. **Документы, на основании которых ведется разработка**

Приказ Национального исследовательского университета «Высшая школа экономики» №2.3-02/1804-01 от 18.04.17

1. **НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ**
   1. **Назначение программы**

Данный плагин является расширением программы VTMine for Visio и предназначен для визуализации графовых моделей Process Mining, полученных в ходе работы программы.

* 1. **Краткая характеристика области применения**

Process Mining представляет собой дисциплину, целью которой является извлечение моделей процессов из журналов событий, анализ таких моделей и их дальнейшее усовершенствование. Данное приложение предоставляет возможность отрисовки графовых моделей в рамках данного направления. Визуализация структур данных может быть использована для анализа тех или иных процессов и их редактирования.

Таким образом, плагин может быть задействован в следующих видах деятельности:

1. Управление бизнес-процессами
2. Построение моделей на основе логов событий
3. Усовершенствование существующих моделей
4. Прогнозирование на основе анализа модели
5. Работа с графами и представленными ими структурами данных
6. **ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ**
   1. **Постановка задачи на разработку программы**

Разрабатываемый плагин должен:

1. Предоставлять интерфейсы и их реализации в виде основных графовых структур данных, используемых в Process Mining
2. Предоставлять алгоритмы для их визуализации на страницах Microsoft Visio
3. Рисовать модели средствами Microsoft Visio
   1. **Описание алгоритма программы и функционирования программы**

Задача отрисовки графа представляет собой графическую укладку графа на плоскость. Для каждого графа существует большое количество способов его визуализации. В общем смысле можно сказать, что все они представляют собой графическое отображение множества вершин и множества ребер, но на практике могут существенно различаться между собой в методах расположения вершин и ребер. В плагине «PMVisualizer» были использованы следующие алгоритмы укладки графов:

1. Произвольная укладка
2. Circular Layout
3. Simple Tree Layout
   * 1. **Произвольная укладка**

Для каждой вершины графа генерируется псевдослучайным образом точка, которая затем наносится на плоскость, представленную страницей Visio документа.

Псевдокод:

*Graph g;*

*Map<vertex, point> pos;*

*foreach(v in g.Vertices)*

*point p = GeneratePoint(v);*

*pos.Add(v,p);*

* + 1. **Circular Layout Algorithm**

Крайне распространенный алгоритм. Основной целью является расположение вершин на оси круга. При этом возможно учитывать порядок расположения с целью снижения количества пересечений.

Алгоритм может быть описан следующим псевдокодом:

*Graph g;*

*Map<vertex, point> pos;*

*Double P = 0;*

*Double[] halfSize;*

*i=0;*

*foreach(v in g.Vertices)*

*halfSize[i]=squareRoot(v.height\*v.height + v.width\*v.width)\*0.5;*

*P = P + halfSize[i]\*2;*

*i=i+1;*

*Double radius = P/2\*π;*

*angle=i=0;*

*foreach(v in g.Vertices)*

*angle+=sin(halfSize[i]\*0.5/radius)\*4;*

*radius=angle/(2\*π)\*radius;*

*angle=i=0;*

*foreach(v in g.Vertices)*

*a = sin(halfSize[i]\*0.5/radius)\*2;*

*angle+=a;*

*pos.Add(v,point(cos(angle)\*radius + radius, sin(angle)\*radius+radius;*

*angle+= a;*

*Visualize(g, pos);*

* + 1. **Simple Tree Layout Algorithm**

Рекурсивный метод разбиения дерева на поддеревья и расположения их симметрично относительно текущего узла.

Код алгоритма на языке C#:

void CalculateLoc(INode n, double x, double y)

{

int childrenCount = tree.Children(n).ToList<INode>().Count;

if (childrenCount == 0)

locations.Add(n, new Point(x, y));

if (childrenCount == 1)

CalculateLoc(tree.Children(n).ToList<INode>()[0], x, y - 2.0);

if (childrenCount % 2 == 0)

{

double stepX = x / childrenCount;

var children = tree.Children(n).ToList<INode>();

int j = 0;

for (int i = childrenCount / 2; i > 0; i--)

{

CalculateLoc(children[j], x - i \* stepX, y - 2.0);

CalculateLoc(children[childrenCount - j - 1], x + i \* stepX, y - 2.0);

j++;

}

if (!locations.ContainsKey(n))

locations.Add(n, new Point(x, y));

}

else

{

if (childrenCount != 1)

{

double stepX = x / (childrenCount - 1);

var children = tree.Children(n).ToList<INode>();

int j = 0;

for (int i = childrenCount / 2; i > 0; i--)

{

CalculateLoc(children[j], x - i \* stepX, y - 2.0);

CalculateLoc(children[childrenCount - j - 1], x + i \* stepX, y - 2.0);

j++;

}

CalculateLoc(children[childrenCount / 2], x, y - 2.0);

}

}

if (!locations.ContainsKey(n))

locations.Add(n, new Point(x, y));

}

* 1. **Описание функционирования программы**
     1. **Библиотека MGraph**

Библиотека, содержащая описание и интерфейсы графовых моделей. Список классов и интерфейсов с их описаниями: [См. Приложение 2].

* + 1. **Библиотека MGraphLayout**

Библиотека, в которой реализованы алгоритмы укладки и функции-помощники для указания атрибутов узлов и ребер. Список классов и интерфейсов с их описаниями: [См. Приложение 3].

* + 1. **Visio Add-in PMVisualizer**

Модуль расширения MS Visio 2010, содержит в себе дизайнер ленты плагина, классы, отвечающие за основную логику работы плагина, классы рендереры, рисующие объекты и основной класс PMVisualizer, который связывает объектную модель, представленную типом из MGraph с объектом-рендерером.

* 1. **Обоснование выбора алгоритма решения задачи**

В ходе выполнения курсовой данные алгоритмы были выбраны из следующих соображений: простота реализации, быстрое время выполнения (первые два).

* 1. **Входные и выходные данные**

Входными данными для программы служат объектные модели структур данных, представленные типами языка C#. Объектные модели могут быть получены в ходе работы программы VTMine for Visio или созданы в автоматическом режиме для тестирования приложения.

Структуры данных описаны в библиотеке MGraph. При этом их набор может быть расширен путем наследования и реализации основных интерфейсов.

Выходные данные представлены в виде страницы документа Microsoft Visio с расположенными на ней фигурами, набор которых отображает объектную модель.

* 1. **Состав технических и программных средств**

Для работы программы необходим следующий набор программных и технических средств:

1. Компьютер с Windows 7 или более поздней версией
2. Оперативная память: 256 МБ или более
3. Не менее 2 ГБ свободного места на жестком диске
4. Разрешение экрана не менее 1024\*768
5. Мышь или трекпад
6. установленный Microsoft .NET Framework 4.0;
7. Установленная программа Microsoft Visio 2010 или более поздняя версия
8. Установленное расширение MS Visio VTMine for Visio версии 0.1 или более поздней
9. **ОЖИДАЕМЫЕ ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ**

Быстрый поиск в сети Интернет на момент написания плагина не выявил приложения, которое бы позволяло работать с графовыми моделями Process Mining в среде MS Visio.

Так как плагин используется для работы в VTMine for Visio и представляет собой его часть, то он распространяется в соответствии с требованиями, предъявленными разработчиками VTMine.

# ПРИЛОЖЕНИЕ 1

**СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ**

1. Book on Graph Drawing by G. Di Battista, P. Eades, R. Tamassia, and I. G. Tollis, ISBN 0-13-301615-3, Prentice Hall
2. Graph Drawing Tutorial [Электронный ресурс] //URL:   
   <http://graphdrawing.org/literature/gd-constraints.pdf>
3. Circular Drawing of Graphs [Электронный ресурс] //URL: <http://www.csd.uoc.gr/~hy583/reviewed_notes/circular.pdf>
4. Create VSTO Add-ins for Office by using Visual Studio [Электронный ресурс] //URL: <https://msdn.microsoft.com/ru-ru/library/jj620922.aspx>

# ПРИЛОЖЕНИЕ 2

**ОПИСАНИЕ БИБЛИОТЕКИ MGRAPH**

|  |  |
| --- | --- |
| Интерфейс / Класс | Назначение |
| ILabeledObject | Интерфейс, описывающий объект, у которого есть надпись или текст. |
| IVertex | Интерфейс, описывающий вершину |
| IEdge<TVertex> | Интерфейс для ребра между двумя вершинами обобщенного типа TVertex |
| IGraph<TVertex, TEdge> | Основной интерфейс графа |
| INode | Интерфейс узла – вершины, у которой указан родитель |
| IMutableGraph<TVertex, TEdge> | Граф, который может быть очищен |
| IVertexSet<TVertex> | Множество вершин |
| IEdgeSet<TEdge> | Множество граней |
| IVertexAndEdgeSet<TVertex, TEdge> | Множество граней и вершин – базовое представление графа |
| IMutableVertexSet<TVertex> | Множество вершин с возможностью добавления и удаления вершины |
| IMutableEdgeSet<TEdge> | Множество ребер с методами добавления и удаления ребра |
| IMutableVertexAndEdgeSet<TVertex, TEdge> | Граф, в который можно добавлять вершины и грани или же удалять их |
| IDirectedGraph<TVertex, TEdge> | Интерфейс, описывающий ориентированный граф |
| IUndirectedGraph<TVertex, TEdge> | Интерфейс неориентированного графа |
| IHierarchy<TVertex, TEdge> | Интерфейс для описания ацикличного графа |
| IFlow<TVertex, TEdge> | Интерфейс для транспортной сети |
| ITree<TVertex, TEdge> | Интерфейс для дерева |
| Lable | Метка-надпись у объекта |
| SimpleVertex | Вершина графа |
| Edge<TVertex> | Ребро графа |
| UndirectedEdge<TVertex> | Ребро неориентированного графа, отличается методом ToString(); |
| Node | Узел дерева |
| FlowNode | Узел в транспортной сети |
| UndirectedGraph<TVertex, TEdge> | Неориентированный граф |
| DirectedGraph<TVertex, TEdge> | Ориентированный граф |
| Tree | Класс, представляющий дерево |
| Flow<TVertex, TEdge> | Класс, представляющий транспортную сеть |

Таблица 1. Описание классов и интерфейсов MGraph и их функционального назначения

# ПРИЛОЖЕНИЕ 3

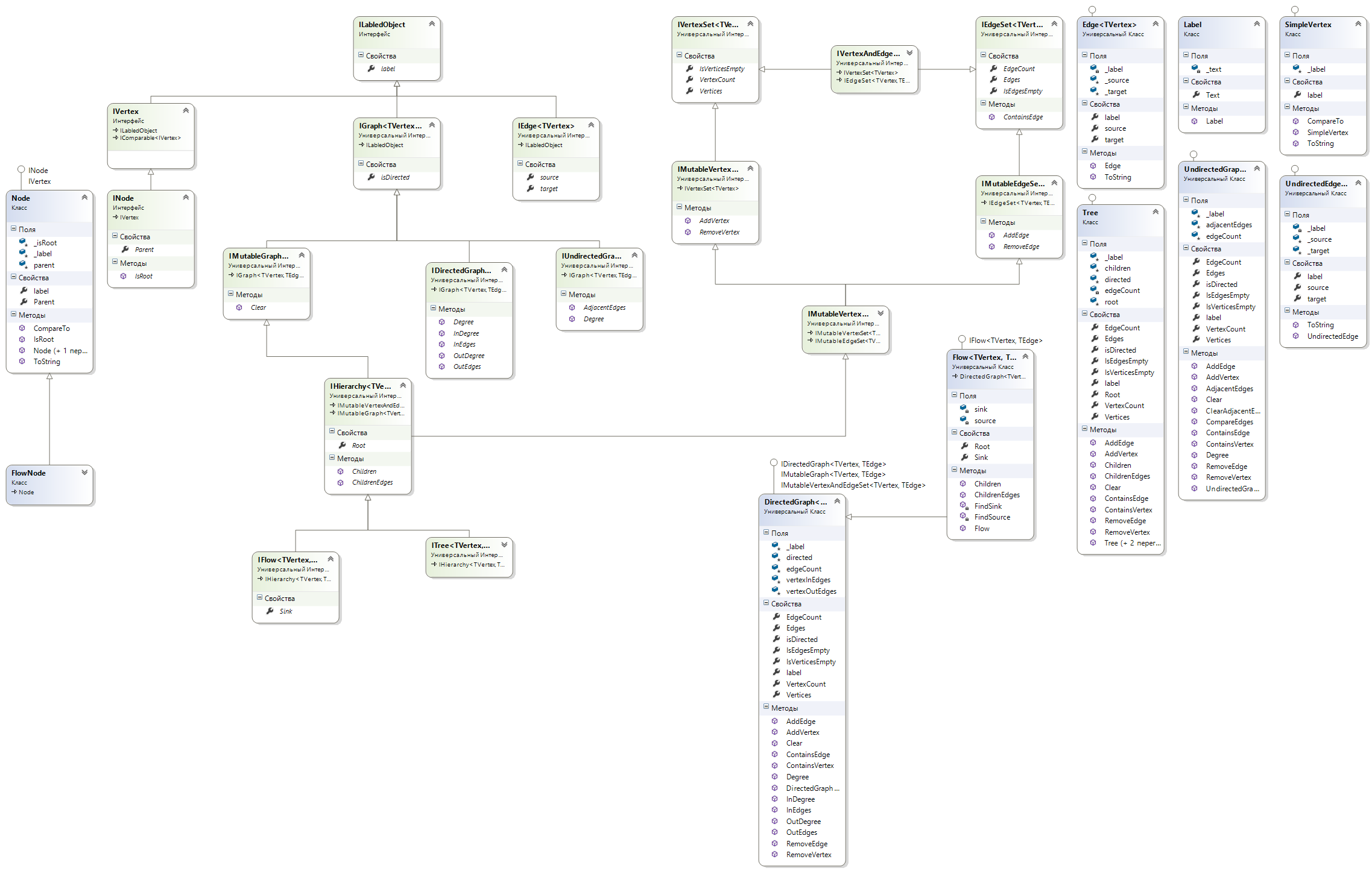
**ОПИСАНИЕ БИБЛИОТЕКИ MGRAPHLAYOUT**

Таблица 2. Описание содержимого библиотеки MGraphLayout

|  |  |
| --- | --- |
| Интерфейс / Класс | Назначение |
| IAlgorithm | Интерфейс алгоритма компоновки графа |
| SimpleTreeAlgo | Простой рекурсивный алгоритм расположения дерева на плоскости |
| CircularAlgo<TVertex> | Circular Algorithm |
| IWrappedGraph<TGraph> | Интерфейс для «оберток» графа |
| IWrappedVertex<TVertex> | Интерфейс для «обертки» вершины (напр. Атрибутами) |
| ITagged<TTag> | Объект с тэгом, отличным от Label |
| GraphGenerator | Предоставляет функции для генерирования графов с произвольным расположением вершин |
| WeightedEdge<TEdge> | Взвешенное ребро |
| PointedVertex<TVertex> | Вершина, которой сопоставлена некоторая точка. Может быть использована в некоторых частных случаях |
| PointGraph<TVertex, TEdge, TGraph> | Граф, вершины которого представлены класом PointedVertex |
| Wrappers | Предоставляет набор методов для обертки вершин, граней, графов атрибутами |
| TaggedVertex<TVertex> | Вершина с тэгом |
| AttributedGraph<TGraph> | Граф, элементы которого имеют специальные атрибуты (размер, цвет и т.д.) |
| VertexSize | Структура. Содержит размеры узла |
| EdgeAttr | Структура. Атрибуты грани |
| LabelAttr | Структура. Атрибуты метки |
| VertexAttr | Структура. Атрибуты вершины |
| LabelColor | Цвет метки |
| VertexShape | Фигура, представленная вершиной |
| EdgeDirection | Направление грани |
| VertexColor | Цвет вершины |
| LegendPosition | Расположение легенды |
| EdgeArrowType | Тип стрелки грани |
| LabelFontType | Тип шрифта метки |
| EdgeColor | Цвет грани |

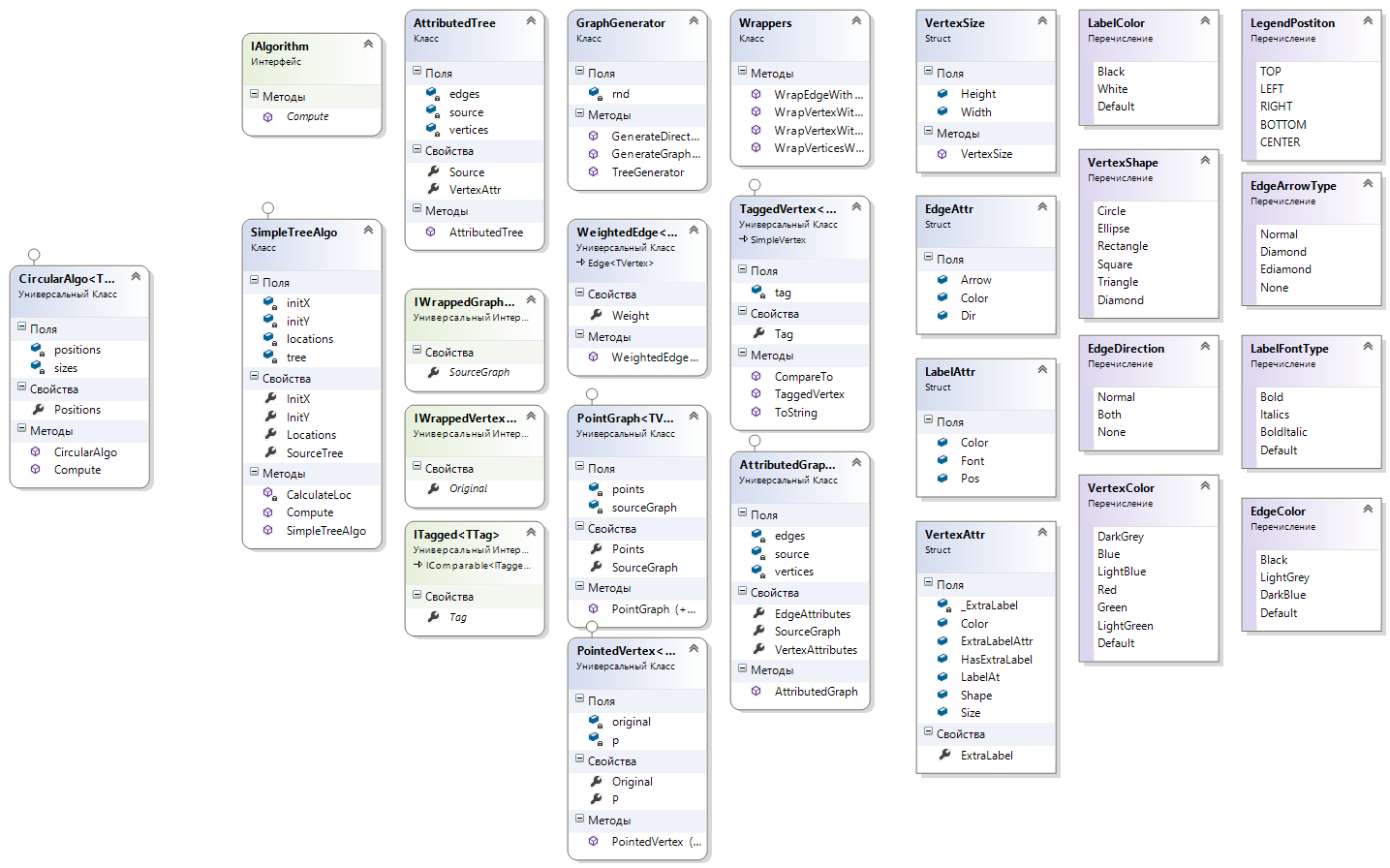
# ПРИЛОЖЕНИЕ 4

**СХЕМА КЛАССОВ MGRAPH**



# ПРИЛОЖЕНИЕ 5

**СХЕМА КЛАССОВ MGRAPHLAYOUT**



# ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | |
| Номера листов (страниц) | | | | | Всего листов (страниц в докум.) | № документа | Входящий № сопроводительного докум. и дата | Подп. | Дата |
| Изм. | Измененных | Замененных | Новых | Аннулированх |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |