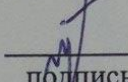


Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
Кафедра систем искусственного интеллекта

УТВЕРЖДАЮ

Заведующий кафедрой

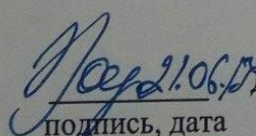

Г. М. Цибульский
подпись

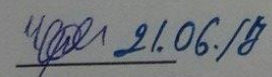
« 21. » 06 2017 г.

БАКАЛАВРСКАЯ РАБОТА

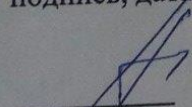
09.03.02 «Информационные системы и технологии»

Редактор базы знаний системы поддержки принятия решений в
сельскохозяйственном производстве

Руководитель  21.06.17 доц. кафедры СИИ, канд. техн. наук К. В. Раевич
подпись, дата

Выпускник  21.06.17 С. И. Черкасов
подпись, дата

Консультант  21.06.17 Ю. А. Маглинец
подпись, дата

Нормоконтролер  21.06.17 М. А. Аникьева
подпись, дата

Красноярск 2017

Продолжение титульного листа бакалаврской работы по теме «Редактор базы знаний системы поддержки принятия решений в сельскохозяйственном производстве».

Защитный лист

Нормоконтролер

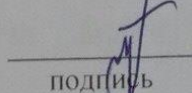
М.06.17
подпись, дата

М. А. Аникьева

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
Кафедра систем искусственного интеллекта

УТВЕРЖДАЮ

Заведующий кафедрой


Г. М. Цибульский

подпись
« 01 » 03 2017 г.

ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ
в форме бакалаврской работы

Студенту Черкасову Станиславу Игоревичу

Группа КИ13-14Б, направление подготовки 09.03.02 «Информационные системы и технологии», профиль 09.03.02.04 «Информационные системы в медиаиндустрии»

Тема выпускной квалификационной работы: Редактор базы знаний системы поддержки принятия решений в сельскохозяйственном производстве.

Утверждена приказом по университету № 2567/с от 01.03.2017.

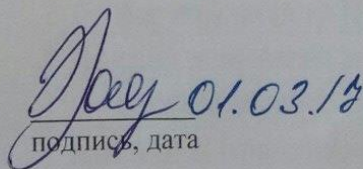
Руководитель ВКР: Раевич К. В. старший преподаватель кафедры СИИ.

Исходные данные к ВКР: материалы научно-учебной лаборатории кафедры систем искусственного интеллекта института космических и информационных технологий.

Перечень разделов ВКР:

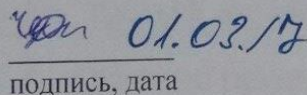
- введение;
- обзор предметной области;
- проектирование программы;
- реализация приложения;
- заключение;
- список использованных источников;
- приложения.

Руководитель ВКР


подпись, дата

К. В. Раевич

Задание принял к исполнению


подпись, дата

С. И. Черкасов

« 01 » 03 2017 г.

График

выполнения выпускной квалификационной работы студентом направления 09.03.02 «Информационные системы и технологии», профиля 09.03.02.04 «Информационные системы в медиаиндустрии».

График выполнения выпускной квалификационной работы приведен в таблице 1.

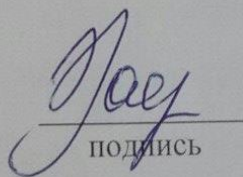
Таблица 1 – График выполнения этапов ВКР

Наименование этапа	Срок выполнения этапа	Результат выполнения этапа	Примечание руководителя (отметка о выполнении этапа)
Ознакомление с целью и задачами работы	2.03 – 8.03	Краткое эссе по теме ВКР	Выполнено
Сбор литературных источников	8.03 – 19.03	Список источников информации	Выполнено
Анализ собранных литературных источников	20.03 – 26.03	Реферат о проблемно-предметной области	Выполнено
Уточнение и обоснование актуальности цели и задач ВКР	27.03 – 02.04	Окончательная формулировка цели и задач ВКР	Выполнено
Подготовка доклада и презентации по теме ВКР	03.04 – 09.04	Доклад и презентация по ВКР	Выполнено
Решение первой задачи	10.04 – 14.04	Доклад и презентация по первой задаче ВКР	Выполнено
Решение второй задачи	15.04 – 19.04	Доклад и презентация по второй задаче ВКР	Выполнено
Решение третьей задачи	20.04 – 25.04	Доклад и презентация по третьей задаче ВКР	Выполнено
Компоновка отчета по результатам решения задач ВКР	25.05 – 06.06	Отчет по результатам решения ВКР	Выполнено
Предварительная защита результатов ВКР	07.06	Доклад и презентация о проделанной работе	Выполнено

Окончание таблицы 1

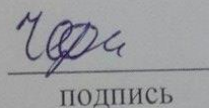
Наименование этапа	Срок выполнения этапа	Результат выполнения этапа	Примечание руководителя (отметка о выполнении этапа)
Нормоконтроль	08.06 – 21.06	Пояснительная записка	Выполнено
Защита ВКР	23.06	Пояснительная записка и презентация по результатам бакалаврской работы	

Руководитель ВКР


подпись

К. В. Раевич

Студент гр. КИ13-14Б


подпись

С. И. Черкасов

СОДЕРЖАНИЕ

Введение.....	5
1 Обзор предметной области	6
1.1 Системы поддержки принятия решений	6
1.2 Базы знаний	10
1.3 Существующие редакторы баз знаний	12
1.4 Вывод по главе 1	15
2 Проектирование программы	16
2.1 Требования к редактору базы знаний	16
2.2 Варианты использования	17
2.2.1 Диаграмма вариантов использования	17
2.2.2 Реестр вариантов использования.....	18
2.2.3 Описание вариантов использования	19
2.3 Диаграммы деятельности	21
2.4 Диаграмма классов.....	24
2.5 Диаграмма взаимодействия	26
2.6 Диаграмма кооперации.....	27
2.7 Диаграмма компонентов	28
2.8 Диаграмма развертывания	29
2.9 Вывод по главе 2	29
3 Реализация приложения	31
3.1 Выбор средства для разработки	31
3.2 Хранилище базы знаний.....	31
3.3 Редактор базы знаний	32
3.4 Вывод по главе 3	36
Заключение	37
Список сокращений	38

Список использованных источников	39
Приложение А Диаграмма классов	41
Приложение Б Техническое задание.....	42
Приложение В Плакаты презентации	48

ВВЕДЕНИЕ

Земельные ресурсы требуют рационального использования, так как почва имеет свойство терять свое плодородие. В основном это обуславливается нерациональной деятельностью человека. Для контроля состояния и правильного использования свойств земли, необходимо накапливать данные о качестве и отслеживать динамику их развития.

В ходе выполнения пилотного проекта, сотрудниками научно-учебной лаборатории института космических и информационных технологий (НУЛ ИКИТ), «Система агромониторинга масштаба муниципального района» [1], накоплен значительный объем информации о состоянии земель сельскохозяйственного назначения (ЗСХН). Эта информация представлена, как в виде данных, привязанных к конкретным исследуемым территориям, так и в форме знаний, обобщающих описание земель сельскохозяйственного назначения и ее свойств, как сложного, многофакторного объекта. Для принятия объективного управленческого решения, в области землепользования, необходим дополнительный инструментарий — система поддержки принятия решений (СППР). Такая система поможет формировать оценки земель ЗСХН, а так же осуществлять информационную поддержку принятия решений в области управления земельными ресурсами.

В состав СППР входит база знаний (БЗ), для работы с которой требуется, в связи с узкой направленностью предметной области, разработать редактор, для сокращения времени, затрачиваемого на работу с БЗ.

Темой выпускной квалификационной работы в виде бакалаврской работы является «Редактор базы знаний системы поддержки принятия решений в сельскохозяйственном производстве».

В рамках работы решаются следующие задачи:

- обзор предметной области;
- проектирование приложения;
- разработка редактора базы знаний.

1 Обзор предметной области

1.1 Системы поддержки принятия решений

Системы поддержки принятия решений (СППР) — это узко профилированные приложения, предназначенные для управленцев определенного профиля и характеризуется наличием возможности реализовывать относительно высокий уровень формализации выработки рекомендаций по принятию управленческих решений [2].

Существующие и разрабатываемые СППР, ориентированные на знания, обеспечивают специализированные решения проблем, основанные на фактах и правилах применения этих фактов. Такие системы находят широкое применение во многих областях, например в телекоммуникациях, медицине, сельском хозяйстве, управлении водохозяйствами, оптимизации технологических процессов производств и т. д.

Перед СППР ставится задача предоставления, лицу принимающему решения, инструмента для выполнения анализа данных. Система не генерирует правильных решений, а только данные в обработанном виде для изучения и анализа, именно поэтому такие системы обеспечивают выполнение функции поддержки принятия решений.

Для того чтобы определить необходимость создания редактора базы знаний, нужно рассмотреть структурные решения систем поддержки принятия решений.

По мнению группы ученых из Самарского государственного технического университета интеллектуальная система поддержки принятия решений состоит из двух основных подсистем (рисунок 1):

- «Ядро» ИСППР — содержит «Навигатор» системы и модули системного анализа, оптимизации, расчетов и формирования необходимой отчетности. «Ядро» ИСППР реализует общие алгоритмы работы. Модули системного анализа, оптимизации и статистической обработки используются встраиваемой

лицензионное программное обеспечение — математический пакет «Mathworks MatLab»;

- комплекс специализированных модулей — содержит систему управления базами данных, базу знаний и экспертную систему [3].



Рисунок 1 – Структура системы поддержки принятия решений при управлении областными целевыми программами

Структура СППР в земледелии представлена на рисунке 2 [4].

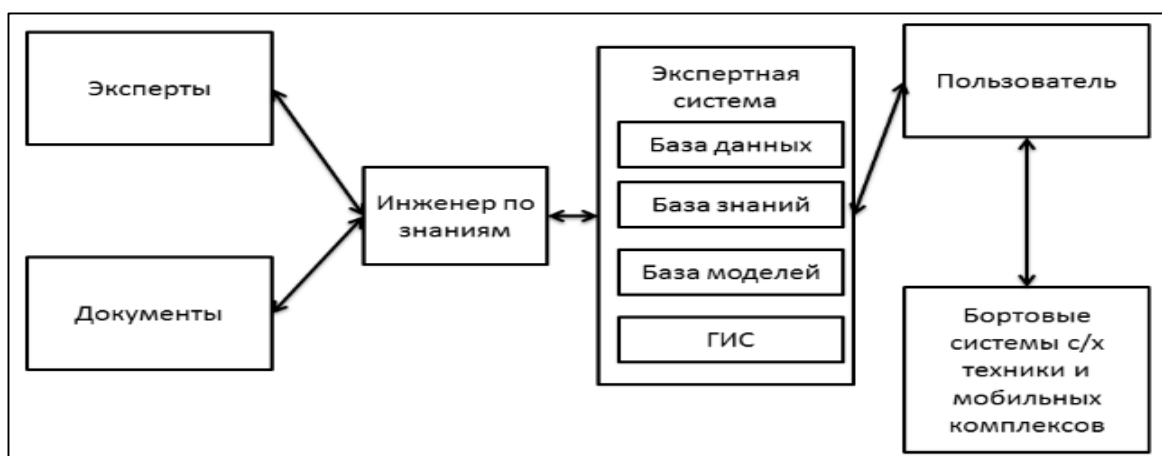


Рисунок 2 – Структура системы поддержки принятия решений в земледелии

На рисунке 3 представлена функциональная структура СППР для земель сельскохозяйственного назначения по версии сотрудников НУЛ ИКИТ [5].

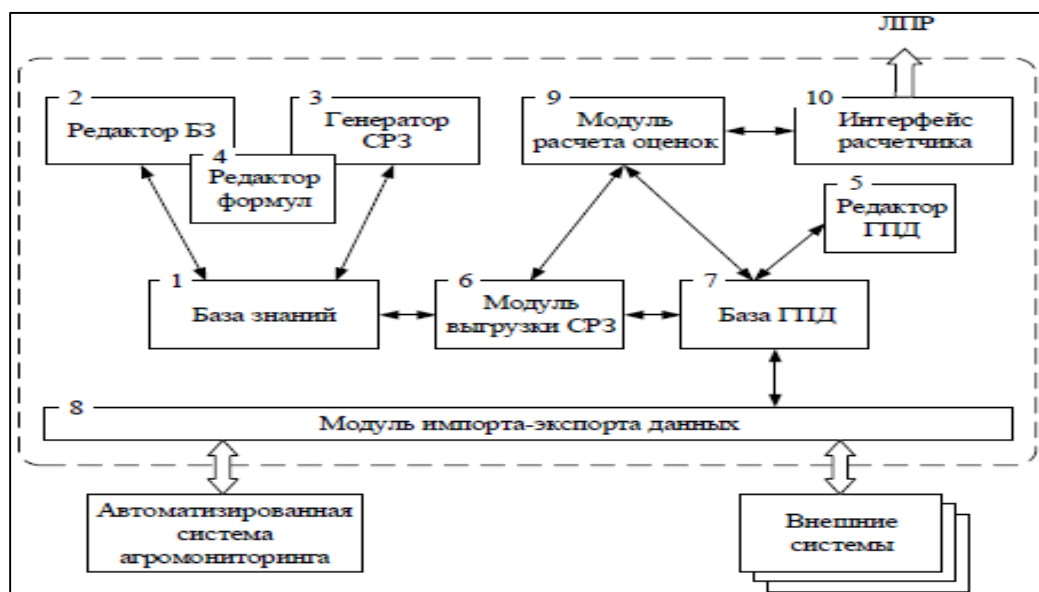


Рисунок 3 – Структура системы поддержки принятия решений для земель сельскохозяйственного назначения

СППР состоит из следующих подсистем:

1) оценка ЗСХН — это сложный многофакторный анализ, потому что необходимо учитывать климатические факторы, характеристики почвенного и растительного покровов, особенности инфраструктуры, геопространственные характеристики оцениваемых участков земной поверхности и др. Взвешенные достоверные оценки базируются на знании экспертов в различных областях: геоботанике, экономике, экологии, сельском хозяйстве. Построение системы оценок специфично для конкретной решаемой задачи и для конкретной территории, что затрудняет непосредственное привлечение экспертов. Для отчуждения экспертных знаний от их носителей и представления их в формализованном виде, доступном для использования, применяются методы «Инженерии знаний». Поэтому база знаний (БЗ) является ключевым блоком системы;

2) редактор БЗ позволяет инженеру, ответственному за содержание онтологии, осуществлять диалоговые операции по созданию новой онтологии

(версии онтологии) и ее редактированию в части описания таксономии, метрик измерения натуральных параметров объектов оценки и вычислительных соотношений;

3) генератор схемы решения задачи (СРЗ) позволяет эксперту, ответственному за представление задачи, создавать и поддерживать модель описания задачи в рамках определенной версии онтологии, а также осуществлять операции по миграции описания задачи в обновленные версии онтологии;

4) при создании и модификации онтологии возникает процедура описания вычислительных соотношений для метрик оценивания. Данная процедура базируется на использовании редактора формул, который входит как компонента в редактор базы знаний и генератор схемы решения задачи. Помимо базовых математических операций генератор формул содержит возможность формирования таблично заданных функций;

5) редактор геопространственных данных (ГПД) служит для подготовки и оперирования геопространственным описанием экземпляра решаемой задачи;

6) при создании пустого экземпляра решаемой задачи (модуль 5) модуль выгрузки СРЗ порождает шаблон векторного слоя с предопределенным набором атрибутов, заданных в схеме решения задачи, который помещается в базу ГПД;

7) база ГПД представляет собой набор слоев, наполненных данными о пространственных координатах объектов оценки, тем самым формируется векторное представление данных на анализируемую территорию, содержащее множество объектов оценки как геопространственных объектов;

8) атрибутивная информация содержит значения натуральных параметров объектов оценки. Поставка атрибутивной информации – ответственность модуля импорта-экспорта данных. Данный модуль взаимодействует с автоматизированной системой агромониторинга и другими внешними системами, что позволяет получать данные на основе обработки и анализа космоснимков, наземных измерений, данных метеостанций и других источников;

9) расчет промежуточных и финишных оценок осуществляется модулем расчета оценок;

10) визуализация оценок осуществляется через интерфейс расчетчика, позволяющий формировать пространственное представление данных в виде векторных слоев с вычисленными атрибутами, а также отображать данные в табличной форме либо средствами деловой графики.

Анализируя структурные решения для систем поддержки принятия решений, рассмотренные выше, можно сказать, что количество модулей и их состав на прямую зависит от области применения и назначения системы. Так же можно выделить основные модули для СППР:

- интерфейс пользователя;
- модуль обработки данных;
- база знаний;
- редактор базы знаний.

1.2 Базы знаний

Подходы к созданию СППР, вне зависимости от предметной области, предполагают использование баз знаний (БЗ), содержащих совокупность фактов и правил логического вывода, моделирующих поведение эксперта с использованием процедур логического вывода и принятия решений (продукционные системы).

Эффективность продукционных, нечетко-логических систем и их модификаций во многом определяется объемом и качеством содержащихся в них формализованных экспертных знаний. Для этого инженеры по знаниям при построении СППР, должны представить в некой форме рассуждения и заключения экспертов о предметной области, которые ложатся в основу базы знаний.

Базы знаний СППР создаются с участием следующих специалистов:

- эксперты предметной области, к которой относятся решаемые СППР задачи;

- инженеры по знаниям, разрабатывающие СППР;

- программисты, осуществляющие непосредственную реализацию СППР в виде программного кода.

Как правило, базы знаний СППР содержат информацию двух видов:

- факты, представляющие собой статические сведения о предметной области;

- правила, представляющие собой наборы инструкций, по которым на основе известных фактов можно получать новые факты.

Факты в БЗ описывают те явления, которые являются постоянными для исследуемой предметной области. Характеристики, значения которых зависят от условий текущей задачи, СППР получает от пользователя в процессе и сохраняет их в своей рабочей памяти. Например, факт — «температура тела здорового человека равна 36,6 градусам по шкале Цельсия» хранится в базе знаний, а факт «температура тела пациента равна 37,9 градусам по шкале Цельсия» — в рабочей памяти СППР [6].

Функционирование баз знаний строится в соответствии с какой-либо моделью. Наиболее распространенными моделями в настоящее время являются:

- деревья (вывода, целей);

- семантические сети (ассоциативные сети);

- фреймы;

- нечёткие множества.

В системе поддержки принятия решений разрабатываемой научно-учебной лабораторией ИКИТ [7] знания представляются в виде семантической сети.

Семантическая сеть — это ориентированный граф, в узлах которого находятся имена объектов, а стрелки указывают на отношения между ними (рисунок 4). Из неизмеримого множества отношений часто используются «часть-целое», свойства и функциональные связи («наследник», «находится» и т. д.).



Рисунок 4 – Семантическая сеть

Семантическая сеть обрабатывается на основе принципа сопоставления объекта и отношения, которые указаны в запросе, с объектами и отношениями, имеющимися в семантической сети. Например, если запрос имеет вид: «Что производит ОАО «Горизонт»», будет выделен тот фрагмент сети, где фигурируют указанный объект («Горизонт») и отношение «производит». Ответом будет: «ОАО «Горизонт» производит кинескопы».

1.3 Существующие редакторы баз знаний

В ходе выполнения поиска аналогичных редакторов баз знаний, работающих с семантическими сетями, было выявлено, что большинство редакторов существуют только в интегрированном в экспертные системы, и им подобные, виде.

«**Semantic network v4.0**» — это программа, предназначенная для работы с семантическими сетями, а именно — позволяющая их строить, изменять и выполнять простейшие поисковые запросы [8].

Главное окно, после запуска программы, представляет из себя «холст» для графического отображения и редактирования знаний и панель инструментов (рисунок 5).

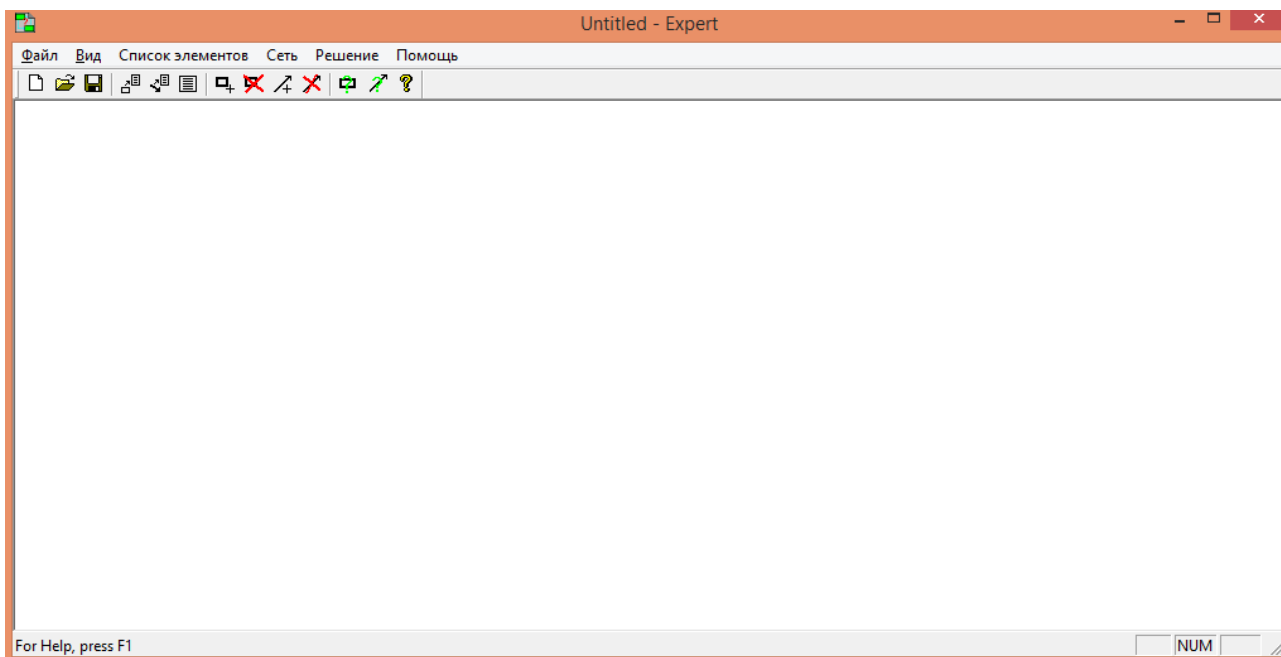


Рисунок 5 – Главное окно программы «Semantic network v4.0»

В данной программе можно осуществлять:

- создание новой семантической сети;
- открытие сохраненной ранее семантической сети в формате «Tsf»;
- сохранение созданной сети;
- добавление узлов;
- добавление дуг.

Чтобы в данной программе создать узел или дугу (связь), сначала нужно добавить их в «Доступные». Для этого необходимо, каждый раз вызывать окно для ввода, нажимая на кнопку «Добавить связь в список» или «Добавить узел в список», либо открыть «Список узлов и связей» (рисунок 6) и непосредственно из этого окна заполнять список.

Для того чтобы связать узлы их необходимо разместить на «Холсте», затем из меню быстрого доступа выбрать «Добавление связи в сеть», из появившегося окна выбрать нужную из списка «Доступных» и нажать левой кнопкой мыши на узел от которого будет идти связь, и аналогичным действием указать к какому узлу.

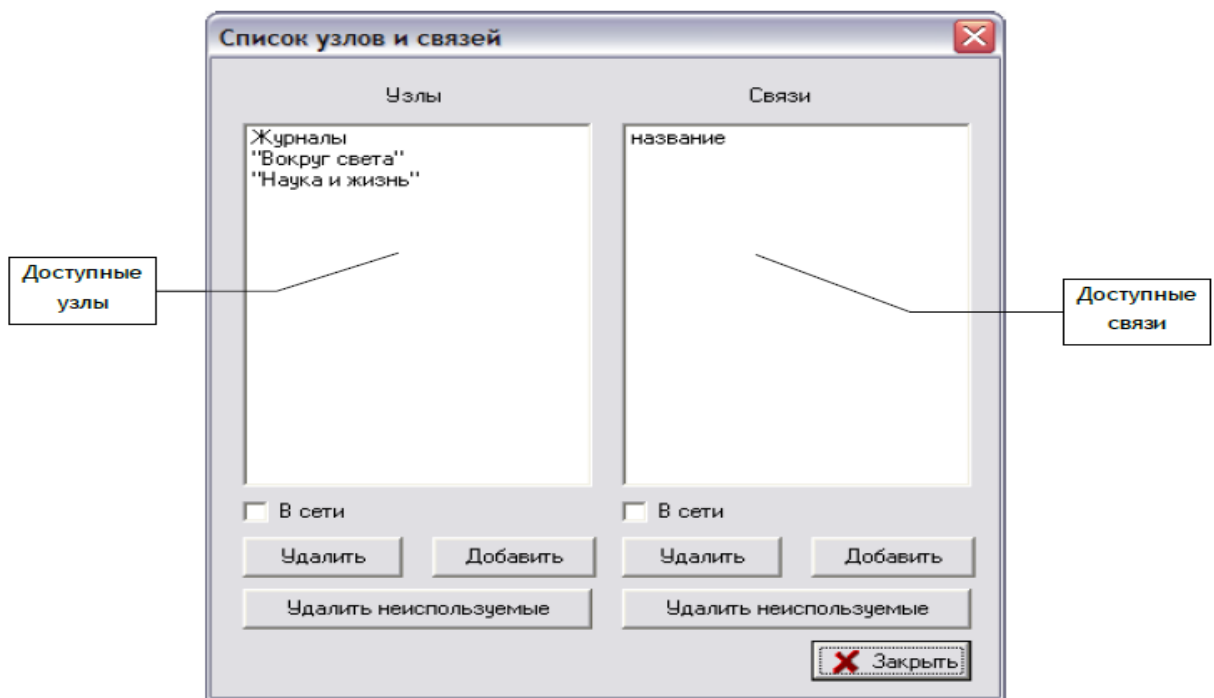


Рисунок 6 – Окно «Список узлов и связей»

Пример работы программы приведен на рисунке 7.

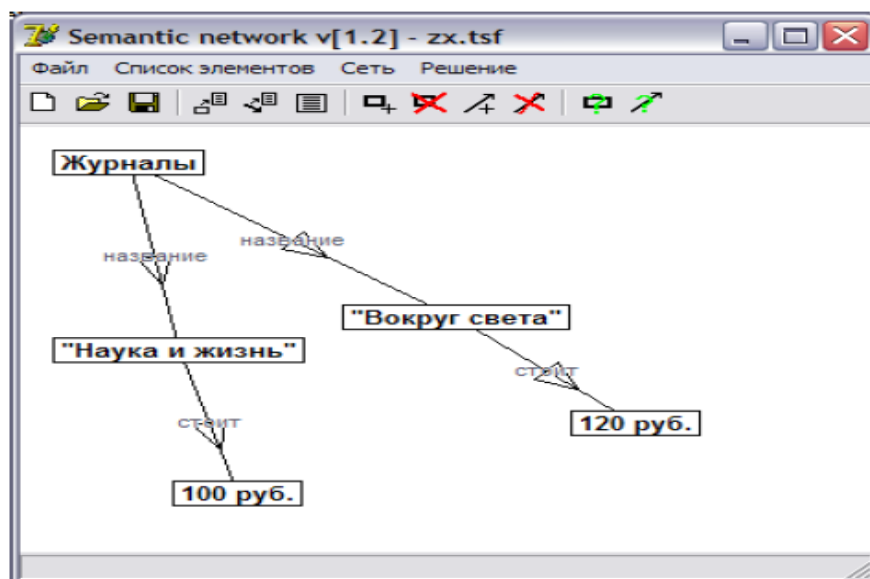


Рисунок 7 – Пример работы программы

Данная программа представляет собой инструмент для создания семантических сетей и выполнения простейших поисковых операций. Использование «Semantic network v4.0» в качестве редактора базы знаний

для СППР, разрабатываемой НУЛ ИКИТ, не подходит из-за того, что для создаваемых объектов нельзя задавать параметры, так же интерфейс в виде диалоговых операций замедляет работу по наполнению базы знаний и не предусматривается возможность интеграции с какими-либо системами.

1.4 Вывод по главе 1

В ходе выполнения задачи обзора предметной области, были рассмотрены: системы поддержки принятия решений, базы знаний и существующие редакторы баз знаний.

Анализируя структуры различных СППР видно, что для работы с базами знаний в таких системах выделяют модуль «Редактор базы знаний». В связи с узкой направленностью предметных областей использующих системы поддержки принятия решений, а так же формата представления и хранения знаний в БЗ, разработка такого модуля является необходимой.

Редактор базы знаний обеспечит:

- создание баз знаний;
- редактирование баз знаний;
- графическое представление данных в виде древовидной структуры.

2 Проектирование программы

Под проектированием понимается процесс определения компонентов, архитектуры, интерфейсов и других характеристик системы. Обнаружение и исправление ошибок на этапе проектирования позволяет избегать временных и финансовых затрат, увеличивающихся на порядок при разработке. Результатом этапа является технический проект, который разрабатывают с целью выявления окончательных технических решений, дающих полное представление о конструкции изделия, когда это целесообразно сделать до разработки рабочей документации [9].

Встречаются разнообразные подходы к проектированию. На сегодняшний день, стандартом при разработке систем является унифицированный язык моделирования («UML»). Язык «UML» служит для определения, отображения и описания элементов объектно-ориентированных систем в процессе их создания. Он содержит стандартный набор диаграмм и нотаций, применяемых для визуализации разрабатываемой системы с разных точек зрения.

2.1 Требования к редактору базы знаний

Редактор БЗ позволяет «Эксперту», ответственному за содержание, осуществлять операции по созданию и редактированию базы знаний, записывать ее в файл и открывать сохраненные файлы. Он должен визуализировать граф на экране монитора и отображать параметры вершин графа, предоставлять пользователю возможность редактировать знания связанные с вершинами.

Исходя из этих задач, можно сформулировать следующие требования к редактору. Редактор должен обеспечивать:

- возможность создания и удаления вершин;
- возможность изменять следующие параметры вершин графа: имя, параметры. Имена вершин графа должны быть уникальны;
- вывод на экран монитора графа и параметров его вершин;

- редактирование каждого узла.
- сохранение базы знаний в файле, записанной на исходном языке. Чтение таких файлов.

2.2 Варианты использования

Варианты использования описывают поведение системы, когда она взаимодействует с кем-то (или чем-то) из внешней среды. Система может отвечать на внешние запросы «Актёра» может сама выступать инициатором взаимодействия. Другими словами, сценарий использования описывает, «кто» и «что» может сделать с рассматриваемой системой, или что система может сделать с «кем» или «чем». Методика сценариев использования применяется для выявления требований к поведению системы, известных также как пользовательские и функциональные требования. [9]

2.2.1 Диаграмма вариантов использования

Диаграмма вариантов использования описывает функциональное назначение системы, и являясь её концептуальной моделью, позволяет отобразить, описать и «задокументировать» желаемое поведение системы с точки зрения взаимодействия с ней внешних объектов («Актеров»).

В ходе выявления вариантов использования, выделяется один вариант использования «Работа с базой знаний». На диаграмме (рисунок 8) показаны действия, доступные «Эксперту», который взаимодействует с системой.

«Работа с базой знаний» включает в себя функции: «Создать базу знаний», «Открыть базу знаний» и «Сохранить базу знаний». Функции «Создать базу знаний» и «Открыть базу знаний» включают в себя функцию «Визуализация знаний» и расширяются функциями: «Добавление знаний», «Изменение знаний» и «Удаление знаний».

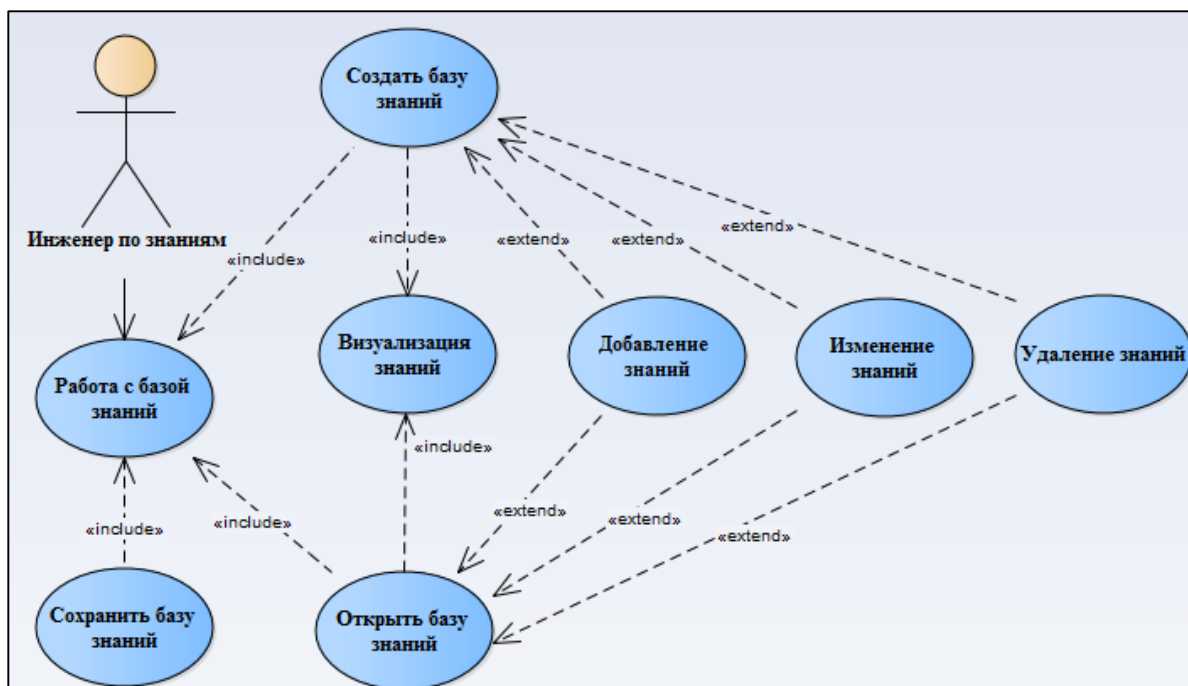


Рисунок 8 – Вариант использования «Работа с базой знаний»

2.2.2 Реестр вариантов использования

Для разрабатываемого приложения выделяется один пользователь — «Эксперт», который взаимодействует со всеми вариантами использования.

В таблице 1 приведен список вариантов использования.

Таблица 1 – Список вариантов использования

Код	Основной актер	Наименование	Формулировка
И1	Эксперт	Создать базу знаний	Данный вариант использования позволяет актеру создать чистую базу знаний
И2	Эксперт	Открыть базу знаний	Данный вариант использования позволяет открывать существующую базу знаний
И3	Эксперт	Визуализация знаний	Вариант использования, позволяющий визуализировать знания в виде ориентированного графа
И4	Эксперт	Сохранение базы знаний	Данный вариант использования позволяет сохранять созданную или измененную базу знаний

Окончание таблицы 1

Код	Основной актер	Наименование	Формулировка
И5	Инженер	Добавление знаний	Данный вариант использования позволяет добавлять новые знания в уже созданную или открытую базу знаний
И6	Инженер	Изменение знаний	Вариант использования, позволяющий изменять знания в базе
И7	Инженер	Удаление знаний	Вариант использования, позволяющий удалять знания из базы

2.2.3 Описание вариантов использования

И1. Создать базу знаний (таблица 2).

Таблица 2 – Вариант использования И1

Код	Основной актер	Наименование	Формулировка
И1	Инженер по знаниям	Создать базу знаний	Данный вариант использования позволяет актеру создать чистую базу знаний

Основное действующее лицо: Инженер по знаниям.

Связи с другими вариантами использования: отсутствуют.

Данный вариант использования позволяет «Инженеру» создавать чистую форму для дальнейшего использования в целях создания новой базы знаний.

И2. Открыть базу знаний (таблица 3).

Таблица 3 – Вариант использования И2

Код	Основной актер	Наименование	Формулировка
И2	Инженер по знаниям	Открыть базу знаний	Данный вариант использования позволяет открывать существующую базу знаний

Основное действующее лицо: Инженер по знаниям.

Связи с другими вариантами использования: отсутствуют.

Данный вариант использования позволяет «Инженеру» открывать наполненные базы знаний, хранимые в формате «XML».

ИЗ. Создать базу знаний (таблица 4).

Таблица 4 – Визуализация знаний ИЗ

Код	Основной актер	Наименование	Формулировка
ИЗ	Инженер по знаниям	Визуализация знаний	Вариант использования, позволяющий визуализировать знания

Основное действующее лицо: Инженер по знаниям.

Связи с другими вариантами использования: отсутствуют.

Данный вариант использования позволяет «Инженеру» визуализировать знания в виде ориентированного графа.

И4. Создать базу знаний (таблица 5).

Таблица 5 – Сохранение базы знаний И4

Код	Основной актер	Наименование	Формулировка
И4	Инженер по знаниям	Сохранение базы знаний	Данный вариант использования позволяет сохранять созданную или измененную базу знаний

Основное действующее лицо: Инженер по знаниям.

Связи с другими вариантами использования: отсутствуют.

Данный вариант использования позволяет «Инженеру» сохранять результат работы с базой знаний, а так же её визуализацию.

И5. Добавление знаний (таблица 6).

Таблица 6 – Вариант использования И5

Код	Основной актер	Наименование	Формулировка
И5	Инженер по знаниям	Добавление знаний	Вариант использования добавление новых знаний

Основное действующее лицо: Инженер по знаниям.

Связи с другими вариантами использования: отсутствуют.

Данный вариант использования позволяет «Инженеру» добавлять знания и вершины графа и дуги.

И6. Создать базу знаний (таблица 7).

Таблица 7 – Изменение знаний И6

Код	Основной актер	Наименование	Формулировка
И6	Инженер по знаниям	Изменение знаний	Вариант использования, позволяющий изменять знания в базе

Основное действующее лицо: Инженер по знаниям.

Связи с другими вариантами использования: отсутствуют.

Данный вариант использования позволяет «Инженеру» редактировать открытую базу знаний, путем изменения значений параметров знаний или полной их заменой на другие.

И7. Удаление знаний (таблица 8).

Таблица 8 – Вариант использования И7

Код	Основной актер	Наименование	Формулировка
И7	Инженер по знаниям	Удаление знаний	Вариант использования, позволяющий удалять знания из базы

Основное действующее лицо: Инженер по знаниям.

Связи с другими вариантами использования: отсутствуют.

Данный вариант использования позволяет «Инженеру» удалять знания, тем самым удалять узлы и связи визуального графа.

2.3 Диаграммы деятельности

Для дальнейшего рассмотрения были выбраны два прецедента:

- И1. Создать базу знаний;
- И2. Открыть базу знаний.

Прецедент И1. «Создать базу знаний».

Краткое описание: Инженер создает новую базу знаний.

Поток событий

Прецедент начинается, когда «Инженер» выбирает деятельность «Создать базу знаний» из выпадающего меню программы «Проект».

Базовый поток — «Создать базу знаний».

1. «Инженер» выбирает «Создать базу знаний»;
2. «Система» проверяет, велась ли до этого работа с другой базой знаний (является ли она, на данный момент, активной);
3. «Система» очищает форму интерфейса.

Точки расширения

Если при выполнении 2 пункта базового потока велась работа с другой базой знаний, то «Система» выдаст сообщение, в котором будет предложено сохранить активную базу знаний. Если потребуется сохранение, то «Система» сохранит базу и очистит форму интерфейса. Если сохранение не потребуется, то «Система» очистит форму интерфейса. Если будет выбран вариант «Отменить», означающий, что «Инженер» продолжит работать с активной базой знаний, то новая база создана не будет и прецедент закончится.

На рисунке 9 представлена диаграмма деятельности прецедента «Создать базу знаний».

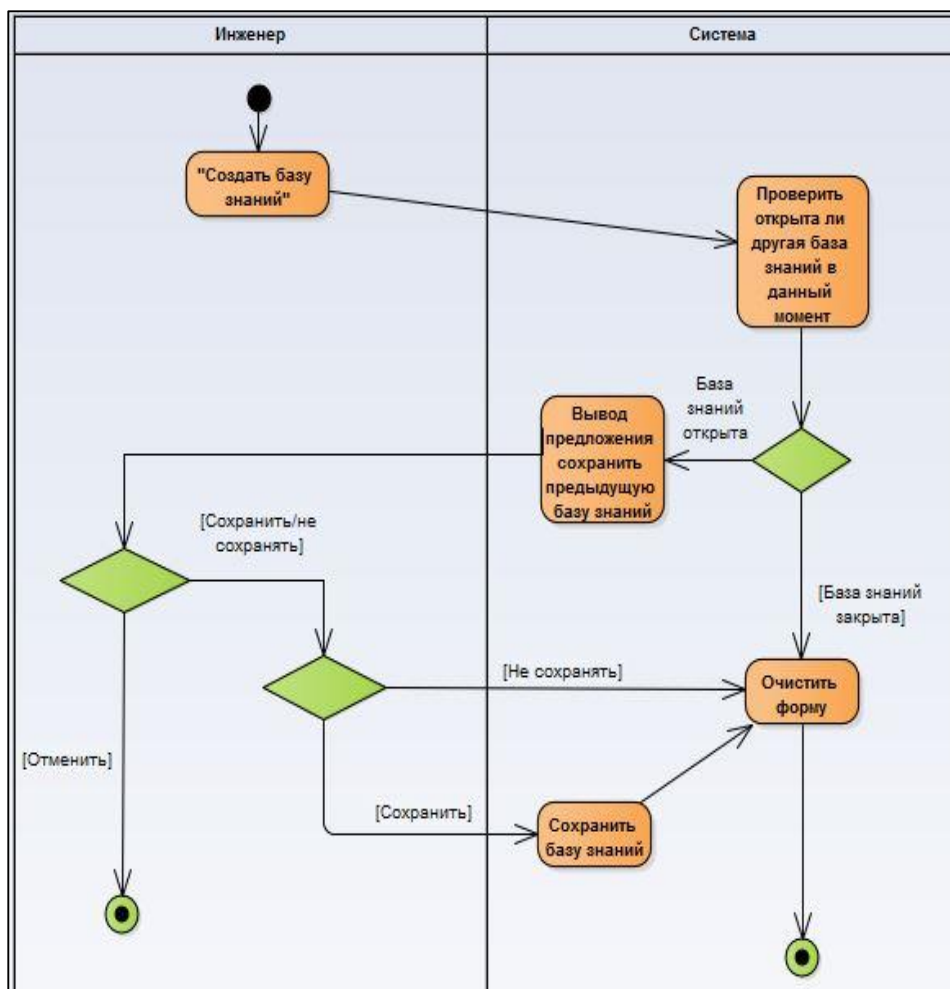


Рисунок 9 – Диаграмма деятельности прецедента «Создать базу знаний»

Прецедент И2. «Открыть базу знаний».

Краткое описание: позволяет «Инженеру» открывать существующие базы знаний.

Поток событий

Прецедент начинается, когда Инженер выбирает деятельность «Открыть базу знаний» в выпадающем меню программы «Проект».

Базовый поток — «Открыть базу знаний»

1. «Инженер» выбирает «Открыть базу знаний»;
2. «Система» открывает окно для выбора файла с базой;
3. «Инженер» выбирает файл;
4. «Система» проверяет формат файла;
5. «Система» отображает базу знаний.

Точки расширения

Если при выполнении пункта 4 базового потока был выбран не поддерживаемый формат, то «Система» выдаст сообщение об ошибке и «Инженеру» будет выведено окно, для повторного выбора файла.

На рисунке 10 представлена диаграмма прецедента «Открыть базу знаний».

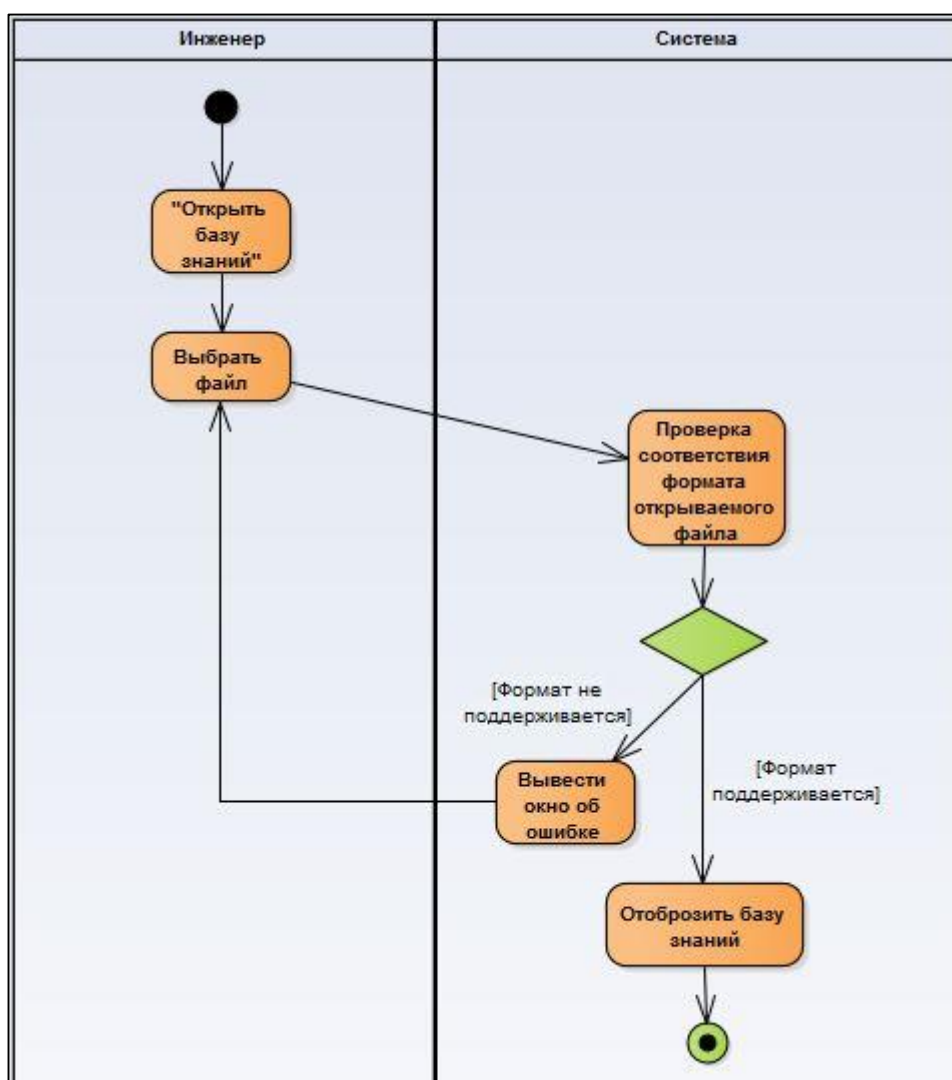


Рисунок 10 – Диаграмма деятельности прецедента И2

2.4 Диаграмма классов

Диаграмма классов — диаграмма, демонстрирующая классы системы, их атрибуты и методы, с точки зрения реализации, используемые непосредственно в программном коде [9].

В связи с выбором шаблона проектирования приложения «MVVM», общая диаграмма реализации которого представлена на рисунке 11, диаграмма классов будет иметь аналогичный вид, так как бизнес-логика приложения концентрируется в модели, взаимодействие представления и модели в модель-представлениях, а само представление использует привязки и команды за счет синтаксиса «XAML».

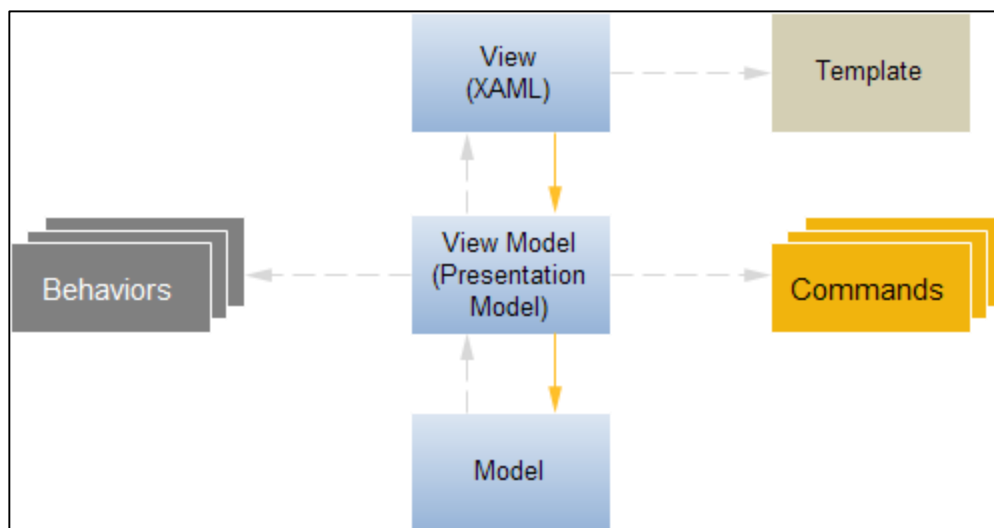


Рисунок 11 – Общая диаграмма реализации шаблона «Model-View-ViewModel»

Диаграмма классов, представляющая конечную структуру приложения, представлена в приложении А.

Библиотека классов «Model» представляет бизнес-логику приложения.

Класс «Node» содержит следующие свойства:

- «id» узла;
- название узла;
- ссылка на родительский узел;
- ссылки на узлы-потомки;
- позиция на карте по оси «У»;
- позиция на карте по оси «Х».

Интерфейс «IRepository» содержит формальное описание всех методов для реализации чтения и записи узлов в файлы, а также внутренние методы взаимодействия и редактирования узлов.

Класс «Repository» является реализацией интерфейса «IRepository».

Библиотека классов «NodeEditorViewModel» содержит методы и свойства (и их реализацию), с которыми связывается класс «View». Этот класс реализует интерфейс «INotifyPropertyChanged», с помощью которого можно создать свойство, которое будет уведомлять всех, кто на него подписан о своем изменении (в случае изменения).

Библиотека классов «NodeEditorView» — графическая визуализация. Разрабатываемая программа содержит в себе разметку «XAML», в которой все элементы связаны на методы и свойства из «NodeEditorViewModel».

2.5 Диаграмма взаимодействия

Диаграммы взаимодействия — являются моделями, описывающими поведение взаимодействующих групп объектов. Как правило, диаграмма взаимодействия охватывает поведение только одного варианта использования. На такой диаграмме отображается ряд объектов и те сообщения, которыми они обмениваются между собой в рамках данного варианта использования [9].

Диаграмма взаимодействия для прецедента «Создать базу знаний» представлена на рисунке 12.

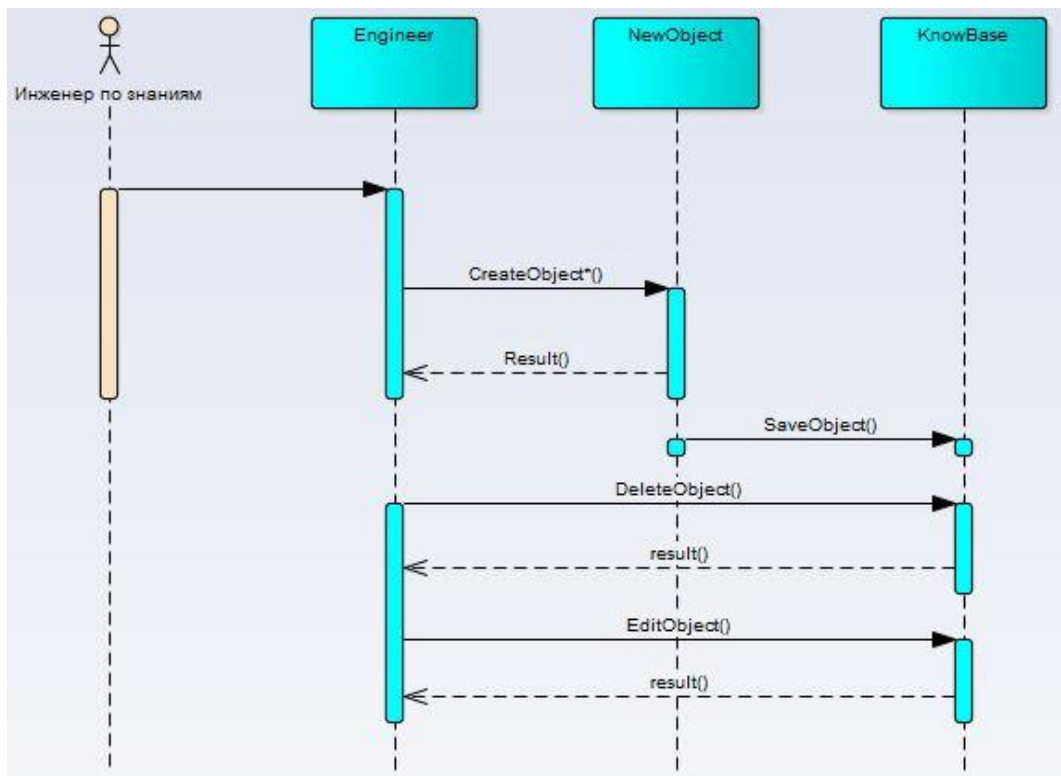


Рисунок 12 – Диаграмма взаимодействия прецедента
«Создать базу знаний»

2.6 Диаграмма кооперации

Диаграммы кооперации похожи на диаграммы последовательности. Такой тип диаграмм также отображает взаимодействие между классами, через набор объектов и сообщений между ними. Отличие состоит в том, что диаграмма последовательности акцентирует внимание на временном упорядочивании взаимодействия объектов, а диаграмма кооперации на общую организацию взаимодействующих объектов в соответствии с их пространственным расположением.

Диаграмма кооперации представлена на рисунке 13.

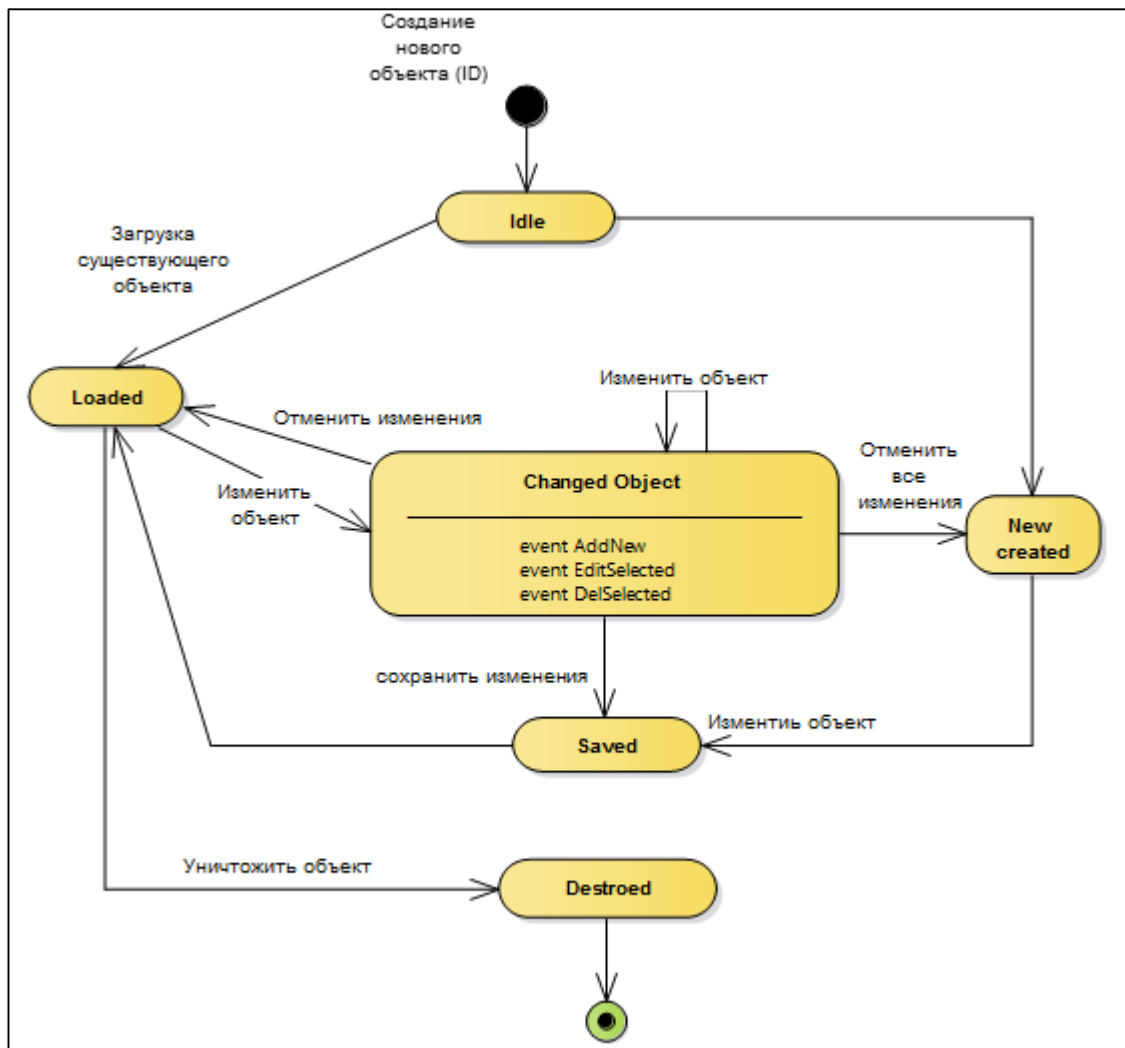


Рисунок 13 – Диаграмма кооперации

2.7 Диаграмма компонентов

Разбив функционал системы на компоненты, можно выделить следующие:

- «Repository» — компонент реализующий взаимодействие с БЗ системы;
 - «NodEditor» — компонент для работы с редактором;
 - «View» — компонент для представления интерфейса программы;
 - «IEditor» — интерфейс, реализующий графический интерфейс программы;
 - «IRepository» — интерфейс, реализующий представление данных из БЗ.
- На рисунке 14 представлена диаграмма компонентов.

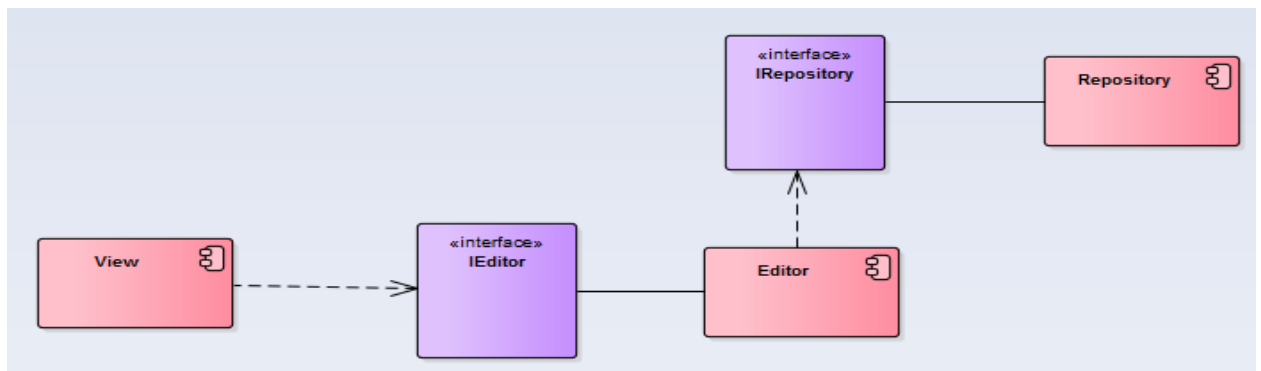


Рисунок 14 – Диаграмма компонентов.

2.8 Диаграмма развертывания

Диаграмма развертывания (рисунок 15) отображает способ взаимодействия компонентов с аппаратными средствами в физической системе, а также соединение аппаратных средств между собой.

Разрабатываемое приложение работает на локальном компьютере и не требует дополнительных узлов для развертывания.

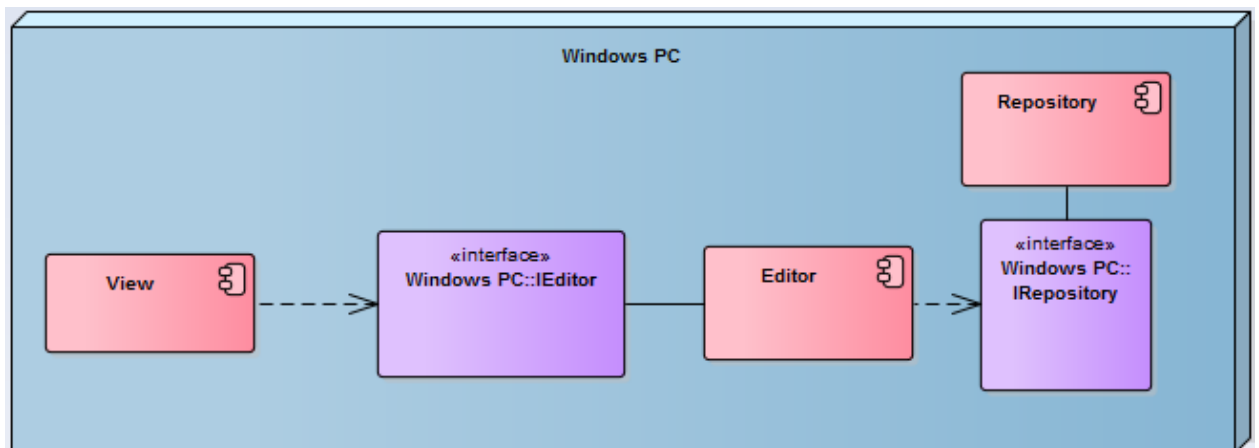


Рисунок 15 – Диаграмма развертывания

2.9 Вывод по главе 2

В ходе выполнения задачи — проектирование программы, был проведен анализ требований, предъявляемых в техническом задании (приложение Б),

на основании которых были сформулированы и отображены в UML-диаграммах варианты использования, представлены алгоритмы работы программы.

В ходе проектирования программы, для её разработки, был выбран шаблон проектирования «Model-View-ViewModel».

3 Реализация приложения

3.1 Выбор средства для разработки

Для реализации ПО была выбрана среда разработки «Visual Studio 2015», язык программирования «С#», так как разработанное приложение является модулем СППР [10], и для корректного взаимодействия с остальными модулями необходимо вести разработку при помощи соответствующего языка программирования.

3.2 Хранилище базы знаний

База знаний представляет собой семантическую сеть — метод представления знаний, отображающий совокупность объектов предметной области и отношений между ними. При этом, объектам соответствуют вершины сети, а отношениям — соединяющие их дуги. [11]

В семантическую сеть включаются только те объекты, которые необходимы для решения прикладных задач. В качестве объектов могут выступать события, действия, обобщенные понятия или свойства объекта. Наиболее распространёнными являются следующие типы отношений:

- «является» — означает, что объект входит в состав данного класса;
- «имеет» — позволяет задавать свойства объекта.

Для представления семантической сети будет использоваться расширяемый язык разметки — «XML», так как им можно описать древовидную структуру, что облегчит построение узлов и вершин графа для визуализации.

Структура XML-документа представлена на рисунке 16.

```

<?xml version="1.0"?>
<nodes>
  <node nodeType="Главная" id="1">
    <name>Участок №123</name>
    <previousNode>0</previousNode>
    <childs>2 3</childs>
  </node>
  <node nodeType="Обычная" id="2">
    <name>СоставПочвы</name>
    <previousNode>1</previousNode>
    <childs>3 4</childs>
    <relation>Имеет</relation>
  </node>
  <node nodeType="Обычная" id="3">
    <name>Гумус</name>
    <previousNode>2</previousNode>
    <parameters>
      <parameter>Гумус#132</parameter>
    </parameters>
    <relation>Является</relation>
  </node>
  <node nodeType="Обычная" id="4">
    <name>Джумус</name>
    <previousNode>2</previousNode>
    <parameters>
      <parameter>Джумус#17,2</parameter>
    </parameters>
    <relation>Является</relation>
  </node>
</nodes>

```

Рисунок 16 – Структура XML-документа

3.3 Редактор базы знаний

Редактор реализован в виде Windows-приложения, что обусловлено популярностью данной операционной системы и необходимой функциональностью. При открытии приложения открывается окно, в котором отображена главная форма приложения показана на рисунке 17.

Рисунок 17 – Главная форма редактора базы знаний

Форма содержит два выпадающих меню «Проект» (рисунок 18) и «Редактирование» (рисунок 19), и кнопки для работы с узлами и связями.

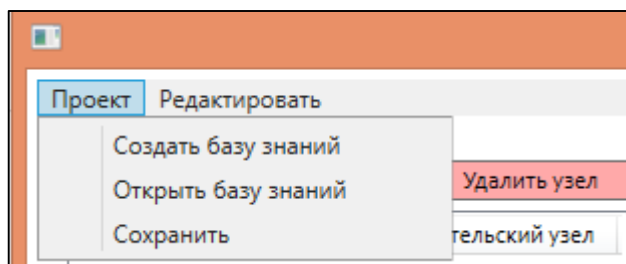


Рисунок 18 – Выпадающее меню «Проект»

Меню проект содержит кнопки: «Создать базу знаний», «Открыть базу знаний» и «Сохранить».

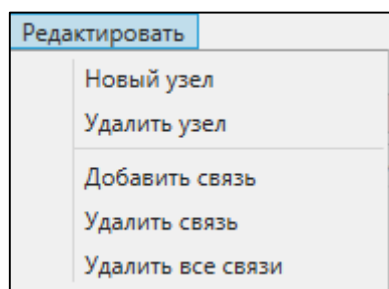


Рисунок 19 – Меню «Редактировать»

Для создания базы знаний достаточно нажать на соответствующую кнопку, при этом форма очистится и можно преступить к заполнению базы. Чтобы создать новый узел нужно нажать на кнопку «Новый узел» (рисунок 20).

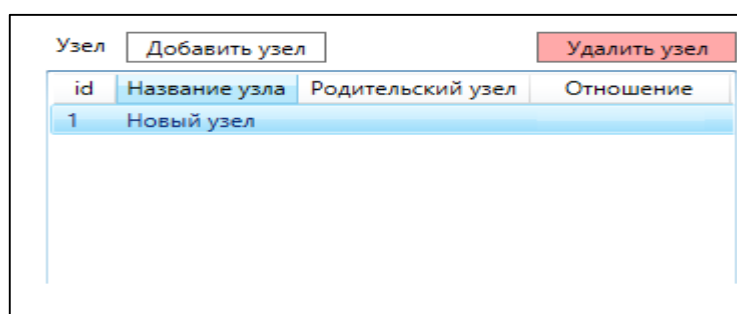


Рисунок 20 – Создание нового узла

После того, как новый узел будет создан, для создания от него дочернего узла, нужно оставить выделенным узел, который будет «главным» и нажать кнопку «Новый узел». Чтобы удостовериться в том, что узел является дочерним, в этой же форме будет указано, какой узел является родительским (рисунок 21).

Узел			
Добавить узел		Удалить узел	
id	Название узла	Родительский узел	Отношение
1	Месяц		
2	Сентябрь	Месяц	

Рисунок 21 – Создание дочернего узла

После создания достаточного количества узлов, можно приступить к заданию связей между ними.

Для того чтобы добавить связь, нужно выделить узел, из формы слева, от которого будет идти связь, нажать кнопку «Добавить связь» и выбрать узел к которому будет идти связь. Если у узла уже был родитель, то он изменится на узел, от которого идет связь.

Чтобы удалить связь, нужно выбрать узел из списка слева, в правом списке (рисунок 22) появится список связанных с узлом «потомков», из этого списка нужно выделить «потомка» и нажать на кнопку «Удалить связь».

id	Название узла	Родительский узел
1	Новый узел	
2	Новый узел	Новый узел
3	Новый узел	Новый узел

id	Название узла
2	Новый узел
3	Новый узел

Добавить связь Удалить связь

Рисунок 22 – Удаление связи

«Удалить все связанные узлы» — аналогично предыдущей команде, только удалятся все «потомки» узла.

Для того, чтобы задать «Отношение», достаточно выбрать любой дочерний узел и вписать отношение в поле с одноименным названием (рисунок 23).

id	Название узла	Родительский узел	Отношение
1	Цвет		
2	Красный	Цвет	Значение

Имя узла:

Отношение:

Рисунок 23 – Пример задания отношения

Создание узлов сопровождается автоматическим расставлением связей и визуализацией (рисунок 24).

The screenshot shows a software interface with three main panels:

- Left Panel (Table):** A table with columns 'id', 'Название узла', 'Родительский узел', and 'Отношение'. It lists 12 nodes, with 'Узел 1' selected.
- Middle Panel (List):** A list titled 'Узлы, связанные с узлом (id):1' showing nodes 2 through 5.
- Right Panel (Diagram):** A 'Карта расположения узлов' (Node Placement Map) showing a hierarchical tree structure of 21 nodes.

Buttons for 'Добавить узел', 'Удалить узел', 'Добавить связь', 'Удалить связь', and 'Удалить все связанные узлы' are visible. Input fields for 'Имя узла' and 'Отношение' are at the bottom.

Рисунок 24 – Узлы, связи и визуализация

Карта расположения узлов активная, нажимая на вершину она выделяется в левой форме, а в центральной форме отображаются связанные с вершиной узлы.

Так же для вершин можно задавать параметры (рисунок 25).

Параметры выбранного узла		Добавить параметр	Удалить параметр
Параметр	Значение		
Новый параметр	Значение нового параметра		
Новый параметр	Значение нового параметра		

Параметр:

Значение параметра:

Рисунок 25 – Форма параметров

Редактор базы знаний выполнен в соответствии с техническим заданием (приложение Б).

3.4 Вывод по главе 3

В третьей главе описаны процессы проектирования и разработки редактора базы знаний. Для проектирования системы использовался унифицированный язык моделирования («UML»). Построены диаграммы, позволяющие показать архитектуру приложения.

В ходе выполнения поставленной задачи, на языке «С#» был разработан «Редактор базы знаний», на технологии «Windows Presentation Foundation» («WPF»). Для разработки системы использован шаблон проектирования «MVVM»

В итоге, выполнена третья задача работы. Все требования, согласно техническому заданию, были выполнены.

ЗАКЛЮЧЕНИЕ

В результате обзора предметной области была выявлена необходимость в редакторе базы знаний, так как база знаний является ключевым блоком системы поддержки принятия решений, а редактор баз знаний играет важнейшую роль в управлении, наполнении и состоянии актуальности БЗ.

В ходе анализа требований технического задания были сформулированы варианты использования и определены прецеденты.

Для проектирования системы использовался унифицированный язык моделирования. Построены диаграммы, позволяющие показать архитектуру приложения.

В ходе выполнения поставленной задачи, на языке «С#» был разработан «Редактор базы знаний», на технологии «Windows Presentation Foundation» («WPF»). Для разработки системы был использован шаблон проектирования «MVVM».

СПИСОК СОКРАЩЕНИЙ

СППР — Система поддержки принятия решения;

БЗ — База знаний;

ИСППР — Интеллектуальная система поддержки принятия решений;

ЗСХН — Земли сельскохозяйственного назначения;

ГПД — Геопространственные данные;

ЭС — Экспертная система;

ИКИТ — Институт космических и информационных технологий;

НУЛ — Научно-учебная лаборатория;

XAML — Extensible application markup language;

XML — Extensible markup language;

UML — Unified modeling language.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1) Система агромониторинга [Электронный ресурс] : гис портал Сухобузимского района Красноярского края. – Режим доступа: <http://activemap.ikit.sfu-kras.ru/>
- 2) Попов, А. Л. Системы поддержки принятия решений: учебное пособие / А. Л. Попов. – Екатеринбург : Урал. гос ун-т, 2008. – 13 с.
- 3) Якушев В. В. Система поддержки принятия решений в земледелии. Принципы построения и функциональные возможности / В. В. Якушев. // Агрофизический НИИ. – 2008. – 10 с.
- 4) Интеллектуальная система поддержки принятия управленческих решений при реализации федеральных и региональных целевых программ [Электронный ресурс] : Самарский государственный технический университет – Режим доступа: <http://vt.samgtu.ru/index.php/predpriyatiam/44-proekt-intellektualnoj-sistemy-podderzhki-prinyatiya-reshenij>
- 5) Раевич К. В. Интеллектуальная система поддержки принятия управленческих решений в задачах оценки земель сельскохозяйственного назначения. / К. В. Раевич; И. В. Зеньков // ВЕСТНИК ИрГТУ – 2016. – № 5. – С. 95-103.
- 6) Верескун В. Д. Информационно-управляющие системы в научных исследованиях и на производстве. / В. Д. Верескун; А. Н. Цуриков // — Ростовский государственный университет путей и сообщения. — 2016. — 53 с.
- 7) Раевич К. В. Разработка системы поддержки принятия решений в управлении использованием земледельческого сектора агропромышленного комплекса в регионах Сибири. / К. В. Раевич; Ю. А. Маглинец; И. В. Зеньков // — 2016 – № 5 – С. 90-97
- 8) Юрин. А. Ю. Редактор баз знаний в формате Clips. / А. Ю. Юрин; М. А. Грищенко // – Журнал «Программные продукты и системы» – 2012 – № 4. – С. 83-87.

9) Фаулер М. UML основы. Краткое руководство по унифицированному языку моделирования. / М. Фаулер. // – 2011. – С. 62-167.

10) Раевич К. В. Интеллектуальная информационная система оценивания земель сельскохозяйственного назначения / К. В. Раевич; Ю. А. Маглинец; Г. М. Цибульский // – Журнал Сибирского федерального университета – 2016. – № 9. – С. 1025-1034.

11) Черняховская Л. Р. Онтологический подход к разработке систем поддержки принятия решений / Л. Р. Черняховская; Р. А. Шкундина; К. Р. Нугаева // – Вестник Уфимского государственного авиационного технического университета. – 2006. – 10 с.

12) ГОСТ 19.201–78 Единая система программной документации. Техническое задание. Требования к содержанию и оформлению. – Введ. 01.01.1980. – Москва : Стандартиформ, 2010.

13) Лаборатория информационных технологий исследования природной и техногенной безопасности [Электронный ресурс] : Проблемно-ориентированный редактор продукционных знаний – Режим доступа: http://www.safety-irk.ru/downloads/POE_Description.pdf

14) Коробова И. В. Методы представления знаний. / И. В. Коробова // – Тамбовский государственный технический университет – 2003. – 12 с.

15) Корончик, Д. Н. Пользовательские интерфейсы интеллектуальных систем / Д. Н. Корончик // Журнал "Кибернетика и программирование". – 2014. – С. 16-22.

16) Гладун А. Я. Репозитории онтологий как средство повторного использования знаний для распознавания информационных объектов. / А. Я. Гладун; Ю. В. Рогушина // – «Онтология проектирования» научный журнал» – 2013 – № 1 – С. 35-50.

17) Петрушин А. Ф. Комплекс программ формирования и обработки баз данных и знаний в агрономии : автореф. дис. канд. технических наук : 05.13.18 / Петрушин Алексей Федорович – Санкт-Петербург, 2005 – С. 4-20.

ПРИЛОЖЕНИЕ А

Диаграмма классов

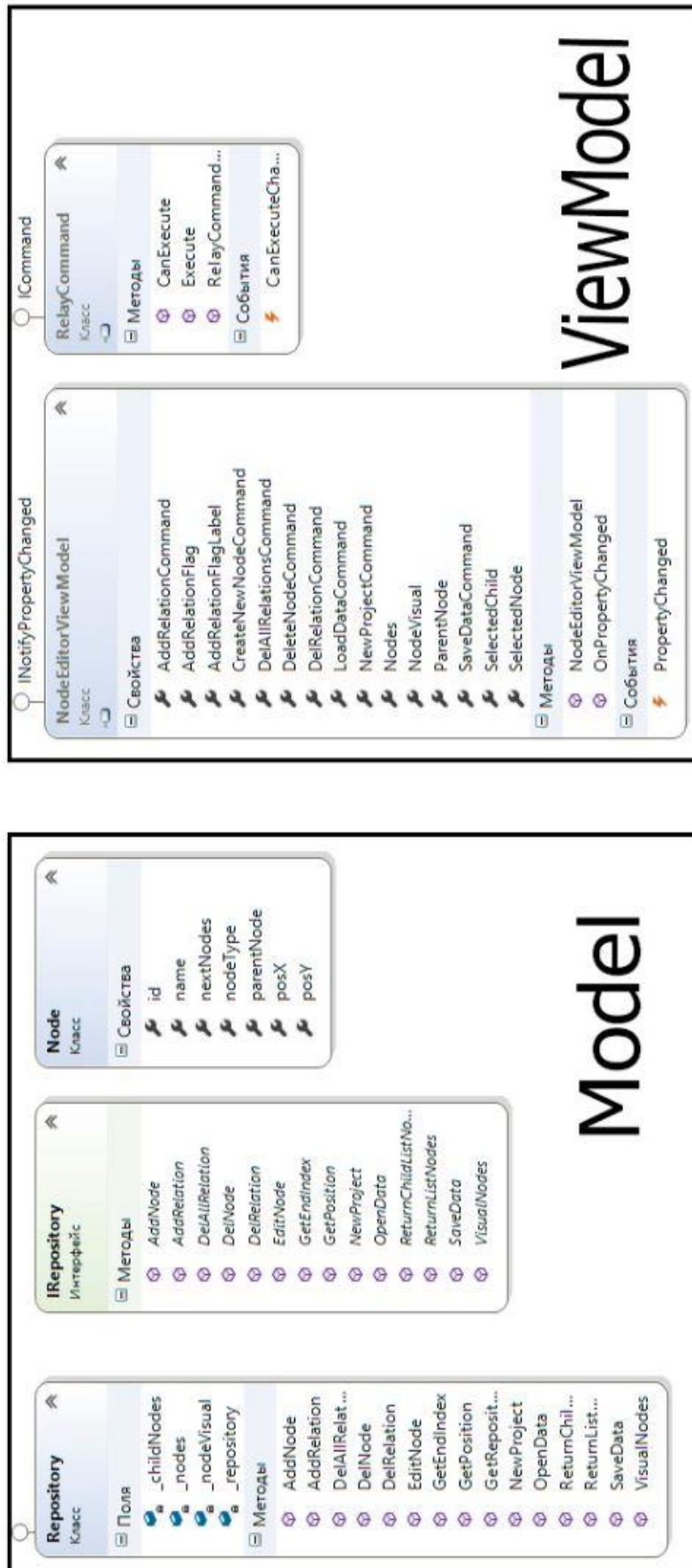


Рисунок А.1 – Диаграмма классов

ПРИЛОЖЕНИЕ Б

Техническое задание

1 Введение

Техническое задание на программу по ГОСТ 19.201-78.

1.1. Наименование программы

«Редактор базы знаний».

1.2 Краткая характеристика области применения

Программа предназначена для упрощения работы с базой знаний в системе поддержки принятия решений.

2 Основания для разработки

Основанием для разработки является полученное задание на бакалаврскую работу от научно-учебной лаборатории кафедры систем искусственного интеллекта института космических и информационных технологий.

2.2 Наименование и условное обозначение темы

Наименование темы разработки — «Редактор базы знаний системы поддержки принятия решений в сельскохозяйственном производстве».

Условное обозначение — «Редактор БЗ».

3 Назначение разработки

3.1 Функциональное назначение

Функциональным назначением программы является предоставление пользователю возможности работы с базой знаний в удобном и понятном виде.

3.2 Эксплуатационное назначение

Программа разрабатывается для НУЛ ИКИТ.

Конечными пользователями являются:

- сотрудники НУЛ;
- Инженеры по БЗ.

4 Требования к программе или программному изделию

4.1 Требования к функциональным характеристикам

4.1.1 Требования к составу выполняемых функций

Программа должна обеспечивать возможность выполнения перечисленных ниже функций:

- создание новой (пустой) базы знаний;
- открытие существующей базы знаний;
- редактирование открытого файла путем ввода, замены, удаления содержимого файла с применением стандартных устройств ввода;
- сохранение содержимого файла;
- визуализация графа на экране монитора;
- отображение параметров вершин графа и узлов.

4.1.2 Требования к организации входных данных

Входные данные программы должны быть организованы в виде отдельных файлов в формате «XML».

Файлы должны храниться на локальных или внешних носителях.

4.1.3 Требования к организации выходных данных

Выходные данные представляют собой список занесенных в БЗ знаний и визуализированный граф.

4.1.4 Требования к временным характеристикам

Требования не предъявляются.

4.2 Требования к надежности

4.2.1 Требования к обеспечению надежного (устойчивого) функционирования программы

Система должна сохранять работоспособность и обеспечивать восстановление своих функций при возникновении внештатных ситуаций.

Для защиты аппаратуры от бросков напряжения и коммутационных помех должны применяться сетевые фильтры и должно быть организовано бесперебойное питание технических средств.

4.2.2 Время восстановления после отказа

Время восстановления после отказа не должно превышать времени, которое требуется для переустановки программных средств и на устранение неисправностей технических средств.

4.2.3 Отказы из-за некорректных действий оператора

Отказы программы возможны вследствие некорректных действий оператора (пользователя) при взаимодействии с операционной системой.

4.3 Условия эксплуатации

4.3.1 Климатические условия эксплуатации

Климатические условия эксплуатации, при которых должны обеспечиваться заданные характеристики, должны удовлетворять требованиям, предъявляемым к техническим средствам в части условий их эксплуатации.

4.3.2 Требования к видам обслуживания

Программа не требует проведения каких-либо видов обслуживания.

4.3.3 Требования к численности и квалификации персонала

Конечный пользователь программы должен обладать навыками работы с графическими интерфейсами.

4.4 Требования к составу и параметрам технических средств

В состав технических средств должен входить персональный компьютер, и средства периферии для него, в составе которого:

- процессор с тактовой частотой не менее 2 мГц;
- оперативная память объемом 2 Гб;
- свободное пространство жесткого диска не менее 10 гб;
- ПО: ОС «Windows 7» и старше; «Net framework 4.5.».

4.5 Требования к информационной и программной совместимости

4.5.1 Требования к информационным структурам и методам решения

Информационная структура файла должна включать в себя текст, содержащий разметку, предусмотренную спецификацией формата «XML».

4.5.2 Требования к исходным кодам и языкам программирования

Исходные коды должны быть реализованы на языке «С#». В качестве интегрированной среды разработки программы должна быть использована «Visual Studio 2015».

4.5.3 Требования к программным средствам, используемым программой

Системные программные средства, используемые программой, должны быть представлены лицензионной локализованной версией операционной системы.

5 Требования к программной документации

Программная документация к разрабатываемой программе не предусматривается.

6 Технико-экономические показатели

6.1 Экономические преимущества разработки

Ориентировочная экономическая эффективность не рассчитывается. Программа распространяется без взимания платы за её использование.

7 Стадии и этапы разработки

7.1 Стадии разработки

Разработка должна быть проведена в три стадии:

- разработка технического задания;
- рабочее проектирование;
- внедрение.

7.2 Этапы разработки

На стадии разработки технического задания должен быть выполнен этап разработки, согласования и утверждения настоящего технического задания.

На стадии рабочего проектирования должны быть выполнены перечисленные ниже этапы работ:

- разработка программы;
- разработка программной документации;
- испытания программы.

На стадии внедрения должен быть выполнен этап разработки, подготовка и передача программы.

7.3 Содержание работ по этапам

На этапе разработки технического задания должны быть выполнены перечисленные ниже работы:

- постановка задачи;
- определение и уточнение требований к техническим средствам;
- определение требований к программе;
- определение стадий, этапов и сроков разработки программы и документации на неё;
- согласование и утверждение технического задания.

На этапе разработки программы должна быть выполнена работа по программированию и отладке программы.

На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованиями к составу документации.

На этапе подготовки и передачи программы должна быть выполнена работа по подготовке и передаче программы и программной документации в эксплуатацию на объектах «Заказчика».


8 Порядок контроля и приемки

8.1 Общие требования к приемке работы

На основании решения «Заказчика» программа передается в эксплуатацию.

ПРИЛОЖЕНИЕ В

Плакаты презентации



СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
SIBIRIAN FEDERAL UNIVERSITY

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
Кафедра систем искусственного интеллекта


БАКАЛАВРСКАЯ РАБОТА

Редактор базы знаний системы поддержки принятия решений
в сельскохозяйственном производстве

Выполнил: студент группы КИ13-14Б
С. И. Черкасов
Руководитель: К. В. Раевич

Красноярск, 2017

Рисунок В.1 – Слайд презентации № 1



СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
SIBIRIAN FEDERAL UNIVERSITY

Цель и задачи работы

Цель:

- разработать редактор базы знаний для системы поддержки принятия решений.

Задачи:

- обзор предметной области;
- проектирование приложения;
- разработка приложения.

Рисунок В.2 – Слайд презентации № 2

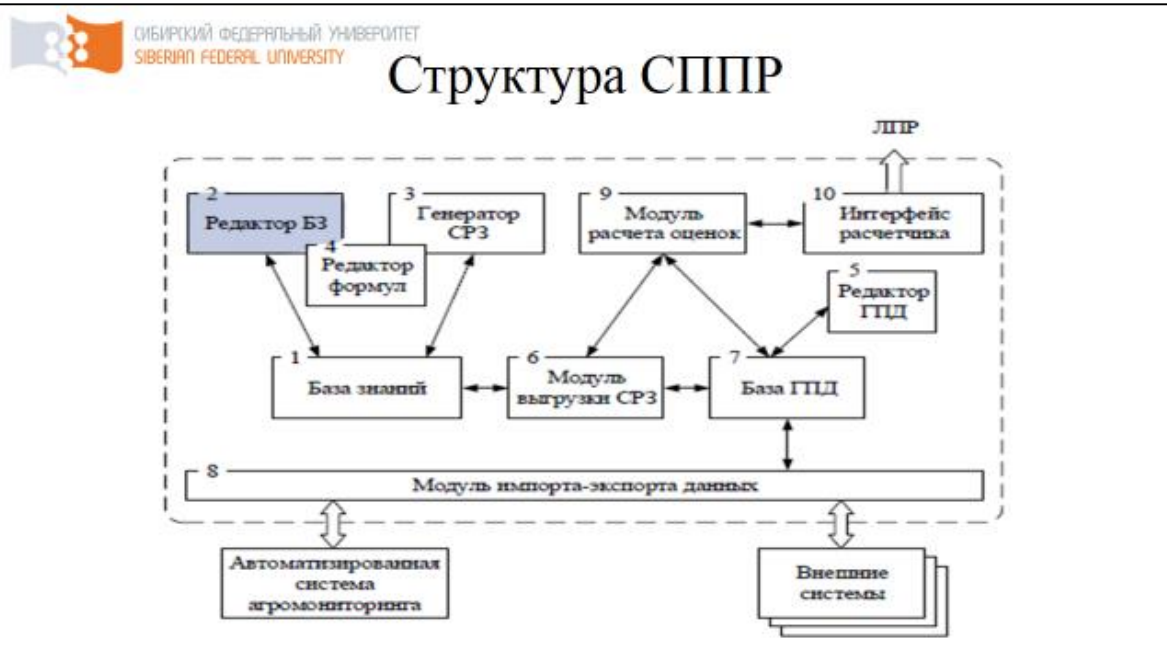


Рисунок В.3 – Слайд презентации № 3



Рисунок В.4 – Слайд презентации № 4

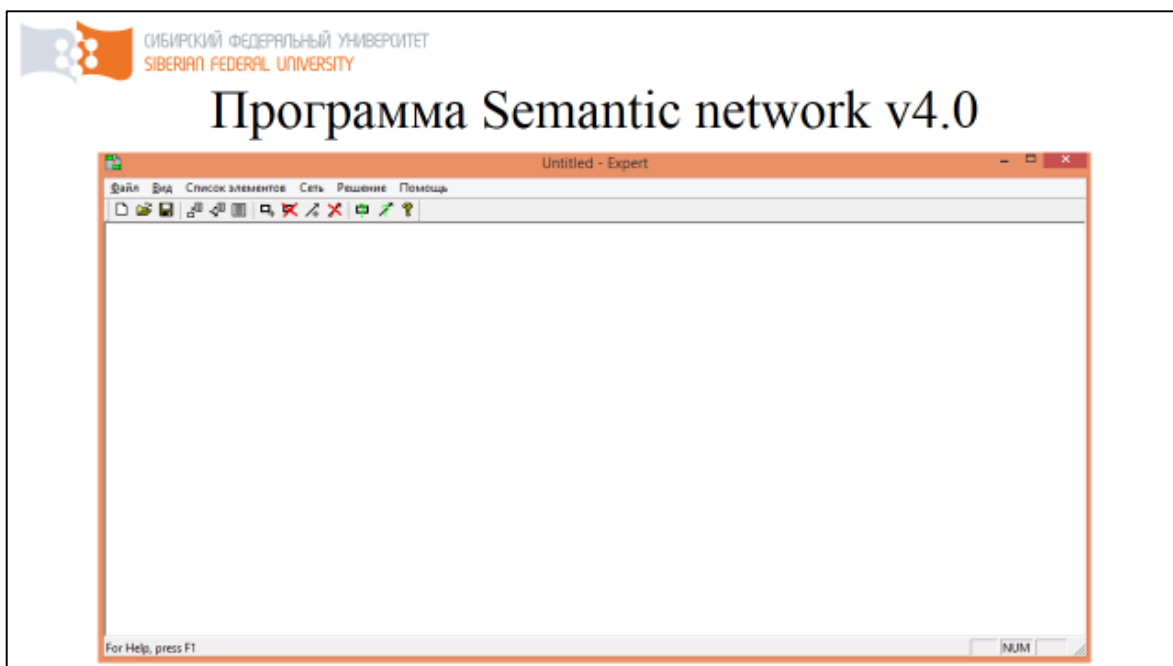


Рисунок В.5 – слайд презентации № 5

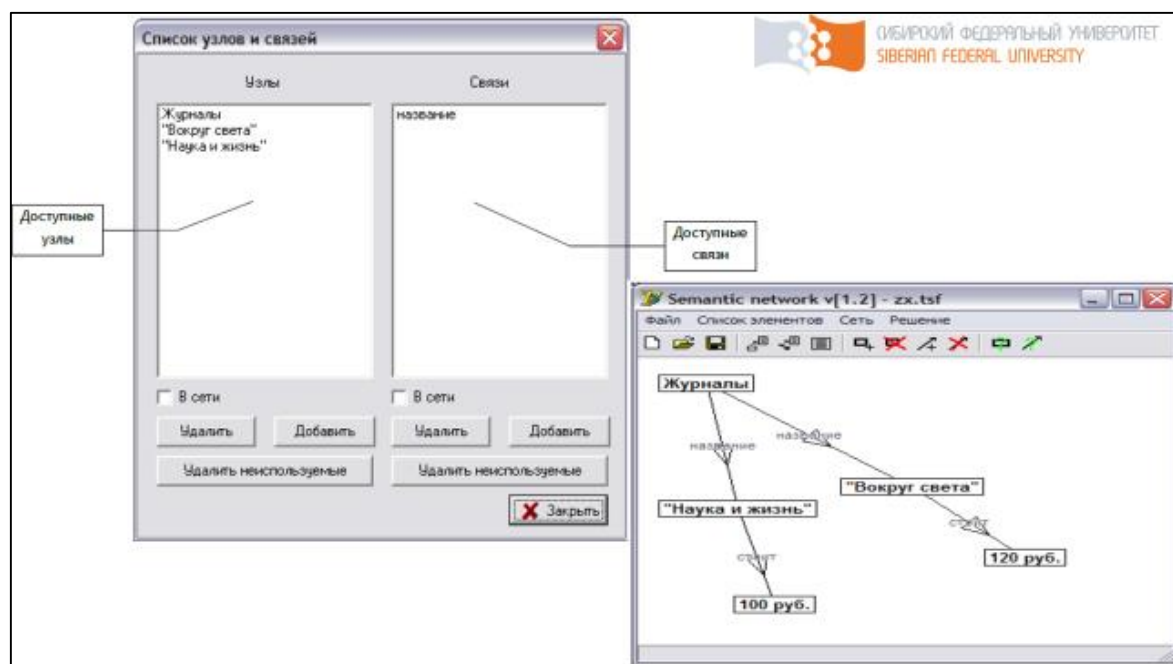


Рисунок В.6 – Слайд презентации № 6



Требования к редактору БЗ

Редактор должен обеспечивать:

- возможность создания и удаления вершин;
- возможность изменять следующие параметры вершин графа: имя, параметры. Имена вершин графа должны быть уникальны;
- вывод на экран монитора графа и параметров его вершин;
- редактирование каждого узла;
- сохранение базы знаний в XML-файле, чтение таких файлов.

Рисунок В.7 – Слайд презентации № 7



Диаграмма вариантов использования

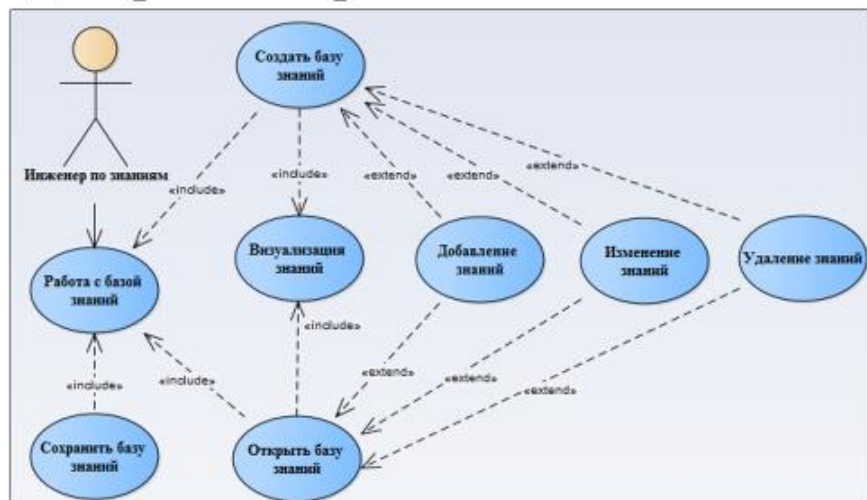


Рисунок В.8 – Слайд презентации № 8



Рисунок В.9 – Слайд презентации № 9



Рисунок В.10 – Слайд презентации № 10

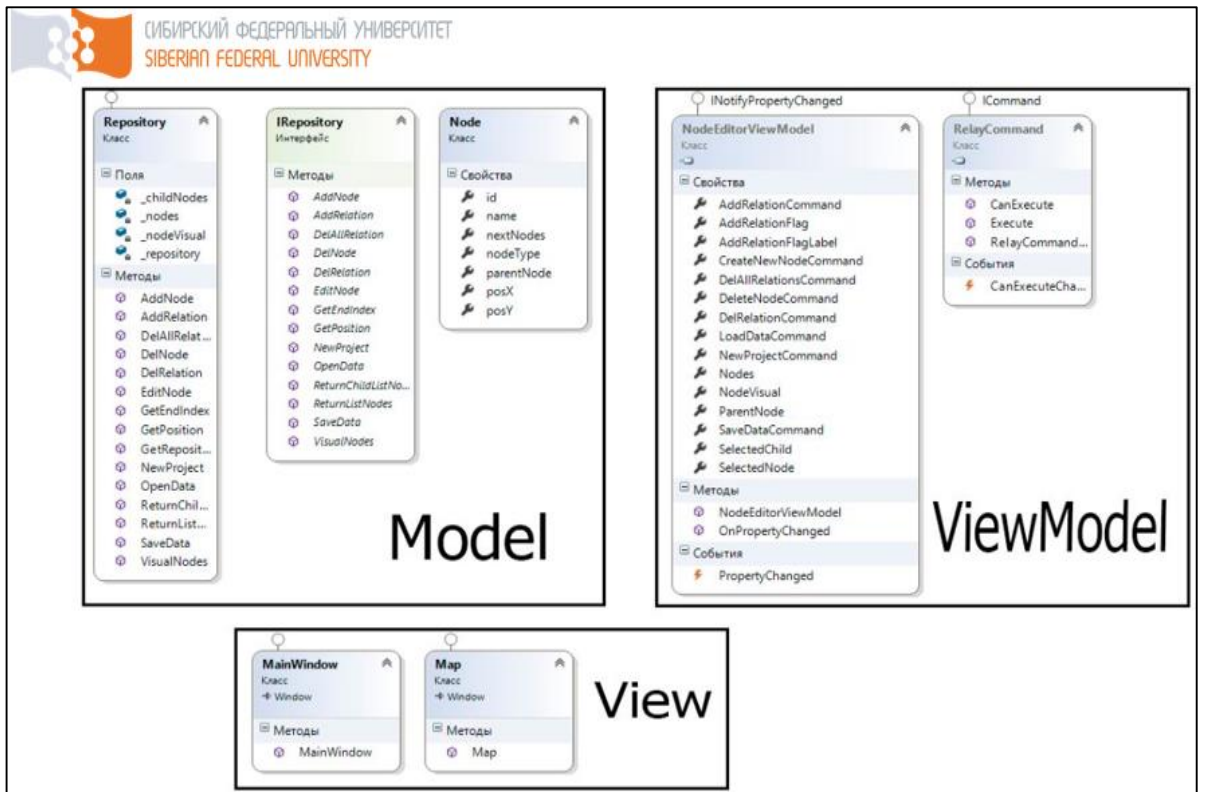


Рисунок В.11 – Слайд презентации № 11

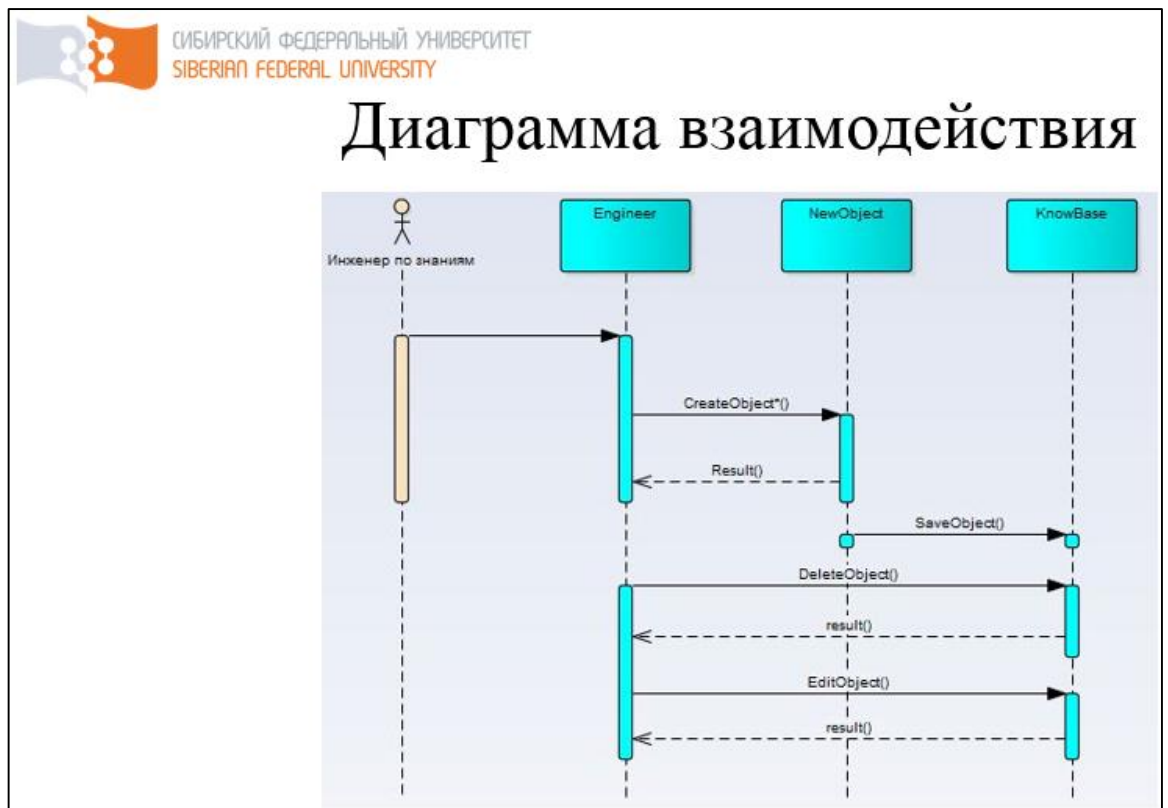


Рисунок В.12 – Слайд презентации № 12

Диаграмма развертывания

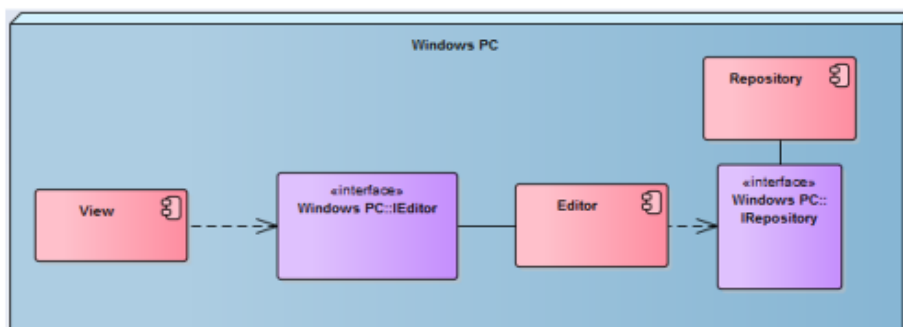


Рисунок В.13 – Слайд презентации № 13

Структура XML-файла

```
<?xml version="1.0"?>
<nodes>
  <node nodeType="Главная" id="1">
    <name>Участок #123</name>
    <previousNode>0</previousNode>
    <childs>2 3</childs>
  </node>
  <node nodeType="Обычная" id="2">
    <name>СоставПочвы</name>
    <previousNode>1</previousNode>
    <childs>3 4</childs>
    <relation>Имеет</relation>
  </node>
  <node nodeType="Обычная" id="3">
    <name>Гумус</name>
    <previousNode>2</previousNode>
    <parameters>
      <parameter>Гумус#132</parameter>
    </parameters>
    <relation>Является</relation>
  </node>
  <node nodeType="Обычная" id="4">
    <name>Джумус</name>
    <previousNode>2</previousNode>
    <parameters>
      <parameter>Джумус#17,2</parameter>
    </parameters>
    <relation>Является</relation>
  </node>
</nodes>
```

Рисунок В.14 – Слайд презентации № 14

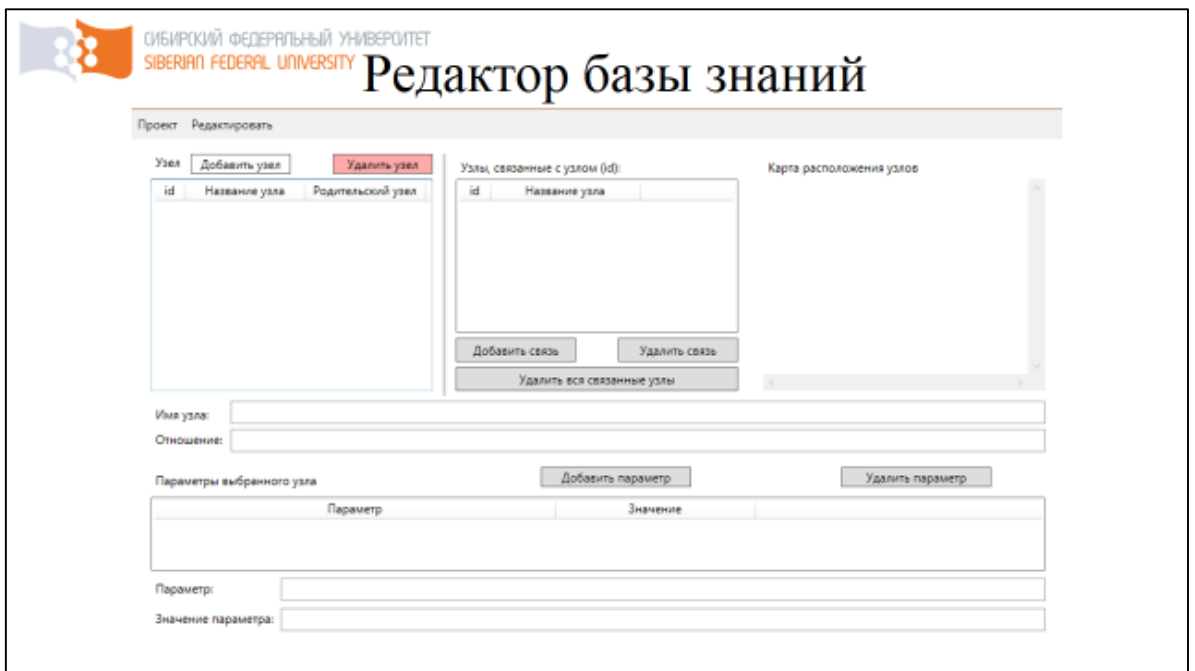


Рисунок В.15 – Слайд презентации № 15

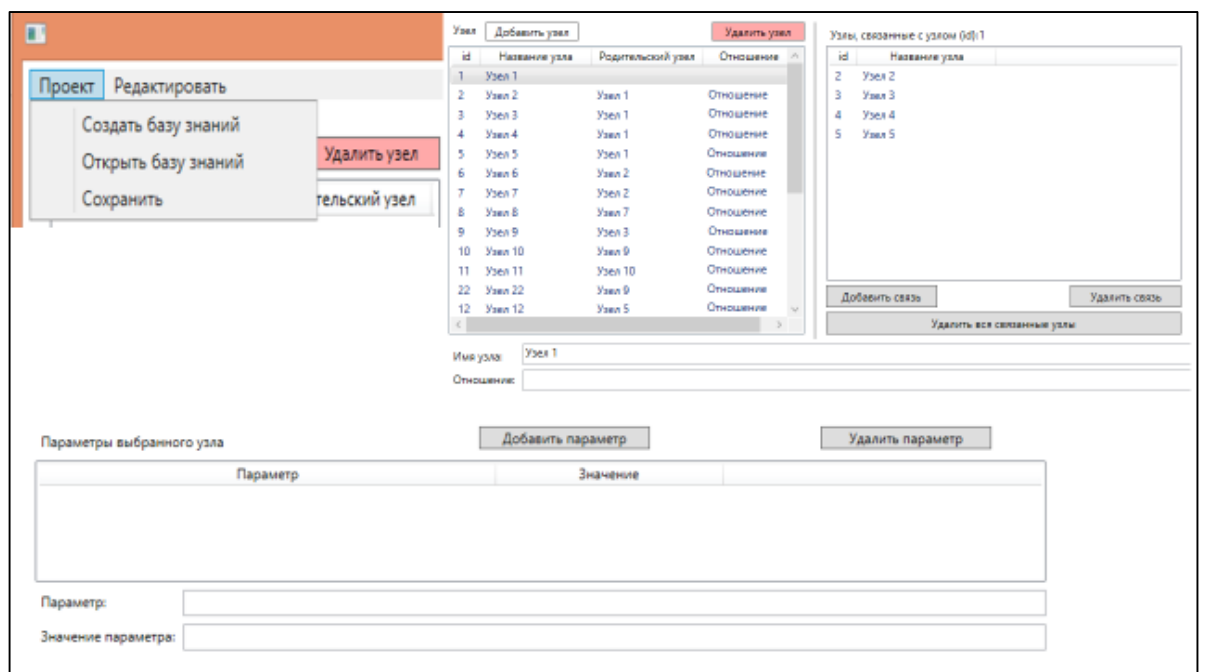


Рисунок В.16 – Слайд презентации № 16

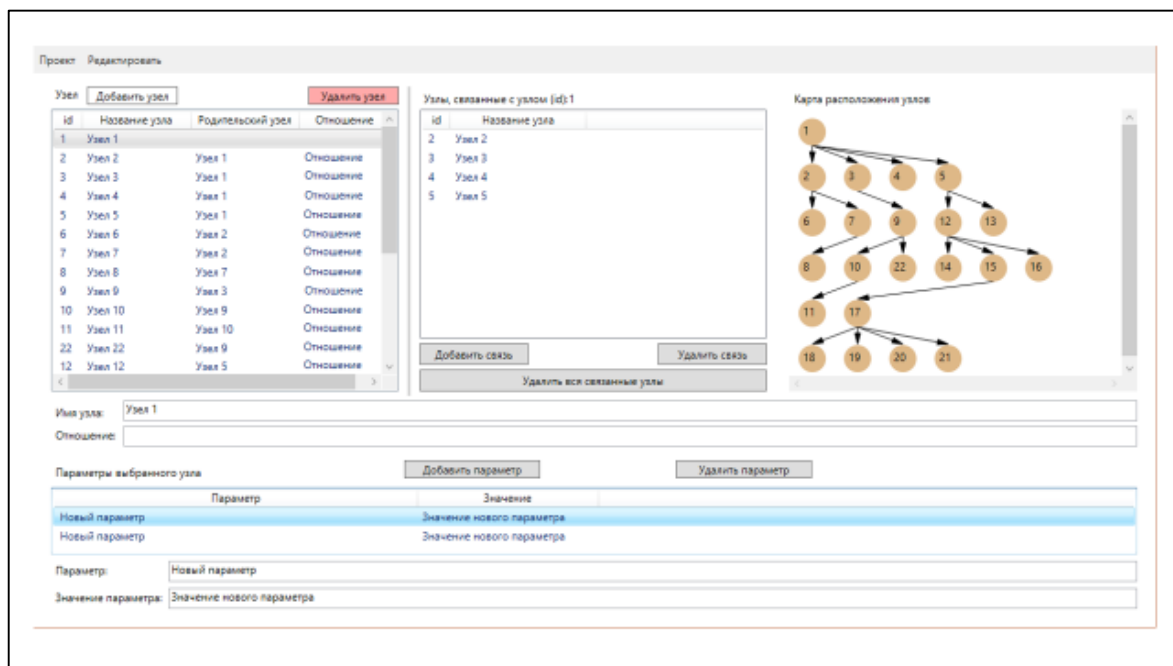


Рисунок В.17 – Слайд презентации № 17

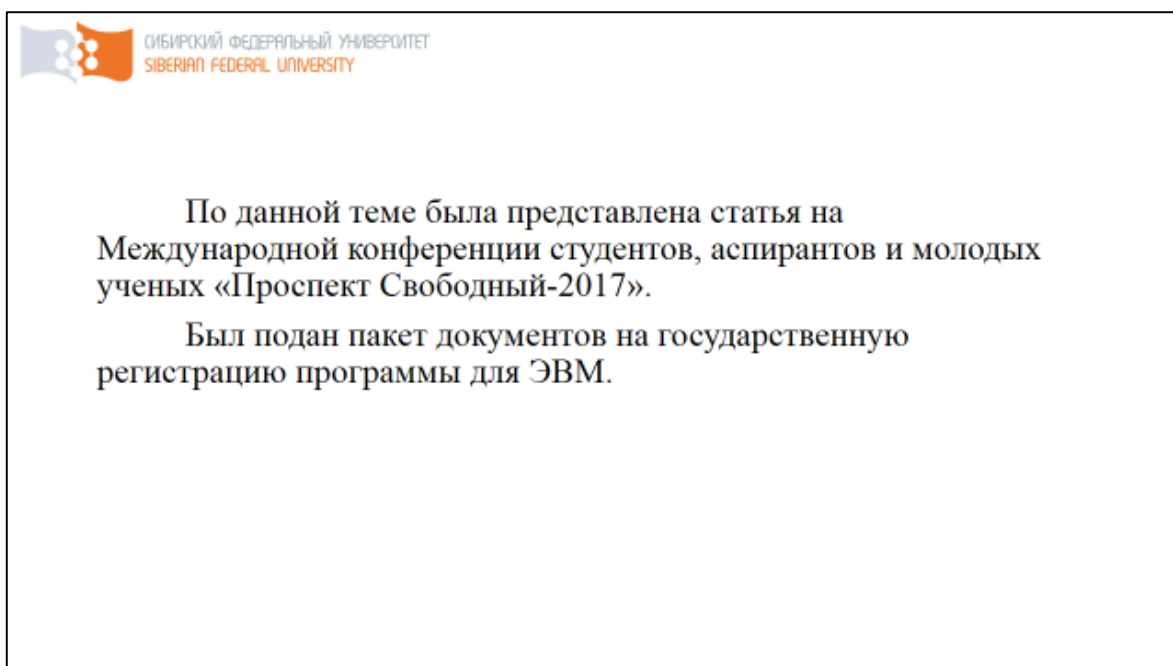


Рисунок В.18 – Слайд презентации № 18

Заключение

В ходе выполнения выпускной квалификационной работы цель была достигнута, путем выполнения задач:

- проведен обзор предметной области.
- спроектирована программа.
- разработан редактор базы знаний

Рисунок В.19 – Слайд презентации № 19