ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ

БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ –

ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

НИЖЕГОРОДСКИЙ ФИЛИАЛ

ФАКУЛЬТЕТ БИЗНЕС-ИНФОРМАТИКИ И

ПРИКЛАДНОЙ МАТЕМАТИКИ

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**к курсовой работе:**

**«**Визуальное редактирование графов**»**

**Выполнил:** студент группы 10ПМИ2

Савиных А.Л.

**Руководитель:** Логинов О.В.

Нижний Новгород

2013

Оглавление

[Введение 3](#_Toc352097305)

[История 4](#_Toc352097306)

[Аналоги 5](#_Toc352097307)

[Задача 6](#_Toc352097308)

[Требования 6](#_Toc352097309)

[Разработка 7](#_Toc352097310)

[Средства разработки 7](#_Toc352097311)

[Архитектура 7](#_Toc352097312)

[Реализация 9](#_Toc352097313)

[Формат документов 11](#_Toc352097314)

[Эксплуатация 12](#_Toc352097315)

[Добавление вершины 12](#_Toc352097316)

[Редактирование подписи 15](#_Toc352097317)

[Просмотр графа 15](#_Toc352097318)

[Удаление вершины 15](#_Toc352097319)

[Соединение вершин рёбрами 16](#_Toc352097320)

[Удаление рёбер 17](#_Toc352097321)

[Сохранение графа 17](#_Toc352097322)

[Загрузка графа 18](#_Toc352097323)

[Изменение положения вершин в документе 18](#_Toc352097324)

[Установка 18](#_Toc352097325)

[Тестирование 18](#_Toc352097326)

[Функционал, подлежащий тестированию: 18](#_Toc352097327)

[Ссылки 19](#_Toc352097328)

[Подход к тестированию 19](#_Toc352097329)

[Шаблон теста 19](#_Toc352097330)

[Тесты 20](#_Toc352097331)

[Заключение 25](#_Toc352097332)

# Введение

Темой данной курсовой работы является визуальное редактирование графов.

Графы используются для представления различных задач и моделирования различных явлений и объектов. Практически любая задача, в которой важны объекты и связи между ними, может быть представлена в виде графа. Вот несколько примеров:

* Идеи, приходящие сотрудникам во время совещания, зарисовываются в виде графов, идеи – вершины графа, связанные по смыслу идеи соединяют линиями – рёбрами графа
* Веб-сайты с различными страницами и ссылками между ними могут быть представлены в виде графов. Страницы – вершины графа, ссылки на другие страницы – рёбра графа.
* Компьютерные сети могут быть представлены в виде графов
* Карты земной поверхности с проложенными по ней линиями коммуникации могут быть представлены в виде графов

Графы воспринимаются проще и эффективнее когда представлены визуально. В то же время, это не единственный способ их представления. Например:

Таблица

|  |  |
| --- | --- |
| Визуальное представление | Текстовое представление |
|  | Вершины: 2, 3, 1, 4, 5  Связи: 3 соединено с 1, 1 соединено с 2, 1 соединено с 4, 1 соединено с 5 |

Для более эффективной и удобной с точки зрения программиста обработки графы как структуры данных хранятся часто в виде наборов вершин и связей между ними, имеющем много общего с тем, как представлено в колонке «текстовое представление» в таблице Таблица 1 выше. В тех случаях, когда, например, человеку необходимо ввести подготовить исходные данные в виде являющиеся графом для последующей обработки программой или же просто для хранения в цифровом виде, требуется визуальное представление графа и его последующее преобразование в пригодный для программной обработки формат.

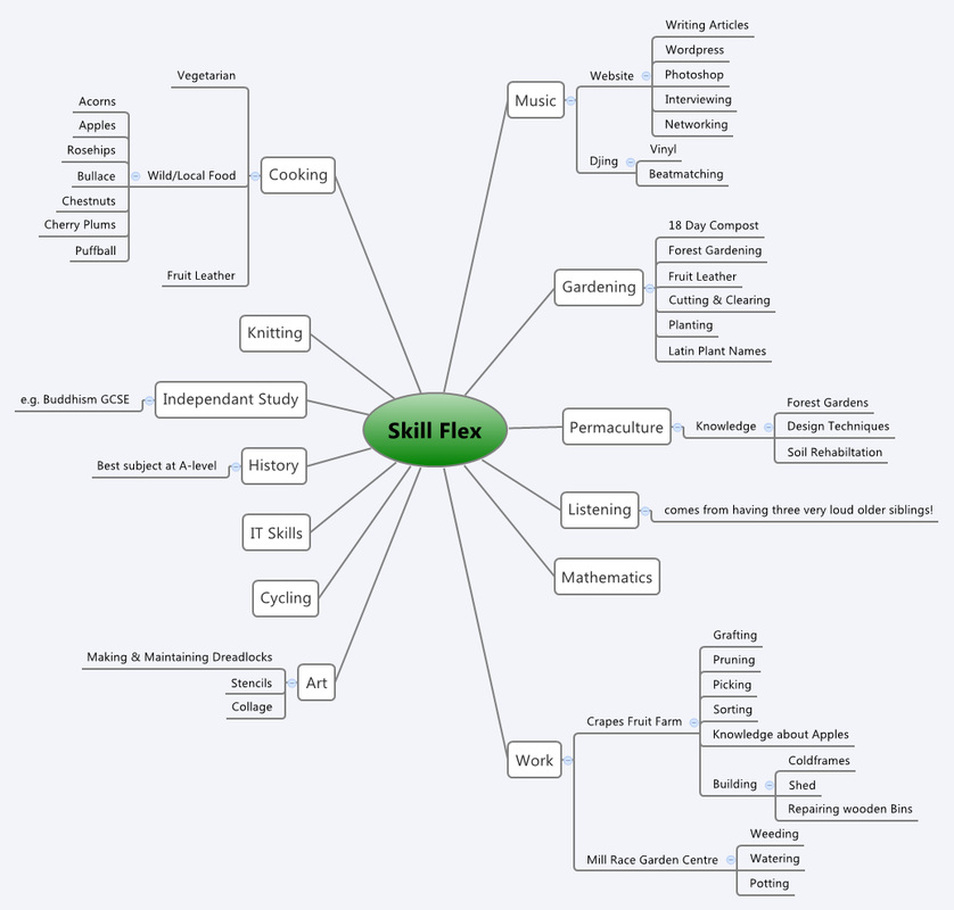
Программа, разработанная в рамках курсовой работы, позволяет эффективно редактировать графы, хранить графы в виде файлов, а так же изменять их в дальнейшем и обрабатывать с помощью иных программ.

# История

Родоначальником теории графов считается Леонард Эйлер. В 1736 году в одном из своих писем он формулирует и предлагает решение задачи о семи кёнигсбергских мостах, ставшей впоследствии одной из классических задач теории графов. Интерес к проблемам теории графов возродился около середины прошлого столетия. Естественные науки оказали свое влияние на это благодаря исследованиям электрических цепей, моделей кристаллов и структур молекул. Развитие формальной логики привело к изучению бинарных отношений в форме графов. Большое число популярных головоломок подавалось формулировкам непосредственно в терминах графов, и это приводило к пониманию, что многие задачи такого рода содержат некоторое математическое ядро, важность которого выходит за рамки конкретного вопроса.

# Аналоги

Существует множество программ выполняющих аналогичные задачи. Вот описание одной из них в качестве примера: приложение XMind имеет платную и бесплатную версии, предоставляет возможности для визуального редактирования графов. На рисунке Рисунок 1 показана схема-граф, созданная с помощью приложения XMind.



Рисунок

XMind может подбирать позиции узлов таким образом, чтобы они выглядели естественно. Так, если пользователь явно не указывает расположение дочерних узлов, они располагаются по кругу возле центрального узла в случае узлов, непосредственно связанных с центральным узлом («Skell Flex» на рисунке Рисунок 1) либо на свободном от других узлов и связей месте в случае более удалённых от центрального узла узлов. XMind сохраняет документы в формате XML, так что они могут быть использованы другими приложениями. При создании документа в XMind возможно изменять цвет, размер и начертание шрифтов подписей узлов, добавлять изображения к узлам, изменять положение узлов, изменять фигуры обрамляющие узлы, добавлять дополнительный текст к узлам, а также цвет и формат прочих элементов.

# Задача

В ходе подготовки к курсовой работе была сформулирована следующая задача: создать приложение для визуального редактирования графов, обладающее некоторыми особенностями. Одной из особенностей должна быть возможность сохранения графов в формат, пригодный для использования другими приложениями.

# Требования

Все требования разделены на две группы: обязательные требования, отмеченные в данном документе закрашенными кружками, и необязательные требования, отмеченные в данном документе незакрашенными кружками.

|  |  |
| --- | --- |
| Приоритет и название требования | Номер требования |
| * Возможность добавления новых узлов в редактируемый граф | 1 |
| * Возможность присоединения подписей к узлам графа | 2 |
| * Возможность редактирования подписи к определённому узлу графа | 3 |
| * Возможность просмотра графов с помощью приложения | 4 |
| * Возможность просмотра графа по частям, прокручивая документ, в случае если граф имеет большую площадь | 5 |
| * Возможность удаления существующих узлов из редактируемого графа | 6 |
| * Возможность соединения узлов связями | 7 |
| * Возможность удаления связей между узлами | 8 |
| * Возможность сохранения редактируемого графа в файл | 9 |
| * Возможность загрузки графа из файла, в который он ранее был сохранён тем же приложением | 10 |
| * Возможность изменения положения узлов в документе | 11 |
| * Возможность работы с несколькими графами одновременно так, чтобы открытые графы были представлены в интерфейсе пользователя вкладками | 12 |
| * Возможность присоединения меток к узлам графа | 13 |
| * Возможность управления приложением с помощью жестов | 14 |

Таблица 1

# Разработка

## Средства разработки

Для написания приложения была выбрана среда разработки Visual Studio Express 2012 для Windows, язык программирования C#. Упомянутая среда разработки является бесплатной для некоммерческого использования. Для работы с отладочными сообщениями использована библиотека NLog. Для хранения исходного кода проекта и документации выбрана система контроля версий Git. Для работы с системой контроля версий использовано приложение SmartGit, представляющее собой графический интерфейс для Git и некоторых других систем контроля версий. Исходный код проекта хранится также на сайте GitHub, предоставляющем возможность хранить репозитории Git с исходным кодом онлайн. Благодаря этому в случае отказа жёсткого диска компьютера, на котором разрабатывался проект, можно было бы скачать репозиторий с сайта GitHub и продолжить работу. Также использованы различные средства и библиотеки, входящие в состав Visual Studio 2012: Windows Presentation Foundation для разработки пользовательского интерфейса, система управления пакетами NuGet, средства для работы с XML и прочее.

## Архитектура

Так как проект разрабатывался на языке программирования C#, использован объектно-ориентированный подход, основными строительными блоками проекта являются классы. Для удобства все классы разбиты на несколько категорий. Ниже слово проект используется в смысле проекта VisualStudio как объединения классов, слово решение используется в смысле решения VisualStudio как объединения нескольких проектов.

На рисунке «Рисунок 2: Структура решения» представлена структура решения.

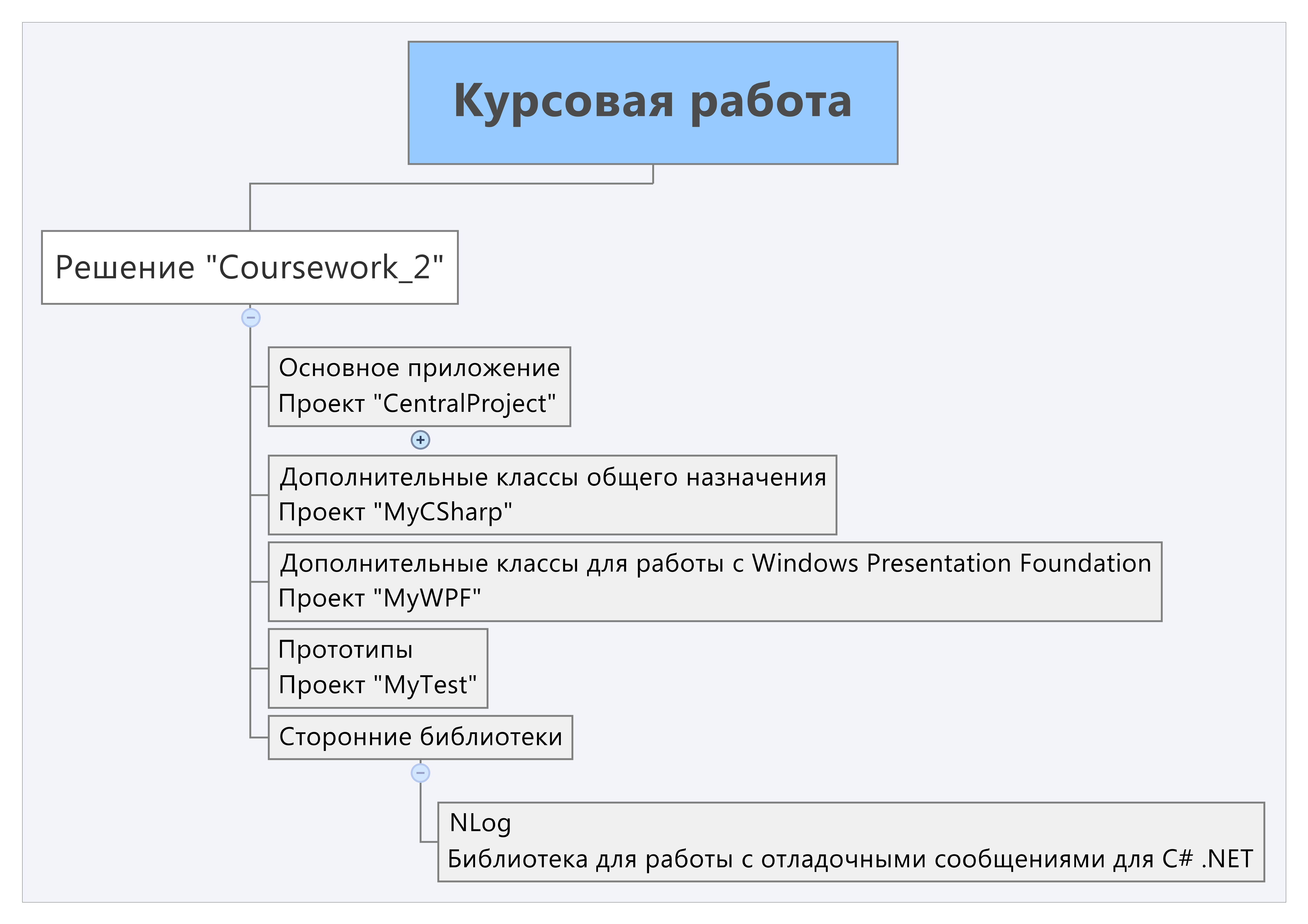


Рисунок : Структура решения

В проекте «CentralProject» собраны классы, непосредственно относящиеся по смыслу к приложению для визуального редактирования графов. Все классы в основном проекте тесно связаны между собой и не обладают большой самостоятельностью. В остальных проектах содержатся различные вспомогательные классы. В приложении явно не создаются никакие дополнительные потоки, большую роль играет Windows Presentation Foundation, который и определил архитектуру приложения, код приложения привязан к событиям Windows Presentation Foundation и выполняется полностью в том же потоке, что и обработчик этих событий, отвечающий за пользовательский интерфейс.

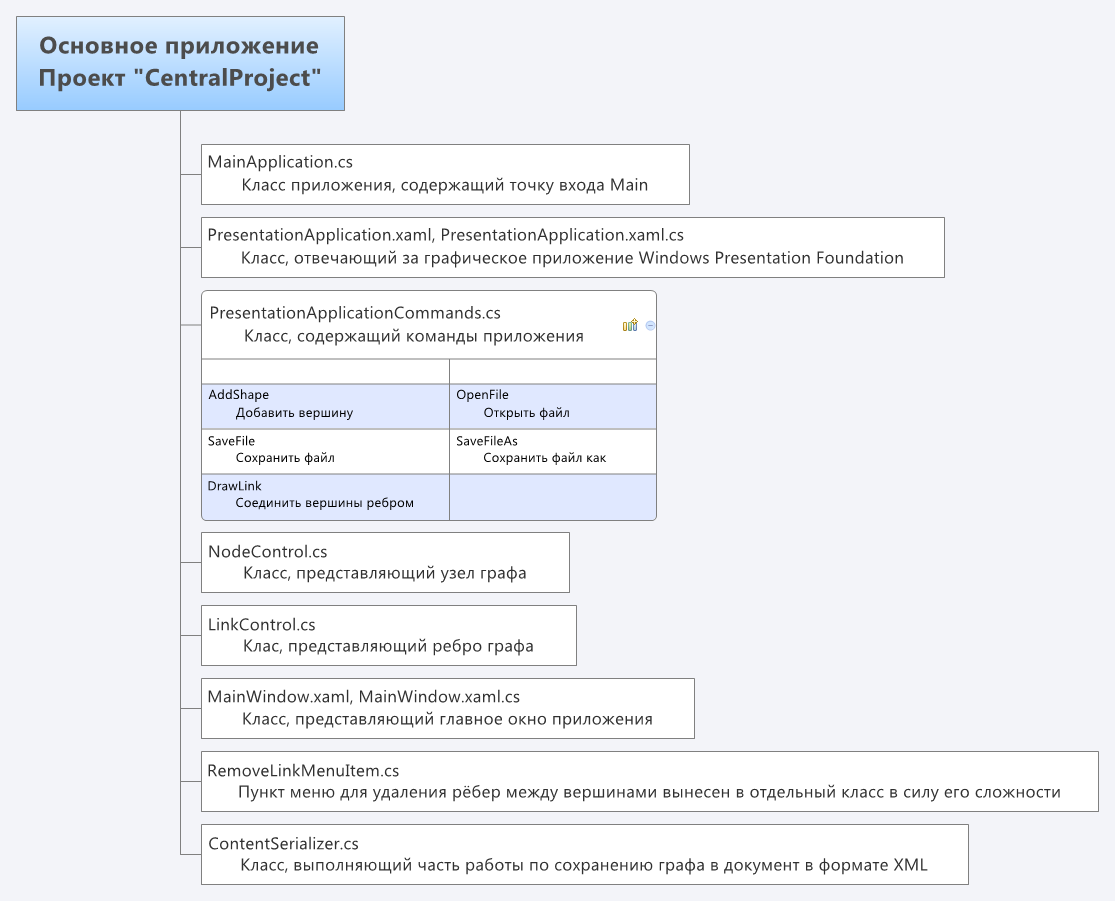


Рисунок : CentralProject

Некоторые классы определены как с помощью исходного кода на языке C# в файлах с расширением .cs, так и с помощью исходного кода на языке XAML в файлах с расширением .xaml, которые используются главным образом при разработке приложений Windows Presentation Foundation для определения пользовательского интерфейса.

## Реализация

Условно приложение можно разбить на две основных подсистемы:

* Визуальное редактирование
* Сохранение и загрузка в формате XML

В то же время, некоторые классы выполняют функции из обеих областей. Например, классы NodeControl и LinkControl относятся в первую очередь к системе визуального редактирования, однако, содержат функции для сохранения своих атрибутов в элемент XML. В то же время, они не способны сохранить граф в файл, так как являются его частью. За присвоение уникальных идентификаторов вершинам и рёбрам, а также за сохранение и восстановление графа как целого отвечает класс ContentSerializer.

В самом начале своей работы программа передаёт управление приложению Windows Presentation Foundation:

var presentation = new PresentationApplication();

presentation.Run();

Метод Run вызывает обработку событий пользовательского интерфейса и работает, пока главное приложения окно не будет закрыто. Затем графическое приложение будет уничтожено.

В файле PresentationApplication.xaml указан класс главного окна приложения:

<Application x:Class="Coursework\_2.PresentationApplication"

… … …

**StartupUri="MainWindow.xaml"**

>

На снимке экрана «Рисунок 4: Окно приложения» представлено главное окно приложения с открытым документом.

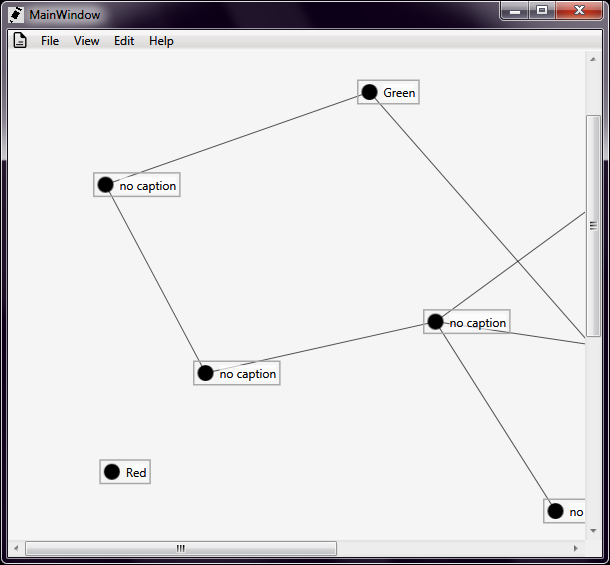


Рисунок : Окно приложения

В файле разметки MainWindow.xaml определены пункты главного меню, пиктограммы для них, а также область документа.

Ниже представлен фрагмент кода из файла MainWindow.xaml:

<MenuItem Header="File" Icon="{StaticResource Image\_File}">

<MenuItem

Header="Open file..."

x:Name="OpenFileMenuItem"

Icon="{StaticResource Image\_OpenFile}"

/>

<MenuItem

Header="Save file"

x:Name="SaveFileMenuItem"

Icon="{StaticResource Image\_SaveFile}"

/>

… … …

Ниже представлен фрагмент кода, привязывающий команды и их обработчики к пунктам главного меню окна приложения.

protected void BindCommands()

{

BindCommand(OpenFileMenuItem, Commands.OpenFile, UserOpenFile);

BindCommand(SaveFileMenuItem, Commands.SaveFile, UserSaveFile);

BindCommand(SaveAsFileMenuItem, Commands.SaveAsFile, UserSaveAsFile);

BindCommand(AddItemMenuItem, Commands.AddShape, UserAddItem);

BindCommand(AddLinkMenuItem, Commands.DrawLink, UserAddLink);

}

protected void BindCommand(MenuItem item, RoutedUICommand command,

ExecutedRoutedEventHandler handler)

{

CommandManager.RegisterClassCommandBinding(

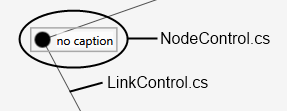
this.GetType(),

new CommandBinding(command, handler)

);

item.Command = command;

}



Рисунок

На рисунке «Рисунок 5» показано, каким видимым элементам управления в пользовательском интерфейсе соответствуют классы NodeControl и LinkControl.

## Формат документов

Ниже представлен фрагмент документа в формате XML, полученного с помощью программы визуального редактирования графов.

<?xml version="1.0" encoding="utf-8"?>

<content>

<NodeControl Text="no caption" Left="629" Top="731" id="1" />

<NodeControl Text="no caption" Left="374" Top="617" id="2" />

<NodeControl Text="no caption" Left="703" Top="536" id="3" />

<NodeControl Text="no caption" Left="535" Top="551" id="4" />

<NodeControl Text="no caption" Left="140" Top="737" id="5" />

<NodeControl Text="Some Other Node" Left="185" Top="413" id="6" />

<NodeControl Text="no caption" Left="875" Top="430" id="7" />

<NodeControl Text="no caption" Left="415" Top="361.60526315789468" id="8" />

<NodeControl Text="SomeNode" Left="85" Top="224.60526315789471" id="9" />

<NodeControl Text="no caption" Left="677" Top="169.60526315789471" id="10" />

<NodeControl Text="Green" Left="349" Top="132" id="11" />

<NodeControl Text="Red" Left="91.331466965285514" Top="511.6973684210526" id="12" />

<LinkControl Node1="11" Node2="9" id="13" />

<LinkControl Node1="11" Node2="3" id="14" />

<LinkControl Node1="10" Node2="8" id="15" />

<LinkControl Node1="8" Node2="6" id="16" />

<LinkControl Node1="9" Node2="6" id="17" />

<LinkControl Node1="8" Node2="4" id="18" />

<LinkControl Node1="7" Node2="8" id="19" />

<LinkControl Node1="2" Node2="5" id="20" />

</content>

Содержимое документа заключено в корневом элементе XML-документа “content”. В нём содержатся элементы “NodeControl”, соответствующие вершинам графа, и элементы “LinkControl”, соответствующие рёбрам графа. Каждый элемент “NodeControl” содержит следующие атрибуты:

* “Text”: подпись вершины
* “Left”, “Top”: расстояния от левой и верхней границ документа, выраженные в единицах измерения расстояний Windows Presentation Foundation, соответствующих одному пикселю на экране с разрешением 96 пикселей на дюйм.
* “id”: уникальный в рамках одного документа номер элемента

Каждый элемент LinkConrol соответствует ребру графа и содержит следующие атрибуты:

* “Node1” и “Node2”: уникальные номера вершин, связываемых данным ребром и содержащиеся в их атрибутах “id”.
* “id”: уникальный в рамках одного документа номер элемента

Таким образом, из документа, генерируемого приложением визуального редактирования графов можно полностью восстановить структуру графа и, при необходимости, его внешний вид при редактировании.

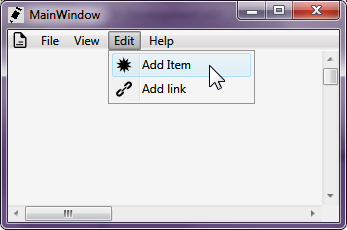
В процессе реализации приложения были удовлетворены все обязательные требования, описанные выше в секции «Требования», но не было удовлетворено ни одно из необязательных требований.

# Эксплуатация

Ниже приведены инструкции для выполнения различных действий в приложении в том порядке, в котором приведены соответствующие этим действиям требования.

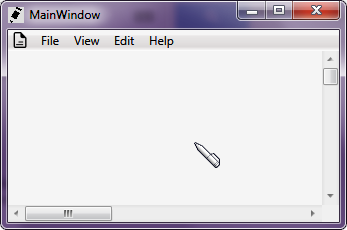
## Добавление вершины

* Выбрать пункт меню Edit → C:\Users\hinst\Docs\Pro\Coursework_2\CentralProject\Images\NewItemBlack16.png Add Item, как показано на рисунке «Рисунок 6»



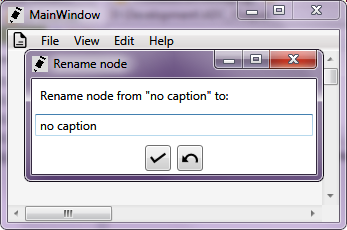
Рисунок

При этом если навести мышь на область документа, указатель изменится, как показано на рисунке «Рисунок 7».



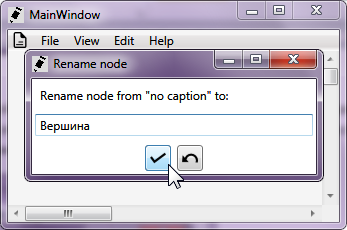
Рисунок

* Нажать левой кнопкой мыши в той области документа, в которой необходима новая вершина графа. При этом появится окно, в котором будет предложено ввести имя для вершины, как показано на рисунке «Рисунок 8»
* Для отмены добавления вершины на этом шаге следует нажать правой кнопкой мыши в тот момент, когда указатель принял форму авторучки. При этом указатель примет первоначальную форму.



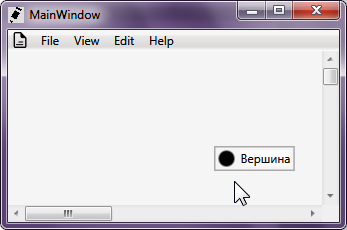
Рисунок

* Ввести имя вершины и нажать кнопку C:\Users\hinst\Docs\Pro\MyCSharp\MyWPF\Images\SubmitBlack16.png для подтверждения, как показано на рисунке «Рисунок 9», либо кнопку C:\Users\hinst\Docs\Pro\MyCSharp\MyWPF\Images\UndoBlack16.png для отмены.



Рисунок

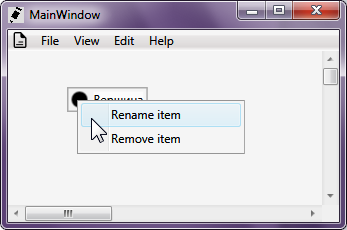
* В случае подтверждения в документе появляется новая вершина, как показано на рисунке «Рисунок 10».



Рисунок

## Редактирование подписи

* Для того чтобы отредактировать подпись вершины графа, следует вызвать её контекстное меню путём нажатия левой кнопкой мыши по пиктограмме вершины графа C:\Users\hinst\Docs\Pro\Coursework_2\CentralProject\Images\BulletShapeBlack16.png и выбрать пункт меню Rename item, как показано на рисунке «Рисунок 11».

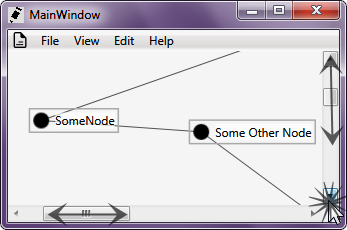


Рисунок

* При этом отобразится окно, в котором можно изменить имя вершины и затем подтвердить изменение [C:\Users\hinst\Docs\Pro\MyCSharp\MyWPF\Images\SubmitBlack16.png] либо отменить изменение [C:\Users\hinst\Docs\Pro\MyCSharp\MyWPF\Images\UndoBlack16.png], при этом в последнем случае, что бы ни находилось в поле ввода для подписи вершины в окне редактирования имени вершины, окно редактирования подписи вершины скроется, и подпись вершины останется прежней.

## Просмотр графа

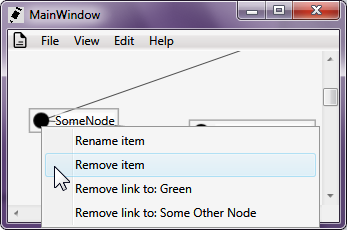
* Чтобы просматривать различные части документа с графом, следует воспользоваться полосами прокрутки



Рисунок

## Удаление вершины

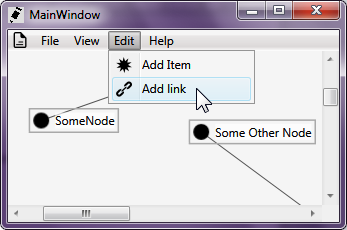
* Удалить вершину можно с помощью пункта “Remove item” контекстного меню вершины, как показано на рисунке «Рисунок 13».



Рисунок

## Соединение вершин рёбрами

* Выбрать в главном меню пункт Edit → C:\Users\hinst\Docs\Pro\Coursework_2\CentralProject\Images\LinkBlack16.png Add link (Рисунок 14).

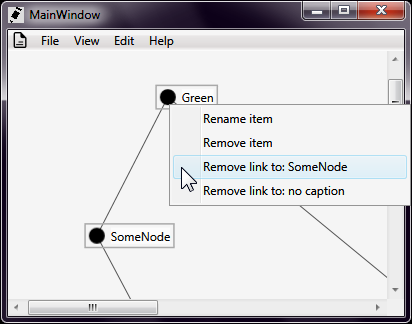


Рисунок

* Последовательно щёлкнуть мышью пиктограммы вершин графа, которые следует соединить C:\Users\hinst\Docs\Pro\Coursework_2\CentralProject\Images\BulletShapeBlack16.png.
* Отменить операцию можно нажатием правой кнопки мыши.

## Удаление рёбер

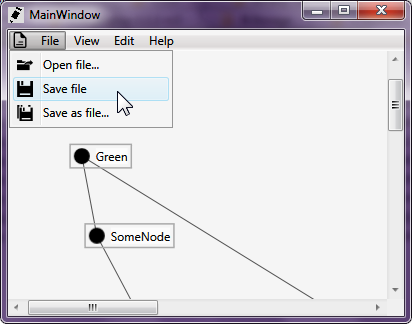
* Чтобы удалить ребро следует вызвать контекстное меню одной из вершин, лежащей на этом ребре и выбрать пункт меню “Remove link to:”, в котором указана подпись противоположенной вершины, связь с которой следует удалить.



Рисунок

## Сохранение графа

* Для того чтобы сохранить граф в файл, следует выбрать пункт меню File → C:\Users\hinst\Docs\Pro\Coursework_2\CentralProject\Images\SaveFileBlack16.png Save file («Рисунок 17»)



Рисунок

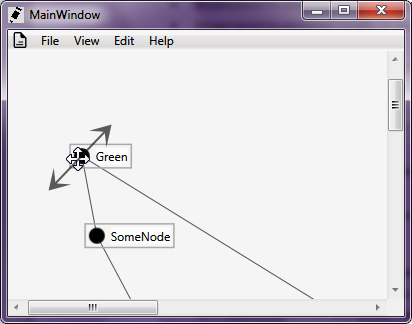
* После этого отобразится диалог сохранения файла, в котором следует указать имя и расположение сохраняемого документа
* Если имя файла было указано, и документ был сохранён, то при последующих нажатиях диалог выбора имени файла больше не будет отображаться, и выбранный ранее файл будет записан заново, при этом будет сохранено текущее содержимое документа. Чтобы сохранить содержимое под другим именем, следует использовать пункт меню Edit → C:\Users\hinst\Docs\Pro\Coursework_2\CentralProject\Images\SaveAsFileBlack16.png Save File As…

## Загрузка графа

* Чтобы открыть ранее сохранённый граф, следует выбрать пункт меню File → C:\Users\hinst\Docs\Pro\Coursework_2\CentralProject\Images\OpenFileBlack16.png Open File…
* В появившемся диалоге открытия файла следует выбрать файл, в который был ранее сохранён граф.

## Изменение положения вершин в документе

* Для того чтобы изменить положение вершины в документе следует перетащить её, удерживая левую кнопку мыши (Рисунок 16).



Рисунок

## Установка

Для работы приложения требуются исполняемые и другие файлы, получаемые в процессе её сборки, то есть, все файлы в каталоге вывода Visual Studio: “Coursework\_2\CentralProject\bin\Debug” либо “Coursework\_2\CentralProject\bin\Release”. Копируя их можно переносить приложение на различные компьютеры. Для запуска приложение следует использовать исполняемый файл CentralProject.exe. Для работы программы необходима среда выполнения .NET framework 4.0.

# Тестирование

## Функционал, подлежащий тестированию:

См. таблицу «Таблица 1», все обязательные требования.

* Добавление новых узлов в редактируемый граф
* Присоединение подписей к узлам графа
* Редактирования подписи к определённому узлу графа
* Просмотр графов с помощью приложения
* Просмотр графа по частям, прокручивая документ, в случае если граф имеет большую площадь
* Удаление существующих узлов из редактируемого графа
* Соединение узлов связями
* Удаление связей между узлами
* Сохранения редактируемого графа в файл
* Загрузка графа из файла
* Перемещение узлов

## Ссылки

Действия, необходимые для использования того или иного функционала, описаны в разделе «Эксплуатация», поэтому в тестах вместо того, чтобы подробно описывать действия, будут лишь кратко названы соответствующие функции. Например, «добавить вершину» вместо «Выбрать пункт меню Edit → Add Item, в появившемся окне ввести имя…»

## Подход к тестированию

Основная задача тестирования – выявить как можно больше критических ошибок, возникающих при использовании базового функционала. В первую очередь следует проверять стандартные тесты. В то же время, следует понизить приоритет нестандартных тестов, таких как, например, попытка загрузить в приложение веб-страницы вместо XML-документов, сгенерированных приложением и соответствующих его формату.

## Шаблон теста

|  |  |
| --- | --- |
| Номер | В десятичной системе счисления |
| Название |  |
| Приоритет | Низкий, высокий |
| Тип | Позитивный, негативный |
| *Окружение* | Указывать если необходимо |
| Система | Должна быть указана одна из двух основных систем приложения визуальное редактирование либо XML |
| Номер требования | Указывать если тест относится к конкретному требованию |
| *Цель* | Указывать если необходимо, иначе целью является проверка требования, если указан номер требования, либо корректности работы подсистемы, если указана подсистема |
| Начальные условия |  |
| Действия |  |
| Ожидаемый результат |  |
| *Конечное состояние* |  |

Замечание к полю «конечное состояние»: если не указано, то подразумевается «приложение остаётся в работоспособном состоянии», то есть, возможно приступить к любому следующему тесту, не запуская приложение заново. Если же приложение не сохраняет работоспособность, то есть, «падает», или перестаёт отвечать, тест считается частично проваленным даже если все предыдущие действия выполнены успешно и ожидаемый результат достигнут. Такие случаи следует отмечать отдельно, например, «тест завершился успешно, сразу после чего произошла критическая ошибка».

## Тесты

|  |  |
| --- | --- |
| Тест | |
| Номер | **1** |
| Название | Добавление вершины |
| Приоритет | Высокий |
| Тип | Позитивный |
| *Окружение* |  |
| Система | Визуальное редактирование |
| Номер требования | 1, 2 |
| *Цель* |  |
| Начальные условия | Приложение запущено |
| Действия | Добавить несколько вершин с произвольными подписями в различные области документа. Провести хотя бы одну проверку с заданием пустой подписи вершины, хотя бы одну проверку с заданием подписи по умолчанию (не изменяя подпись) и хотя бы одну проверку с непустой подписью, отличной от подписи по умолчанию. |
| Ожидаемый результат | Вершины появляются в той области документа, в которой было совершено нажатие. Рядом с пиктограммой вершины отображается та подпись, которая была указано. |
| *Конечное состояние* |  |

|  |  |
| --- | --- |
| Тест | |
| Номер | **2** |
| Название | Отмена добавления вершины |
| Приоритет | Высокий |
| Тип | Позитивный |
| *Окружение* |  |
| Система | Визуальное редактирование |
| Номер требования | 1 |
| *Цель* |  |
| Начальные условия | Приложение запущено |
| Действия | Начать добавление вершины выбором соответствующего пункта меню. Отменить. Повторить несколько раз. Попробовать изменить содержимое поля ввода для имени вершины. Провести хотя бы одну проверку без изменения содержимого этого поля ввода и хотя бы одну проверку с изменением содержимого поля ввода. Провести проверку, когда в документе уже присутствуют несколько вершин и рёбер. |
| Ожидаемый результат | Новые вершины не появляются во всех случаях |
| *Конечное состояние* |  |

|  |  |
| --- | --- |
| Тест | |
| Номер | **3** |
| Название | Редактирование подписи вершины |
| Приоритет | Высокий |
| Тип | Позитивный |
| *Окружение* |  |
| Система | Визуальное редактирование |
| Номер требования | 3 |
| *Цель* |  |
| Начальные условия | В документе имеются вершины |
| Действия | Переименовать любую вершину. Проверить возможность задания пустой подписи. Проверить возможность отмены переименования. |
| Ожидаемый результат | Подпись переименованной вершины изменяется на заданную. Проверить, что после появления окна переименования вершины в поле ввода для подписи вершины содержится её текущая подпись. В случае задания пустого имени: после переименования пиктограмма и рамка вершины отображаются, текст подписи не виден. В случае отмены операции переименования текст подписи вершины остаётся прежним не зависимо от содержимого поля ввода имени вершины. |
| *Конечное состояние* |  |

|  |  |
| --- | --- |
| Тест | |
| Номер | **4** |
| Название | Просмотр документа |
| Приоритет | Высокий |
| Тип | Позитивный |
| *Окружение* |  |
| Система | Визуальное редактирование |
| Номер требования | 4, 5 |
| *Цель* |  |
| Начальные условия | В программе открыт документ с не менее 10 вершинами и связями |
| Действия | В случае если все вершины видны одновременно, перетащить хотя бы одну вершину за край видимой области документа. Прокручивать область просмотра документа с помощью полос прокрутки. |
| Ожидаемый результат | Вершины и связи между ними отображаются корректно. При прокрутке граф сохраняет свою структуру. Вершины и связи не изменяют своих позиций относительно друг друга и краёв документа во время прокрутки. |
| *Конечное состояние* |  |

|  |  |
| --- | --- |
| Тест | |
| Номер | **5** |
| Название | Удаление вершин |
| Приоритет | Высокий |
| Тип | Позитивный |
| *Окружение* |  |
| Система | Визуальное редактирование |
| Номер требования | 6 |
| *Цель* |  |
| Начальные условия | В открытом документе присутствует хотя бы одна вершина |
| Действия | Удалить любую вершину. Проверить в нескольких вариантах: когда в документе присутствуют другие вершины и связи, когда в документе остаётся только одна вершина, когда с вершиной связано одно или несколько рёбер, когда вершина не связана с другими вершинами ни одним ребром. |
| Ожидаемый результат | Удаляемые вершины исчезают, рёбра, соединённые с удаляемыми вершинами, исчезают, все остальные элементы документа остаются. |
| *Конечное состояние* |  |

|  |  |
| --- | --- |
| Тест | |
| Номер | **6** |
| Название | Добавление ребра |
| Приоритет | Высокий |
| Тип | Позитивный |
| *Окружение* |  |
| Система | Визуальное редактирование |
| Номер требования | 7 |
| *Цель* |  |
| Начальные условия | В открытом документе присутствуют хотя бы две вершины |
| Действия | Соединить две вершины ребром |
| Ожидаемый результат | Между выбранными вершинами появляется ребро, остальные элементы документа остаются неизменными |
| *Конечное состояние* |  |

|  |  |
| --- | --- |
| Тест | |
| Номер | **7** |
| Название | Отмена добавления ребра |
| Приоритет | Высокий |
| Тип | Позитивный |
| *Окружение* |  |
| Система | Визуальное редактирование |
| Номер требования | 7 |
| *Цель* |  |
| Начальные условия | В открытом документе присутствуют хотя бы две вершины |
| Действия | Во время добавления ребра выбрать одну из вершин, после чего отменить добавление ребра нажатием правой кнопки мыши |
| Ожидаемый результат | Между вершинами не появляется ребро, остальные элементы документа остаются неизменными |
| *Конечное состояние* |  |

|  |  |
| --- | --- |
| Тест | |
| Номер | **8** |
| Название | Удаление ребра |
| Приоритет | Высокий |
| Тип | Позитивный |
| *Окружение* |  |
| Система | Визуальное редактирование |
| Номер требования | 8 |
| *Цель* |  |
| Начальные условия | В открытом документе присутствуют хотя бы две вершины, соединённые ребром |
| Действия | Удалить ребро между двумя вершинами через контекстное меню одной из вершин |
| Ожидаемый результат | Между выбранными вершинами исчезает ребро, остальные элементы документа остаются неизменными |
| *Конечное состояние* |  |

|  |  |
| --- | --- |
| Тест | |
| Номер | 9 |
| Название | Сохранение и ручная проверка результирующего документа |
| Приоритет | Высокий |
| Тип | Позитивный |
| *Окружение* |  |
| Система | XML |
| Номер требования | 9, 10 |
| *Цель* |  |
| Начальные условия | В документе присутствуют вершины и рёбра в количестве до 10 |
| Действия | Сохранить документ в файл. Открыть файл с помощью текстового редактора. |
| Ожидаемый результат | Содержимое соответствует формату, описанному в разделе «Формат документов» |
| *Конечное состояние* |  |

|  |  |
| --- | --- |
| Тест | |
| Номер | 10 |
| Название | Загрузка документа с перезапуском приложения |
| Приоритет | Высокий |
| Тип | Позитивный |
| *Окружение* |  |
| Система | XML |
| Номер требования | 9, 10 |
| *Цель* |  |
| Начальные условия | В документе присутствуют вершины и рёбра в количестве от 10 |
| Действия | Сохранить документ в файл. Закрыть приложение. Открыть приложение. Открыть ранее сохранённый документ. |
| Ожидаемый результат | После открытия документ полностью восстанавливает тот вид, который он имел перед сохранением |
| *Конечное состояние* |  |

|  |  |
| --- | --- |
| Тест | |
| Номер | 11 |
| Название | Загрузка документа без перезапуска приложения |
| Приоритет | Высокий |
| Тип | Позитивный |
| *Окружение* |  |
| Система | XML |
| Номер требования | 9, 10 |
| *Цель* |  |
| Начальные условия | Приложение запущено, имеются различные файлы с документами, сохранённые ранее в количестве не менее трёх |
| Действия | Последовательно открыть не менее трёх различных файлов с документами |
| Ожидаемый результат | После открытия документы восстанавливают вид, который они имели перед сохранением. Если открыть документ, в то время как один из документов уже был открыт ранее, то содержимое открываемого документа полностью замещает содержимое старого документа; содержимое старого документа больше не видно. |
| *Конечное состояние* |  |

|  |  |
| --- | --- |
| Тест | |
| Номер | 12 |
| Название | Загрузка документа без перезапуска приложения |
| Приоритет | Высокий |
| Тип | Позитивный |
| *Окружение* |  |
| Система | XML |
| Номер требования | 9, 10 |
| *Цель* |  |
| Начальные условия | Приложение запущено, имеются различные файлы с документами, сохранённые ранее в количестве не менее трёх |
| Действия | Последовательно открыть не менее трёх различных файлов с документами |
| Ожидаемый результат | После открытия документы восстанавливают вид, который они имели перед сохранением. Если открыть документ, в то время как один из документов уже был открыт ранее, то содержимое открываемого документа полностью замещает содержимое старого документа; содержимое старого документа больше не видно. |
| *Конечное состояние* |  |

|  |  |
| --- | --- |
| Тест | |
| Номер | 13 |
| Название | Перетаскивание вершин |
| Приоритет | Высокий |
| Тип | Позитивный |
| *Окружение* |  |
| Система | XML |
| Номер требования | 9, 10 |
| *Цель* |  |
| Начальные условия | В документе имеется несколько вершин |
| Действия | Перетащить вершину из одной точки в другую |
| Ожидаемый результат | Вершина меняет позицию вместе с указателем мыши, пока левая кнопка мыши не будет отпущена. Таким образом, можно задать любое расположение вершины в видимой области документа. После того, как левая кнопка мыши отпущена, вершина остаётся на месте. |
| *Конечное состояние* |  |

|  |  |
| --- | --- |
| Тест | |
| Номер | 14 |
| Название | Дополнительный тест на перетаскивание вершин |
| Приоритет | Низкий |
| Тип | Позитивный |
| *Окружение* |  |
| Система | XML |
| Номер требования | 9, 10 |
| *Цель* |  |
| Начальные условия | В документе имеется несколько вершин |
| Действия | Перетащить вершину из одной точки в другую, при этом провести перетаскиваемой вершиной по другим вершинам |
| Ожидаемый результат | Вершина меняет позицию вместе с указателем мыши, пока левая кнопка мыши не будет отпущена. Когда указатель мыши оказывается над маркерами других вершин, другие вершины не захватываются, а остаются на месте |
| *Конечное состояние* |  |

|  |  |
| --- | --- |
| Результаты тестирования финальной версии приложения | |
| Номер теста | Результат |
| 1 | + |
| 2 | + |
| 3 | + |
| 4 | + |
| 5 | + |
| 6 | + |
| 7 | + |
| 8 | + |
| 9 | + |
| 10 | + |
| 11 | + |
| 12 | + |
| 13 | + |
| 14 | + |

# Заключение

В рамках курсовой работы было создано приложение, отвечающее изначально поставленным требованиям. Тем не менее, некоторые изначально задуманные функции не были реализованы.