Федеральное государственное бюджетное образовательное учреждение высшего образования

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

Кафедра информационных систем и программной инженерии

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту по дисциплине   
"Программирование компьютерной графики"

на тему

Разработка графического приложения на C# с использованием OpenGL.

Предметная область «Аквариум»

Выполнил: студент группы ПРИ-120 Парахин К.В.

Приняла: доц. кафедры ИСПИ

Жигалов И.Е.

Владимир, 2023

**Аннотация**

Курсовой проект выполнен по дисциплине «Программирование компьютерной графики». В данном курсовом проекте было разработано графическое приложения по предметной области «Аквариум», которое формирует заданную двухмерную и трехмерную графическую сцену с интерактивным управлением. При разработке были использованы возможности языка программирования C# и его фреймворка OpenGL по созданию графических приложений, формированию простых графических объектов из примитивов, применению геометрических преобразований, использованию цветов и текстур, анимации сцены и управлению сценой.

В ходе выполнения данного курсового проекта были выполнены следующие задачи:

1. Анализ предметной области;
2. Формирование технического задания;
3. Разработка интерфейса приложения;
4. Разработка 3D моделей обитателей аквариума;
5. Реализация приложения с использованием заданных технологий

**ABSTRACT**

The course project was carried out in the discipline "Computer graphics programming". In this course project, a graphical application for the subject area "Aquarium" was developed, which forms a given two-dimensional and three-dimensional graphic scene with interactive control. During the development, the capabilities of the C# programming language and its OpenGL framework were used to create graphical applications, form simple graphical objects from primitives, apply geometric transformations, use colors and textures, animate the scene and control the scene.

In the course of this course project , the following tasks were completed:

1. Domain analysis;

2. Formation of the technical task;

3. Development of the application interface;

4. Development of 3D models of aquarium inhabitants;

5. Implementation of the application using the specified technologies

**Содержание**

[ВВЕДЕНИЕ 4](#_Toc149587192)

[1 ТЕХНИЧЕСКОЕ ЗАДАНИЕ 5](#_Toc149587193)

[1.2. Описание ТЗ 6](#_Toc149587194)

[1.3. Цель разработки 8](#_Toc149587195)

[1.4. Функциональные требования 8](#_Toc149587196)

[1.5. Нефункциональные требования 8](#_Toc149587197)

[1.6. Инструменты разработки 9](#_Toc149587198)

[1.7. Интерфейс приложения 10](#_Toc149587199)

[1.8. Требования к программной документации 12](#_Toc149587200)

[1.9. Стадии и этапы разработки 12](#_Toc149587201)

[1.10. Порядок контроля и приемки 12](#_Toc149587202)

[2 ЦЕЛИ И ЗАДАЧИ ПРОЕКТА 13](#_Toc149587203)

[3 ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ 14](#_Toc149587204)

[3.1. Моделируемая сцена 14](#_Toc149587205)

[3.2. Моделируемые объекты 14](#_Toc149587206)

[3.3. Моделируемые события 15](#_Toc149587207)

[4 ПРОЕКТИРОВАНИЕ 18](#_Toc149587208)

[4.1. Описание архитектуры 18](#_Toc149587209)

[4.2. Описание технологий 18](#_Toc149587210)

[4.3. Описание реализации 19](#_Toc149587211)

[5 РАЗРАБОТКА АЛГОРИТМА И ДИЗАЙНА ПРИЛОЖЕНИЯ 21](#_Toc149587212)

[5.1. Разработка схемы алгоритма приложения 21](#_Toc149587213)

[5.2. Разработка дизайна приложения 21](#_Toc149587214)

[Заключение 22](#_Toc149587215)

[Список использованных источников 23](#_Toc149587216)

[ПРИЛОЖЕНИЕ А. Схема работы алгоритма 24](#_Toc149587217)

# ВВЕДЕНИЕ

Предметом выполнения данного курсового проекта является разработка графического приложения, формирующего графическую двумерную и трехмерную сцены с интерактивным управлением ими – в соответствии с индивидуальной предметной областью «Аквариум» (по варианту №25).

Для этого необходимо на практике использовать основные возможности языка C# и библиотеки OpenGL по созданию графических приложений, формированию простых графических объектов из примитивов, применению геометрических преобразований, использованию цветов, изменению прозрачности, освещения и применению текстурирования, выполнения и разработки анимации сцены, а также по управлению данной сценой.

Также в соотвествии с индивидуальным вариантом работы (№25) необходимо использовать фракталы ромба, растровый фильтр с использованием аппроксимирующей кривой, задающей увеличение резкости. Для поверхности вращения необходимо использовать кривую линию, представляющую собой полином Лагранжа. Тематика сцены заключается в использовании окружения и текстур из предметной области «Аквариум», то есть – рисование сцены с стеклянной емкостью для аквариума, использование текстур воды (капель) и песка на дне аквариума, моделирование существования (или в некоторых случаях – движения) обитателей аквариума – то есть различных моделей рыбоки т.д. (которых преждевременно необходимо разработать для вставки в сцену).

1 ТЕХНИЧЕСКОЕ ЗАДАНИЕ

1.1. Введение в ТЗ

Наименование приложения: Аквариум (Aquarium) – программа моделирования процессов жизнедеятельности обитателей стеклянного домашнего аквариума.

Глоссарий:

Аквариум (Aquarium) – стеклянная емкость с водой и песком, в котором находятся его обитатели – а именно некоторые декоративные домашние животные или растения (в рамках аквариума моделируется их нахождение в море). Аквариум поэтому имеет сбоку морской фон.

Декоративные животные (Decorative animals) – животные, использующиеся для украшения дома – для получения хозяином положительных эмоций.

Рыбки – основной подвид животных, обитающих в аквариуме.

Основные виды аквариумных декоративных рыбок:

Петушки – небольшие синие рыбки, которые выделяются своими веерообразными плавниками.

Гуппи – одни из самых маленьких домашних рыбок, которые при этом считаются одними из самых стойких и нетребовательных к рациону.

Коридорасы – уменьшенные версии обычных пресноводных сомов, сильно зависят от размера аквариума.

Моллюски – небольшие животные, которые имеют панцирь вокруг всего своего тела.

Водоросли – морские растения различной структуры – которые используются в аквариуме.

Кораллы – разноцветные морские беспозвоночные животные – в рамках аквариума могут быть представимы неподвижными, как растения.

1.2. Описание ТЗ

Назначение и цель разработки:

Разрабатываемое графическое приложение моделирует процессы жизнедеятельности животных в питомнике: заплывы аквариумных рыбок (петушков, гуппи, коридорасов, рыб-бабочек и тд), колыхание водорослей и кораллов от перемещения водяных потоков по аквариуму. Рыбок при этом можно покормить – насыпав корм в аквариум.

Краткая характеристика и область применения:

Данное графическое приложение необходимо разработать с использованием шаблона для проекта графического приложения .NET - Windows Forms. В данном графическом приложении представлены процессы жизнедеятельности животных в и растений в аквариуме – а также возможности взаимодействовать с ними, рисовать дополнительные фракталы и добавлять смоделированные трехмерные объекты на форму приложения.

Сценарий работы алгоритма данного графического приложения представлен в виде схемы на рисунке 1.

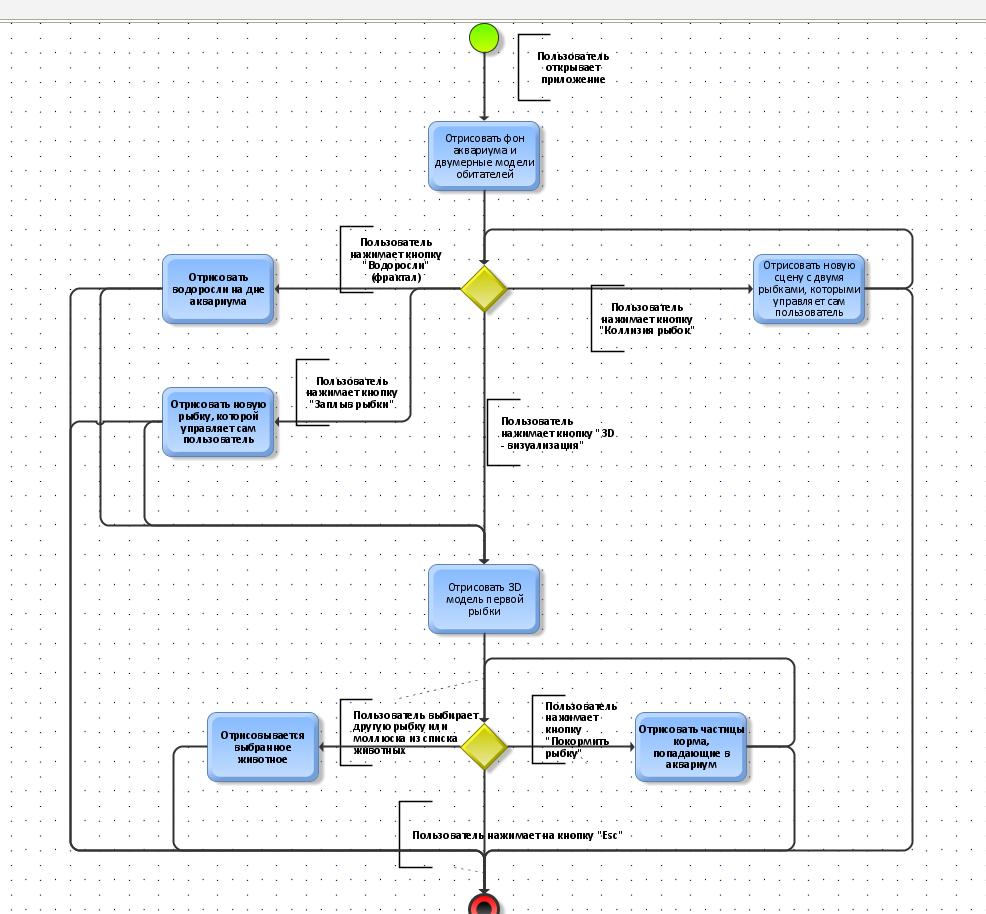


Рисунок 1. Схема работы алгоритма

1.3. Цель разработки

Функциональным назначением программы является наглядное представление и моделирование процессов жизнедеятельности обитателей домашнего аквариума, взаимодействие с ними в различных графических режимах и с использованием растровых графических фильтров.

1.4. Функциональные требования

Программа должна обеспечивать возможность выполнения перечисленных ниже функций:

* визуализация плавания рыбок и моллюсков;
* управление выбранным животным с помощью клавиш клавиатуры (клавиш WASD);
* визуализация фона аквариума в виде фрактала (фрактала-ромба)
* визуализация фона в виде картинки\спрайта;
* визуализация фона с использованием растрового фильтра, настраивающего резкость изображения
* визуализация животных (и растений) в 3D;
* изменение масштаба при просмотре визуализации обитателей аквариума в режиме 3D.

1.5. Нефункциональные требования

Требования к реализации: исходные коды программы должны быть реализованы на языке C#. В качестве интегрированной среды разработки программы должна быть использована среда Microsoft Visual Studio 2022. Использование открытой графической библиотеки OpenGL (а именно библиотеки фреймворка TAO, такие как: Tao.OpenGL.dll, Tao.FreeGlut.dll, Tao.Platform.Windows.Dll).

Требования к производительности: визуализация сцен в приложении должна происходить без зависаний.

Требования к интерфейсу: пользовательский интерфейс должен быть доступным и понятным.

Требования к установке: разрабатываемое приложение не должно требовать установки на компьютер, достаточно запустить исполняемый .exe файл.

Требования к составу и параметрам технических средств: в состав технических средств должен входить персональный компьютер.

Требования к информационной и программной совместимости: системные программные средства, используемые программой, должны быть представлены операционной системы Windows.

1.6. Инструменты разработки

Visual Studio 2022 (C#) – интегрированная среда разработки программного обеспечения. Данный продукт позволяет разрабатывать приложения на основе проектов типа Windows Forms приложения и включает в себя редактор исходного кода с поддержкой технологии IntelliSense и возможностью простейшего рефакторинга кода. Остальные встраиваемые инструменты включают в себя редактор форм для упрощения создания графического интерфейса приложения.

OpenGL — это спецификация, включающая в себя несколько сотен функций. Она определяет независимый от языка программирования кросс платформенный программный интерфейс, с помощью которого программист может создавать приложения, использующие двухмерную и трехмерную компьютерную графику.

Blender (или Blender 3D) — профессиональное свободное и открытое программное обеспечение для создания трёхмерной компьютерной графики, включающее в себя средства моделирования, скульптинга, анимации, симуляции, рендеринга, постобработки и монтажа видео со звуком, компоновки с помощью «узлов», а также создания 2D-анимаций.

1.7. Интерфейс приложения

Основные элементы интерфейса для пользовательского взаимодействия с программой — это кнопки, выпадающий список и ползунки окна – соответствующие элементам управления OC Windows.

Кнопка «Фрактал (водоросли)» обрисовывает фрактал.

Кнопка «Заплыв рыбки» позволяет создать аквариумную рыбку, плаванием которой можно управлять с помощью клавиш WASD.

Кнопка «3D-визуализация» позволяет рассмотреть объекты сцены в режиме 3D. Также этом помогает список с выбором различных видов рыбок (например, петушки, гуппи, коридорасы и тд).

Кнопка «Покормить рыбку» позволяет посмотреть демонстрацию работы частиц (в данном случае частиц корма).

Также присутствуют 2 кнопки: «Выбрать рыбку из списка» и «Выбрать моллюска из списк» - с помощью которых можно добавить на сцену модель рыбки или моллюска на существующую 3D – сцену.

Кнопка «Покормить рыбок» доступны только при визуализации 3D объектов.

Кнопка «Коллизия рыб» позволяет взять управление за двух разных сущестующих рыбок, которые могут столкнуться.

На рисунке 2 представлен интерфейс программы при запуске приложения.

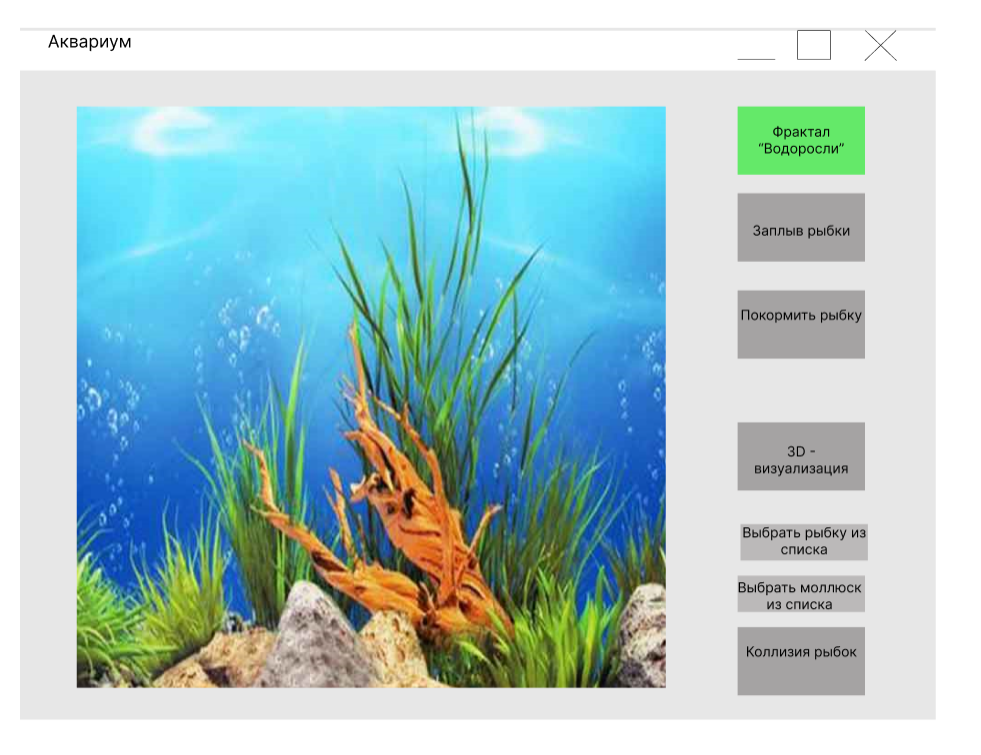


Рисунок 2. Макет интерфейса (дизайн) приложения

1.8. Требования к программной документации

Состав программной документации должен включать в себя:

1) Назначение программы;

2) Условия выполнения программы;

3) Выполнение программы.

1.9. Стадии и этапы разработки

Разработка должна быть проведена в четыре стадии:

1) разработка технического задания;

2) моделирование необходимых объектов 3D сцен;

3) разработка приложения;

4) тестирование;

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

На стадии моделирования должны быть созданы объекты для 3D сцен.

На стадии разработки приложения должен быть написан программный код и документация к нем.

На стадии тестирование должно быть выполнено испытания программы.

1.10. Порядок контроля и приемки

Критерии приемки:

Визуализация сцен в приложении должна происходить быстро (меньше 1 секунды), без зависаний.

Переключение сцен и отрисовка новой сцены должны занимать менее 1 секунды.

Преобразования объектов должно быть плавным, без зависаний.

# 2 ЦЕЛИ И ЗАДАЧИ ПРОЕКТА

Основная цель выполнения курсового проекта по дисциплине «Программирование компьютерной графики» состоит в освоении средств разработки приложений в среде Microsoft Visual Studio C# с использованием графической библиотеки OpenGL.

Основной задачей проекта является разработка графического приложения по предметной области «Аквариум», которое формирует заданную двухмерную и трехмерную графическую сцену с интерактивным управлением. При разработке должны использоваться возможности C# и OpenGL по созданию графических приложений, формированию простых графических объектов из примитивов, применению геометрических преобразований, использованию цветов и текстур, анимации сцены и управлению сценой.

3 ОПИСАНИЕ И МОДЕЛИРОВАНИЕ ОБЪЕКТОВ ПРЕДМЕТНОЙ ОБЛАСТИ

Во время выполнения курсового проекта необходимо разработать следующие графические сцены:

1. Двухмерная графическая сцена с плаванием рыбок и моллюсков, заплывом рыбки пользователя и отрисовкой водорослей в фиде фрактала.
2. Трехмерная графическая сцена с использованием функции 3D-визуализации, отрисовкой новых рыбок (или моллюсков) после их добавления на сцену, а также отрисовкой частиц корма животных.

3.1. Моделируемая сцена

Двухмерная сцена должна включать в себя простые объекты: точки (голова и тело), прямые и кривые линии, полигоны и так далее. Цветы выполнены с помощью фрактала «фрактал-ромб». Также необходимо применить растровый фильтр с использованием аппроксимирующей кривой, задающей увеличение резкости. Для поверхности вращения нужно использовать модель кривой линии, представляющую собой полином Лагранжа.

Трехмерная сцена должна включать в себя визуализированные 3D-объекты двумерных полигонов и многоульников, объектов сфер и поверхностей вращения (с использованием функциональной привязки к полиному Лагранжа).

3.2. Моделируемые объекты

Для внедрениея в объект трехмерных сложных объектов – необходимо ихразработать. Цель моделирования – разработать следующие объекты: аквариумная рыбка (модели гуппи и рыбки-петушка), модель моллюска в раковине, модели коралла (или водорослей). Данные модели были созданы с помощью различных сложных объектов типа Mesh (то есть полнотелых 3-х мерных объектов), 3-х мерных сложных объектов вращения (с использованием кривых типа полином Лагранжа и полином Безье), а также NURBS-моделей - разработанных в программе Blender 3D. Далее данные графические модели необходимо импортировать в проект C# Windows Forms.

3.3. Моделируемые события

Все моделируемые события представлены в таблице 1.

Таблица 1 - Моделируемые события

|  |  |  |
| --- | --- | --- |
| Событие | Предусловие | Описание |
| Пользователь запустил приложение | Первый запуск приложения | Отрисовалась сцена с аквариумом, нарисовалось дно с несколькими моллюсками и пятью плавающими в воде рыбками |
| Пользователь нажал кнопку «Фрактал (водоросли)» | Приложение запущено | Кнопка «Фрактал (водоросли)» обведена синей рамкой, цвет кнопки стал зеленым. В окне визуализирован фрактал. |
| Пользователь нажал «Заплыв рыбки» | Приложение запущено | Кнопка «Заплыв рыбки» обведена синей рамкой, цвет кнопки стал зеленым. В окне визуализации появилась черная точка – модель рыбки. Пользователь может управлять данной рыбкой |
| Пользователь нажал кнопку «Коллизия рыбок» | Приложение запущено, в окне отрисованы хотябы пару рыбок | Кнопка «Коллизия рыбок» обведена синей рамкой, цвет кнопки стал зеленым. В окне приложения у пользователя появилась возможность управлять двумя рыбками независимо друг от друга |
| Пользователь нажал «3D-Визуализация» | Приложение запущено | Кнопка «3D-Визуализация» обведена синей рамкой, цвет кнопки стал зеленым. В окне визуализации отрисовывается новая сцена с существующими аквариумнымии животными и растениями |
| Пользователь нажал кнопку «Добавить новую рыбку из списка» (или аналогичную кнопку «Добавить моллюска из списка») | Список доступен (пользователь нажал кнопку «3D-визуализация») | Отрисовывается выбранная рыбка (или моллюск) – они начинют двигаться вдоль аквариума (по течению воды в нем) |
| Пользователь нажал «Покормить рыбку» | Была нажата кнопка «3D-Визуализация» | Кнопка «Покормить рыбку» обведена синей рамкой, цвет кнопки стал зеленым. Пользователь может нажать на рыбку в окне визуализации и покормить ее кормом (используется система частиц) |

4 ПРОЕКТИРОВАНИЕ

4.1. Описание архитектуры

Разработанное приложение должно будет представлять собой событийно-ориентированное приложение. При его разработке нужно будет ориентироваться на использование на платформе Microsoft .NET Framework (версии 4.7 и ниже).

Большая часть времени работы приложения будет свзяана с ожиданием событий от пользователя, связанных с взаимодействием его с формой, то есть, например, с нажатием левой кнопки на мышина форме.

Главным классом приложения, который занимается обработкой всех событий, происходящих на форме будет стандартный класс Form1.

Функции OpenGL по спефикации необходимо реализовать в модели клиент-сервер. Приложение выступает в роли клиента – оно вырабатывает команды, а сервер OpenGL интерпретирует и выполняет их.

OpenGL является прослойкой между аппаратурой и пользовательским уровнем, что позволяет предоставлять единый интерфейс на разных платформах, используя возможности аппаратной поддержки.

4.2. Описание технологий

Реализацию графического приложения необходимо выполнить на языке программирования C#, с помощью библиотеки Tao Framework – одной из удобных реализаций стандарта OpenGL.

Язык C# (язык для разработки) — объектно-ориентированный язык программирования. Данный язык относится к семье языков с C-подобным синтаксисом, из них его синтаксис наиболее близок к C++ и Java. Язык имеет статическую типизацию, поддерживает полиморфизм, делегаты, атрибуты, события, переменные, свойства, обобщённые типы и методы, итераторы, анонимные функции с поддержкой замыканий, LINQ, исключения, комментарии в формате XML.

Tao Framework – это библиотека, предоставляющая разработчикам .NET и Mono доступ к возможностям популярных библиотек вроде OpenGL и SDL. Она изначально была создана программистом на C# и OpenGL Randy Ridge, и с тех пор значительно расширена другими разработчиками.

Blender (или Blender 3D) — профессиональное свободное и открытое программное обеспечение для создания трёхмерной компьютерной графики, включающее в себя средства моделирования, скульптинга, анимации, симуляции, рендеринга, постобработки и монтажа видео со звуком, компоновки с помощью «узлов», а также создания 2D-анимаций.

4.3. Описание реализации

Чтобы использовать библиотеку Tao Framework, нужно проделать следующии шаги:

Установить .NET Framework (с версией не нежие 4.7);

Установить последнюю версию Tao Framework (версия 3.0.0 и выше);

Создать новый проект типа Winsows Forms с использованием IDE Microsoft Visual Studio 2022;

Добавить ссылки на необходимые графические библиотеки (ниже представлен их список) в созданный ранее Windows Forms проект (которые преждевременно требуется скачать и установить в папку ProgramFiles/TAO/bin – а затем добавить в переменную среды Path на своем компьюетере).

Список требуемых графических библиотек:

* Tao.OpenGL.dll - реализация библиотеки OpenGL;
* Tao.FreeGlut.dll - реализация функций библиотеки Glut;
* Tao.Platform.Windows.dll - поддержка элементов для визуализации на платформе Windows;
* Tao.Devll.dll – отвечает за загрузку текстур;
* DevIL.dll - портативная библиотека, отвечающая за подгрузку текстур;
* freeglut.dll - портативная библиотека, отвечающая за создание контекста OpenGL.

5 РАЗРАБОТКА АЛГОРИТМА И ДИЗАЙНА ПРИЛОЖЕНИЯ

5.1. Разработка схемы алгоритма приложения

Работа приложения должна начинаться с инициализации библиотек и настроек программы и ее интерфейса. Все эти действия необходимо совершать в методе Form1\_Load класса Form1. В конце этого метода вызывается метод Start() у объектной с названием переменной RenderTimer, которая является бесконечным циклом итераций таймера (или тиков – ticks). При этом после каждого тика в данном графическом приложении происходит отрисовка сцены в методе Draw класса Form1, а также остальных методов, подобных ему (с использованием остальных классов приложения). В методе Clear класс Form1 происходит очищение буферов графической сцены (например, буферов цвета и глубины) – а затем каждый раз в методе Draw происходит прорисовка новых полигонов - в зависимости от состояния условий, выставленных интерфейсом.

5.2. Разработка дизайна приложения

Для разработки дизайна приложения (в соотвествии с разработанным макетом интерфейса приложения – рисунок 2) планируется использовать стандартные цвета Windows Forms (то есть светых или серых цветов графической палитры).

Кнопки - серые или зеленые, если активированы, с более темной обводкой, а если не активны, то становятся ощутимо темнее стандартного цвета.

Ползунки изменения масштаба, также нужно выполнить в стандартном цветном стиле.

# Заключение

В ходе выполнения курсового проекта была разработано графическое приложение типа Windows Forms на языке C# с использованием библиотеки TAO.OpenGL в IDE Microsoft Visual Studio.

# Список использованных источников

Рекомендуемая литература

|  |  |
| --- | --- |
| 1. | Жигалов И.Е., Новиков И.А. Программирование компьютерной графики / Влад.гос.ун-т. Владимир, 2014. 96 с. ISBN 978-5-9984-0437-5 |
| 2. | Порев В.Н. Компьютерная графика. – СПб.: БХВ-Петербург, 2002.- 432 с |
| 3. | …. |

ПРИЛОЖЕНИЕ А. Схема работы алгоритма

