1. Липецкий государственный технический университет  
   Факультет автоматизации и информатики  
   Кафедра автоматизированных систем управления
2. КУРСОВОЙ ПРОЕКТ
3. по программированию.
4. РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА.  
     
   «Разработка графического редактора на языке С++
5. с использованием механизмов ООП»
7. Выполнил студент
8. группы ПИ-21-1
9. Камкин И.Д.
10. « » декабря 2022 года
11. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.
12. Преподаватель:
13. Кургасов.В.В
14. « » декабря 2022 года
15. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

РОССИЯ

г. Липецк, Липецкая область

2022 г.

# 1 Задание кафедры

1. Реализовать на языке С++ редактор графической схемы (табл. 8). В ходе выполнения работы обязательно применение объектно-ориентированных возможностей языка С++: наследования и динамического полиморфизма. Каждый тип элемента схемы должен быть представлен в программе в виде отдельного класса, который унаследован от базового класса "графический элемент" (имеющего чисто виртуальную функцию прорисовки). Также необходим один класс "поле рисования", который содержит все графические элементы и отвечает за вызов функций прорисовки. Хранение графических элементов осуществляется с использованием контейнеров стандартной библиотеки С++.
2. Вариант 6. Редактор инфологических схем.

# 2 Аннотация

С. 15. Ил. 6. Прил. 1;  
 В работе описывается векторный графический редактор для создания и редактирования инфологических схем. Данный редактор позволяет создавать схемы на основе различных элементов, которые можно добавлять на схему, удалять и перемещать. Созданную схему можно сохранить в файл, или загрузить ранее созданную схему. Так же есть возможность экспортировать схему в изображение (jpeg, png, bmp)

1. Расчетно-пояснительная записка составлена в соответствии СТО-13-2016.

**Содержание**

[1 Задание кафедры 2](#__RefHeading___Toc489_7010797402)

[2 Аннотация 3](#__RefHeading___Toc489_70107974011)

[3 Диаграмма классов 5](#__RefHeading___Toc491_7010797401)

[4 Описание программы 7](#__RefHeading___Toc6438_3397910709)

[5 Файловая структура 8](#__RefHeading___Toc6440_3397910709)

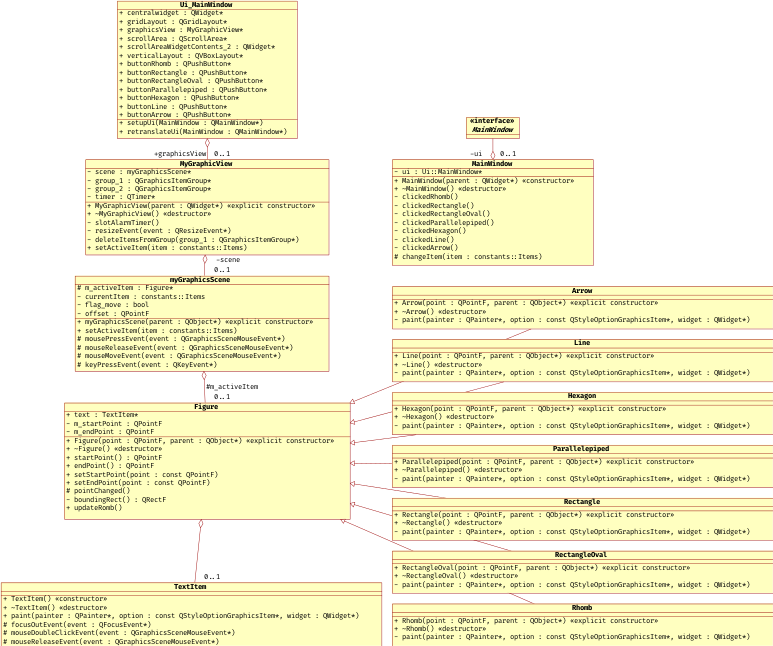
[6 Пример работы программы 10](#__RefHeading___Toc491_7010797403)

[Заключение 13](#__RefHeading___Toc491_7010797405)

[7 Список использованных источников 14](#__RefHeading___Toc6442_3397910709)

[Приложение 15](#__RefHeading___Toc491_70107974051)

# 3 Диаграмма классов

1. Рисунок 1. Диаграмма классов
2. На рисунке 1 представлена диаграмма классов, из которой видно, что логическая часть программы содержит 13 классов.
3. Класс Figure является родителем для остальных классов графических элементов: Arrow, Line, Hexagon, Parallelepiped, Rectangle, RectanleOval, Rhomb. У данного класса реализованы функции для работы с позициями границ отрисовки, но не реализована функция отрисовки, поскольку он является классом-ядром.
4. m\_startPoint — точка, с которой началась отрисовка;
5. m\_endPoint — точка, на которой закончилась отрисовка;
6. startPoint(), endPoint() — методы для получения координат;
7. setStartPoint(), setEndPoint() — методы для установки координат;
8. pointChanged — сигнал изменения размера фигуры;
9. UpdateRhomb — слот обновления области отрисовки.
10. Для фигур определён и реализован метод отрисовки paint, который отрисовывает фигуру на графической сцене.
11. myGraphicsScene — класс, который наследуется от QGraphicsScene, в нём переопределены методы нажатия на сцену мышью, а так же перемещение мыши по сцене.
12. myGraphicsView — класс, который наследуется от QGraphicsView, в нём переопредена функция отрисовки, помимо всех фигур отрисовывается красная обводка вокруг сцены.
13. TextItem — текстовый элемент, который отрисовывается на всех фигурах, кроме Line и Arrow. От стандартного QGraphicsTextItem он отличается шрифтом и размером шрифта.
14. Ui\_MainWindow — автоматически сгенерированный класс фреймворком QT из файла ui.
15. MainWindow — содержит в себе графическую сцену и набор фигур, который расположены списком в левой части экрана.

# 4 Описание программы

1. Программа написана на языке С++ с использованием фреймоврка Qt5 [1]. Она предоставляет пользователю возможность создавать и редактировать инфологические схемы.
2. Основным графическим элементом выступает Figure, от него наследуются основные фигуры инфологических схем. Виджет GraphicsView отрисовывает графическую сцену GraphicsScene, на которой происходит отрисовка всех фигур. TextItem отрисовывается поверх всех элементов, кроме стрелок и линий. По двойному нажатию можно редактировать текст. Каждый элемент можно перемещать зажимая левую клавишу мыши. Если зажать клавишу «Left Ctrl» и перетягивать объект — он скопируется. Графические элементы размещаются мышью, предварительно нужно выбрать его из меню в левой части экрана. Элемент можно удалить, нажав на него правой кнопкой мыши.
3. Сохранить схему можно нажав на кнопку «сохранить». Вводится название файла и происходит сохранение. Загрузить схему так же можно нажав на кнопку «загрузить».
4. Нажав на кнопку «сгенерировать картинку» выбирается название файла и его расширение, после чего создаётся картинка с таким названием.

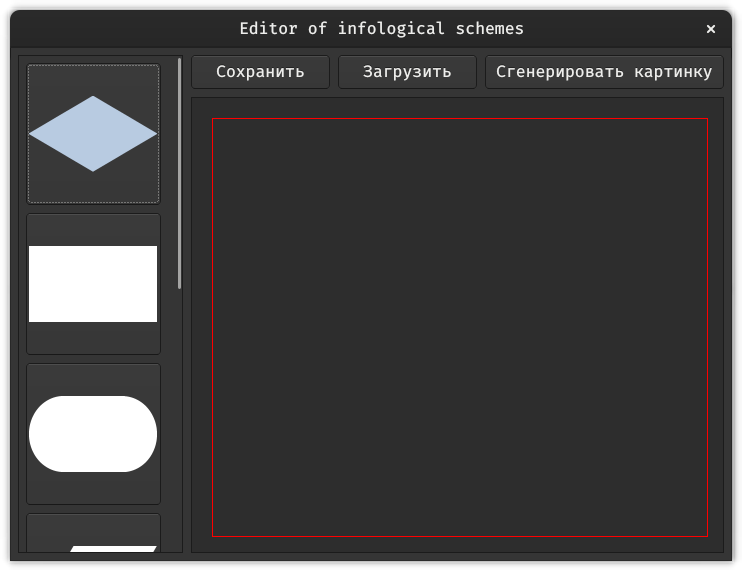
# 5 Файловая структура

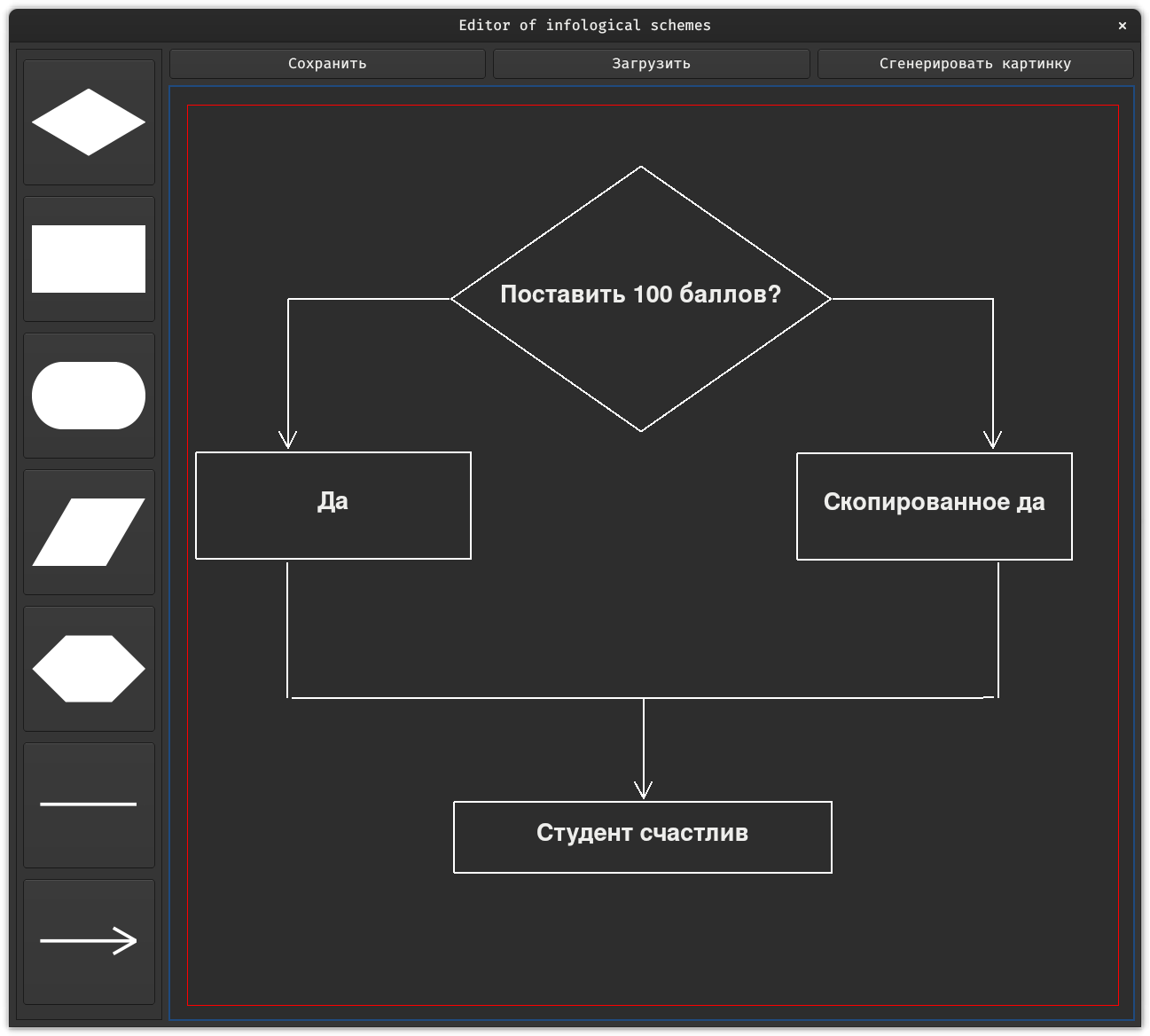
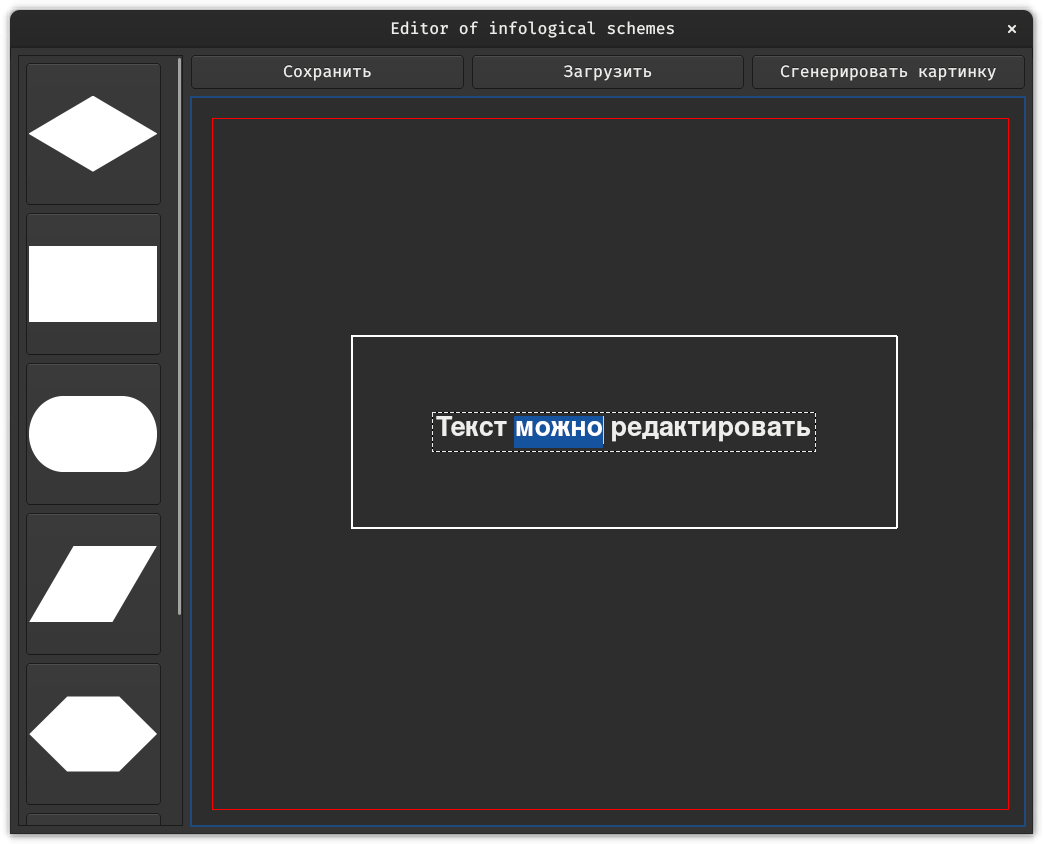
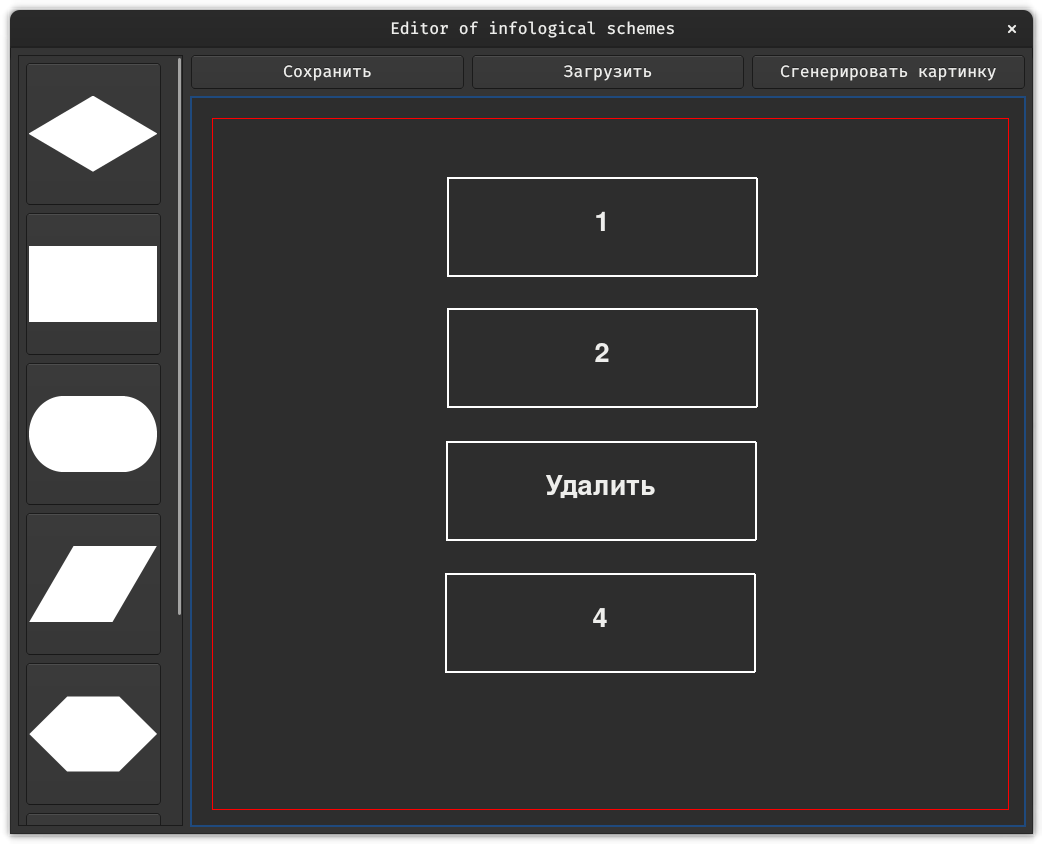
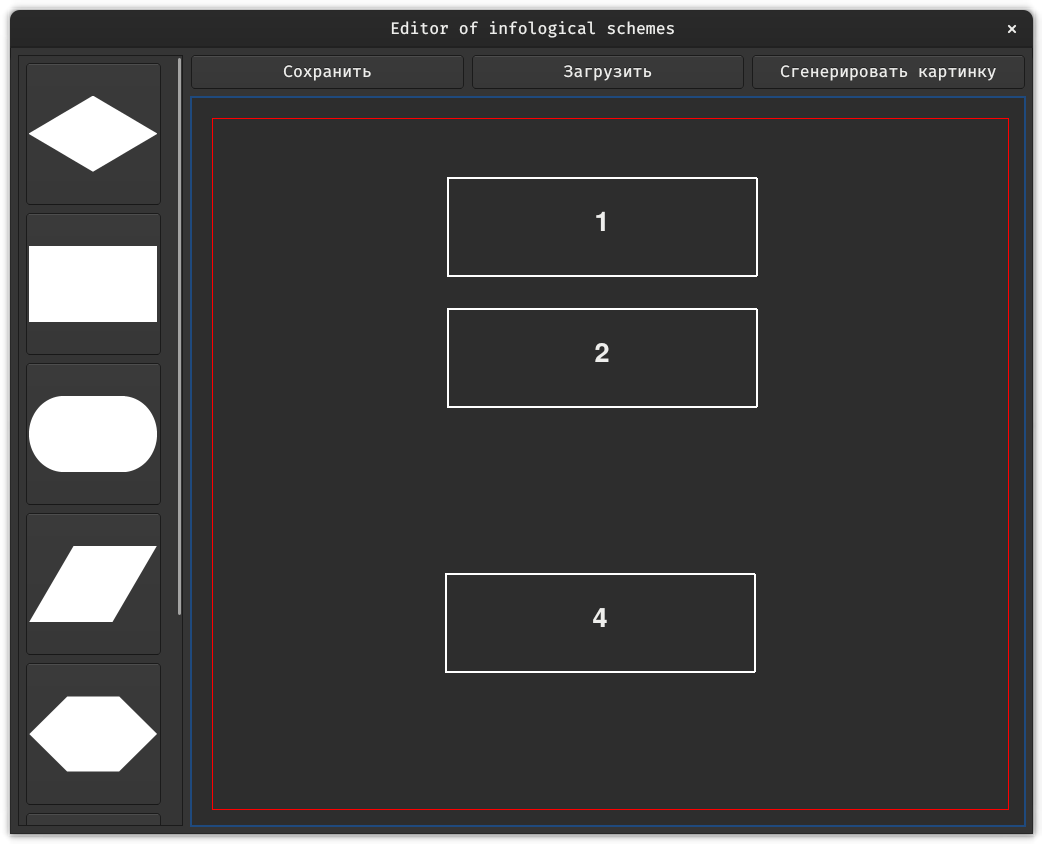
1. Для удобства использования и разработки программа разбита на директории и файлы: заголовочные файлы с++ (.h) и файлы реализации (.cpp), так же присутствует файл описывающий графическое расположение элементов для фреймворка Qt (.ui).

* figures
  + arrow
    - arrow.cpp
    - arrow.h
  + figure
    - figure.cpp
    - figure.h
  + hexagon
    - hexagon.cpp
    - hexagon.h
  + line
    - line.cpp
    - line.h
  + parralelepiped
    - parralelepiped.cpp
    - parralelepiped.h
  + rectangle
    - rectangle.cpp
    - rectangle.h
  + rectangleOval
    - rectangleOval.cpp
    - rectangleOval.h
  + rhomb
    - rhomb.cpp
    - rhomb.h
* main\_window
  + main\_window.cpp
  + main\_window.h
  + main\_window.ui
* myGraphicsScene
  + myGraphicsScene.cpp
  + myGraphicsScene.h
* myGraphicsView
  + myGraphicsView.cpp
  + myGraphicsView.h
* text
  + text.cpp
  + text.h

Из-за данной структуры проекта код будет приложен в виде ссылки на GitHub с указанием коммита, актуального на момент написания курсовой работы.

# 6 Пример работы программы

Рисунок 2. Запуск программы

1. Рисунок 3. Размещение фигур
2. Рисунок 4. Редактирование текста
3. Рисунок 5. Перед удалением элемента
4. Рисунок 6. После удаления элемента

При запуске программы пользователь видит интерфейс (рис. 2) слева располагаются фигуры, которые можно разместить на рабочей области (внутри красного квадрата).

Пользователь так же может изменить размер приложения, потянув за его край.

Пример расположенных графических элементов показан на рисунке 3.

Текст внутри объектов инфологической схемы можно редактировать точно так же, как в системе, выделять, копировать, вырезать, вставлять, и т.д. (рис. 4).

На рисунках 5 и 6 показано удаление элемента. Удаление происходит по нажатию в любую область удаляемого элемента правой кнопкой мыши.

# Заключение

1. В ходе работы было реализовано приложение на языке C++ с использованием фреймворка Qt. Приложение представляет собой графический редактор инфологических схем.
2. При выполнении курсовой работы были получены навыки использования механизмов ООП на примере реализации графического редактора инфологических схем.

# 7 Список использованных источников

1. 1. Документация Qt 5 [Электронный ресурс] – Электрон. дан. – Qt Company 2022 – GNU – режим доступа: https://doc.qt.io/qt-5.15/ свободный.

# **Приложение**

1. GitHub коммит: [be6e0f4583490b7e2991e6102dd34be8929b20be] (https://github.com/MR-Geri/coursework)