

**Министерство науки и высшего образования Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТОМСКИЙ  
ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа: Инженерная школа информационных технологий и робототехники  
Направление подготовки: 09.03.04 «Программная инженерия»  
Отделение: Отделение информационных технологий

**БАКАЛАВРСКАЯ РАБОТА**

| Тема работы   |
|---|
| <b>Программное обеспечение для определения, построения и визуализации двухмерной и трехмерной графики</b> |

УДК 004.42:004.925.82

Студент

| Группа | ФИО                     | Подпись | Дата |
|--------|-------------------------|---------|------|
| 8К51   | Клюшов Павел Дмитриевич |         |      |

Руководитель

| Должность | ФИО                       | Ученая степень,<br>звание | Подпись | Дата |
|-----------|---------------------------|---------------------------|---------|------|
| Доцент    | Ротарь Виктор Григорьевич | к.т.н.                    |         |      |

**КОНСУЛЬТАНТЫ:**

Научный консультант от организации ООО «Интех»

| Должность                  | ФИО         | Подпись | Дата |
|----------------------------|-------------|---------|------|
| Зам. директора ООО «Интех» | Осипов А.В. |         |      |

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

| Должность | ФИО                          | Ученая степень, звание | Подпись | Дата |
|-----------|------------------------------|------------------------|---------|------|
| Доцент    | Подопригора Игнат Валерьевич | к.э.н.                 |         |      |

По разделу «Социальная ответственность»

| Должность | ФИО                         | Ученая степень,<br>звание | Подпись | Дата |
|-----------|-----------------------------|---------------------------|---------|------|
| Доцент    | Винокурова Галина Федоровна | к.т.н.                    |         |      |

**ДОПУСТИТЬ К ЗАЩИТЕ:**

| Руководитель ООП        | ФИО                            | Ученая степень,<br>звание | Подпись | Дата |
|-------------------------|--------------------------------|---------------------------|---------|------|
| Доцент ОИТ<br>ИШИТР ТПУ | Чердынцев Евгений<br>Сергеевич | к.т.н.                    |         |      |

Томск – 2019 г.

## ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ООП

| Код результата | Результат обучения<br>(выпускник должен быть готов)   |
|----------------|---|
| P1             | Применять базовые и специальные естественнонаучные и математические знания в области информатики и вычислительной техники, достаточные для комплексной инженерной деятельности.   |
| P2             | Применять базовые и специальные знания в области современных информационных технологий для решения инженерных задач.  |
| P3             | Ставить и решать задачи комплексного анализа, связанные с созданием аппаратно-программных средств информационных и автоматизированных систем, с использованием базовых и специальных знаний, современных аналитических методов и моделей.   |
| P4             | Разрабатывать программные и аппаратные средства (системы, устройства, блоки, программы, базы данных и т. п.) в соответствии с техническим заданием и с использованием средств автоматизации проектирования.   |
| P5             | Проводить теоретические и экспериментальные исследования, включающие поиск и изучение необходимой научно-технической информации, математическое моделирование, проведение эксперимента, анализ и интерпретация полученных данных, в области создания аппаратных и программных средств информационных и автоматизированных систем. |
| P6             | Внедрять, эксплуатировать и обслуживать современные программно-аппаратные комплексы, обеспечивать их высокую эффективность, соблюдать правила охраны здоровья, безопасность труда, выполнять требования по защите окружающей среды.   |
| P7             | Использовать базовые и специальные знания в области проектного менеджмента для ведения комплексной инженерной деятельности.   |
| P8             | Владеть иностранным языком на уровне, позволяющем работать в иноязычной среде, разрабатывать документацию, презентовать и защищать результаты комплексной инженерной деятельности.  |
| P9             | Эффективно работать индивидуально и в качестве члена группы, состоящей из специалистов различных направлений и квалификаций, демонстрировать ответственность за результаты работы и готовность следовать корпоративной культуре организации.  |
| P10            | Демонстрировать знания правовых, социальных, экономических и культурных аспектов комплексной инженерной деятельности.   |
| P11            | Демонстрировать способность к самостоятельному обучению в течение всей жизни и непрерывному самосовершенствованию в инженерной профессии.   |

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное  
учреждение высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ»**

Школа: Инженерная школа информационных технологий и робототехники  
Направление подготовки: 09.03.04 «Программная инженерия»  
Отделение школы: Отделение информационных технологий

УТВЕРЖДАЮ:  
Руководитель ООП  
\_\_\_\_\_ Чердынцев Е.С.  
(Подпись) (Дата)

**ЗАДАНИЕ**  
**на выполнение выпускной квалификационной работы**

В форме:

бакалаврской работы

(бакалаврской работы, дипломного проекта/работы, магистерской диссертации)

Студенту:

| Группа | ФИО                       |
|--------|---------------------------|
| 8К51   | Клюшову Павлу Дмитриевичу |

Тема работы:

|  |  |
|--|--|
| Программное обеспечение для определения, построения и визуализации двухмерной и трехмерной графики |  |
| Утверждена приказом директора (дата, номер)  |  |

|  |               |
|--|---------------|
| Срок сдачи студентом выполненной работы: | 20.06.2019 г. |
|--|---------------|

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ:**

|  |   |
|--|---|
| <b>Исходные данные к работе</b><br><i>(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к особенностям функционирования (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.).</i> | Работа направлена на создание библиотеки для работы с двухмерной и трехмерной графикой и программного обеспечения, которое будет использовать данную библиотеку для трехмерной и двухмерной визуализации структуры проектов организации |
|--|---|

|  |  |
|--|--|
| <p><b>Перечень подлежащих исследованию, проектированию и разработке вопросов</b><br/> <i>(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе).</i></p> | <ol style="list-style-type: none"> <li>1. Исследование предметной области</li> <li>2. Проектирование и реализация программной библиотеки</li> <li>4. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</li> <li>5. Социальная ответственность</li> </ol> |
|--|--|

|  |   |
|--|---|
| <p><b>Перечень графического материала</b><br/> <i>(с точным указанием обязательных чертежей)</i></p> | <ol style="list-style-type: none"> <li>1. Диаграмма организационной структуры компании</li> <li>2. Диаграммы в нотации IDEF0</li> <li>3. Диаграмма вариантов использования в нотации UML</li> <li>4. Диаграмма в нотации IDEF3</li> <li>5. Диаграммы потоков данных в нотации DFD</li> <li>6. Скриншоты визуализируемых объектов</li> <li>7. Диаграмма классов в нотации UML</li> <li>8. Матрица SWOT</li> <li>9. Таблица трудозатрат на выполнение проекта</li> <li>10. Линейный график работ</li> <li>11. Презентация Microsoft PowerPoint</li> </ol> |
|--|---|

|   |   |
|---|---|
| <p><b>Консультанты по разделам выпускной квалификационной работы</b><br/> <i>(с указанием разделов)</i></p> |   |
| <p style="text-align: center;"><b>Раздел</b></p>  | <p style="text-align: center;"><b>Консультант</b></p> |
| <p>Научный консультант от предприятия</p>   | <p>Осипов А.В.</p>                                    |
| <p>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</p>                                      | <p>Подопригора Игнат Валерьевич</p>                   |
| <p>Социальная ответственность</p>   | <p>Винокурова Галина Федоровна</p>                    |

|  |                      |
|--|----------------------|
| <p><b>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</b></p> | <p>10.02.2019 г.</p> |
|--|----------------------|

**Задание выдал руководитель:**

| Должность | ФИО                       | Ученая степень, звание | Подпись | Дата |
|-----------|---------------------------|------------------------|---------|------|
| Доцент    | Ротарь Виктор Григорьевич | к.т.н.                 |         |      |

**Задание принял к исполнению студент:**

| Группа | ФИО          | Подпись | Дата |
|--------|--------------|---------|------|
| 8К51   | Клюшов П. Д. |         |      |

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное  
учреждение высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа: Инженерная школа информационных технологий и робототехники

Направление подготовки: Программная инженерия

Уровень образования: Бакалавр

Отделение школы: Отделение информационных технологий

Период выполнения: осенний / весенний семестр 2018/2019 учебного года

Форма представления работы:

бакалаврская работа

(бакалаврская работа, дипломный проект/работа, магистерская диссертация)

**КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН**  
**выполнения выпускной квалификационной работы**

|  |                 |
|--|-----------------|
| Срок сдачи студентом выполненной работы: | 20 июня 2019 г. |
|--|-----------------|

| Дата контроля | Название раздела (модуля) / вид работы (исследования)                     | Максимальный балл раздела (модуля) |
|---------------|---|------------------------------------|
| 02.05.2019    | Раздел 1. Исследование предметной области                                 | 25                                 |
| 08.05.2019    | Раздел 2. Проектирование и реализация программных модулей                 | 25                                 |
| 25.05.2019    | Раздел 4. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение | 25                                 |
| 28.05.2019    | Раздел 5. Социальная ответственность                                      | 25                                 |
|               |   |                                    |

Составил преподаватель:

| Должность | ФИО                       | Ученая степень, звание | Подпись | Дата |
|-----------|---------------------------|------------------------|---------|------|
| Доцент    | Ротарь Виктор Григорьевич | к.т.н.                 |         |      |

**СОГЛАСОВАНО:**

| Руководитель ООП     | ФИО                         | Ученая степень, звание | Подпись | Дата |
|----------------------|-----------------------------|------------------------|---------|------|
| Доцент ОИТ ИШИТР ТПУ | Чердынцев Евгений Сергеевич | к.т.н.                 |         |      |

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА  
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И  
РЕСУРСОСБЕРЕЖЕНИЕ»**

Студенту:

|               |                         |
|---------------|-------------------------|
| <b>Группа</b> | <b>ФИО</b>              |
| 8К51          | Клюшов Павел Дмитриевич |

|                            |              |                                  |                                   |
|----------------------------|--------------|----------------------------------|-----------------------------------|
| <b>Школа</b>               | <b>ИШИТР</b> | <b>Кафедра</b>                   | <b>ОИТ</b>                        |
| <b>Уровень образования</b> | Бакалавриат  | <b>Направление/специальность</b> | 09.03.04<br>Программная инженерия |

|   |  |
|---|--|
| <b>Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:</b>   |  |
| 1. <i>Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих</i> | 1. <i>З/п научного руководителя – 33 664 руб. З/п научного консультанта – 50 000 руб. З/п студента на предприятии – 12 500 руб.</i>  |
| 2. <i>Нормы и нормативы расходования ресурсов</i>   | 2. <i>Расходы на электроэнергию – 1032,5 руб.</i>  |
| 3. <i>Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования</i>                                  | 3. <i>Отчисления во внебюджетные фонды – 100 725,47 руб. затраты на спец. Оборудование – 141 396 руб., накладные расходы – 92 638,69 руб.</i>  |
| <b>Перечень вопросов, подлежащих исследованию, проектированию и разработке:</b>   |  |
| 1. <i>Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения</i>   | 1. <i>Описание потребителей продукта.</i><br>2. <i>Конкурентоспособность.</i><br>3. <i>SWOT – анализ.</i>  |
| 2. <i>Планирование и формирование бюджета научных исследований</i>  | 1. <i>Структура работ в рамках научного исследования.</i><br>2. <i>Определение трудоемкости выполнения работ и разработка графика проведения научного исследования.</i><br>3. <i>Бюджет проекта.</i>                       |
| 3. <i>Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования</i>          | 1. <i>Определение интегрального финансового показателя разработки.</i><br>2. <i>Определение интегрального показателя ресурсоэффективности разработки.</i><br>3. <i>Определение интегрального показателя эффективности.</i> |
| <b>Перечень графического материала (с точным указанием обязательных чертежей):</b>  |  |
| 1. <i>Матрица SWOT.</i><br>2. <i>Таблица трудозатрат на выполнение проекта.</i><br>3. <i>Линейный график работ.</i>                         |  |
| <b>Дата выдачи задания для раздела по линейному графику</b>   |  |

**Задание выдал консультант:**

| <b>Должность</b> | <b>ФИО</b>                   | <b>Ученая степень, звание</b> | <b>Подпись</b> | <b>Дата</b> |
|------------------|------------------------------|-------------------------------|----------------|-------------|
| Доцент           | Подопригора Игнат Валерьевич | к.э.н.                        |                |             |

**Задание принял к исполнению студент:**

| <b>Группа</b> | <b>ФИО</b>              | <b>Подпись</b> | <b>Дата</b> |
|---------------|-------------------------|----------------|-------------|
| 8К51          | Клюшов Павел Дмитриевич |                |             |

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА  
«СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»**

Студенту:

|               |                           |
|---------------|---------------------------|
| <b>Группа</b> | <b>ФИО</b>                |
| 8K51          | Клюшову Павлу Дмитриевичу |

|                            |              |                                  |                                   |
|----------------------------|--------------|----------------------------------|-----------------------------------|
| <b>Школа</b>               | <b>ИШИТР</b> | <b>Отделение (НОЦ)</b>           | <b>ОИТ</b>                        |
| <b>Уровень образования</b> | Бакалавриат  | <b>Направление/специальность</b> | 09.03.04<br>Программная инженерия |

Тема ВКР:

|  |  |
|--|--|
| Программное обеспечение для определения, построения и визуализации двухмерной и трехмерной графики   |  |
| <b>Исходные данные к разделу «Социальная ответственность»:</b>   |  |
| 1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения   | Объект исследования – кроссплатформенная программная библиотека для определения, построения и визуализации объектов.<br>Рабочее место – рабочий стол с персональным компьютером в офисном помещении  |
| <b>Перечень вопросов, подлежащих исследованию, проектированию и разработке:</b>  |  |
| <b>1. Правовые и организационные вопросы обеспечения безопасности:</b><br><i>- специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства;</i><br><i>- организационные мероприятия при компоновке рабочей зоны.</i> | <ul style="list-style-type: none"> <li>• Рабочее место при выполнении работ сидя регулируется ГОСТом 12.2.032 –78;</li> <li>• Организация рабочих мест с электронно-вычислительными машинами регулируется СанПиНом 2.2.2/2.4.1340 – 03;</li> <li>• Рациональная организация труда в течение рабочего времени предусмотрена Трудовым Кодексом РФ ФЗ-197.</li> </ul>                         |
| <b>2. Производственная безопасность</b><br>2.1. Анализ выявленных вредных и опасных факторов<br>2.2. Обоснование мероприятий по снижению воздействия   | 1.1 <ul style="list-style-type: none"> <li>• Повышенный уровень электромагнитных излучений;</li> <li>• Отклонение показателей микроклимата;</li> <li>• Недостаточная освещенность рабочей зоны;</li> <li>• Повышенный уровень шума на рабочем месте;</li> <li>• Монотонность труда;</li> <li>• Эмоциональные перегрузки;</li> <li>• Повышенное напряжение в электрической цепи.</li> </ul> |
| <b>3. Экологическая безопасность:</b>  | Анализ негативного воздействия на окружающую природную среду: утилизация компьютеров и другой оргтехники, использованных люминесцентных ламп, мусорных отходов, в том числе бумаги.  |
| <b>4. Безопасность в чрезвычайных ситуациях:</b>   | Возможные чрезвычайные ситуации: <ul style="list-style-type: none"> <li>• Пожар.</li> </ul>  |



|  |  |
|--|--|
| Дата выдачи задания для раздела по линейному графику |  |
|--|--|

**Задание выдал консультант:**

| Должность | ФИО                            | Ученая степень,<br>звание | Подпись | Дата |
|-----------|--------------------------------|---------------------------|---------|------|
| Доцент    | Винокурова Галина<br>Федоровна | К.Т.Н., доцент            |         |      |

**Задание принял к исполнению студент:**

| Группа | ФИО                     | Подпись | Дата |
|--------|-------------------------|---------|------|
| 8К51   | Клюшов Павел Дмитриевич |         |      |

## Реферат

Выпускная квалификационная работа содержит 89 страниц, 25 рисунков, 17 таблиц, 3 приложения и 11 литературных источников.

Ключевые слова: графический визуализатор, программная библиотека для рендеринга объектов, рендерер.

Цель работы: Реализация кроссплатформенной программной библиотеки для визуализации двухмерной и трехмерной графики, предоставляющую независимость программного обеспечения от аналогичных библиотек других компаний.

В процессе выполнения работы использовалась среда разработки Qt Creator и СУБД SQLite.

В результате работы была реализована программная библиотека для визуализации объектов под платформу рендеринга OpenGL.

В первой главе представлено описание предметной области, обзор и аналогов, описание выбранного средства разработки.

Во второй главе представлено проектирование и реализация модулей программной библиотеки. Реализация включает в себя создание объектов системы, определение их взаимодействия, программирование модулей.

Третья глава представляет собой выполненное задание по разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение», при выполнении которого были использованы анализа в области проектного и финансового менеджмента, в том числе менеджмента рисков.

Четвертая глава представляет собой выполненное задание по разделу «Социальная ответственность», где были рассмотрены аспекты производственной и экологической безопасности, безопасности в чрезвычайных ситуациях, а также правовые вопросы организации труда.

## Список терминов и сокращений

ЭВМ – электронно-вычислительная машина.

Рендерер (или графический визуализатор, в разговорной речи – «3D движок») – программа, отвечающая за визуализацию графических объектов в компьютерной графике.

БД – база данных.

СУБД – система управления базами данных.

ИС – информационная система.

SDF поле (signed distance field) – это изображение из оттенков серого, сгенерированное из контрастного черно-белого изображения, в котором уровень серого цвета означает дистанцию до ближайшей контрастной границы.

FPS (Frame per second) – частота кадров в секунду. Величина необходима для измерения скорости отрисовки графических объектов.

Шейдер (Шейдерная программа) – компьютерная программа, предназначенная для исполнения графическими процессорами.

Октодерево – тип древовидной структуры данных, в которой у каждого внутреннего узла восемь потомков[5].

Нода – элемент древовидной структуры данных, хранящий в себе информацию и способный проводить над нею манипуляции (поиск, добавление, удаление, выборку и т.д.).

ПО – программное обеспечение.

GPU (графический процессор) – отдельное устройство персонального компьютера или игровой приставки, выполняющее графический рендеринг.

IDEF (Integrated Computer-Aided Manufacturing) – методологии семейства ICAM для решения задач моделирования сложных систем, позволяет отображать и анализировать модели деятельности широкого спектра сложных систем в различных разрезах.

UML (Unified Modeling Language) – язык графического описания для объектного моделирования в области разработки программного обеспечения,

для моделирования бизнес-процессов, системного проектирования и отображения организационных структур.

AAA-проект – неформальный термин, обозначающий класс высокобюджетных компьютерных игр.

ЕРС (Event-Driven Process Chain) – событийная цепочка процессов, используется для описания процессов нижнего уровня.

## Оглавление

|  |    |
|--|----|
| Реферат .....  | 10 |
| Список терминов и сокращений .....   | 11 |
| Введение.....  | 16 |
| Глава 1. Исследование предметной области .....                                   | 19 |
| 1.1. Описание предметной области .....   | 19 |
| 1.1.1. Описание объекта автоматизации .....                                      | 19 |
| 1.1.2. Описание рассматриваемого бизнес-процесса .....                           | 20 |
| 1.2. Обзор существующих аналогов.....  | 25 |
| 1.2.1. «Unreal Engine».....  | 25 |
| 1.2.2. «Frostbite».....  | 25 |
| 1.2.3. «Unity3D».....  | 26 |
| 1.2.4. Результат обзора и анализа аналогов .....                                 | 26 |
| 1.3. Описание выбранного средства разработки.....                                | 29 |
| Глава 2. Проектирование программной библиотеки.....                              | 30 |
| 2.1. Требования к информационной системе .....                                   | 30 |
| 2.2. Роли пользователей в системе. UML-моделирование.....                        | 31 |
| 2.3. Проектирование потоков процессов .....                                      | 33 |
| 2.4. Реализация программной библиотеки .....                                     | 35 |
| 2.4.1. Построение примитивов.....  | 36 |
| 2.4.2. Наложение текстуры на объект .....  | 38 |
| 2.4.3. Создание анимации .....   | 39 |
| Глава 3. Финансовый менеджмент, ресурсоэффективность и<br>ресурсосбережение..... | 42 |

|  |    |
|--|----|
| 3.1. Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения..... | 42 |
| 3.1.1. Потенциальные потребители результатов исследования .....  | 42 |
| 3.1.2 Анализ конкурентных технических решений .....  | 42 |
| 3.1.3. SWOT-анализ.....  | 44 |
| 3.2. Планирование научно-исследовательских работ.....  | 45 |
| 3.2.1. Структура работ в рамках научного исследования .....  | 45 |
| 3.2.2. Определение трудоемкости выполнения работ и разработка графика проведения научного исследования.....                                    | 46 |
| 3.2.3. Бюджет научно-технического исследования .....   | 47 |
| 3.2.3.1. Расчет материальных затрат научно-технического исследования .....   | 47 |
| 3.2.3.2. Расчет амортизационных затрат .....   | 48 |
| 3.2.3.3. Расчет заработной платы.....  | 50 |
| 3.2.3.4. Расчет затрат на электроэнергию .....   | 53 |
| 3.2.3.5. Формирование бюджета затрат научно-исследовательского проекта ...   | 54 |
| 3.3. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования.....             | 55 |
| 3.4 Вывод.....   | 57 |
| Глава 4. Социальная ответственность.....   | 59 |
| Введение.....  | 59 |
| 4.1. Правовые и организационные вопросы обеспечения безопасности .....   | 60 |
| 4.2. Производственная безопасность.....  | 62 |
| 4.2.1. Повышенный уровень электромагнитных излучений.....  | 63 |
| 4.2.2. Отклонение показателей микроклимата.....  | 64 |

|  |    |
|--|----|
| 4.2.3. Недостаточная освещенность рабочей зоны .....                               | 65 |
| 4.2.4. Повышенный уровень шума на рабочем месте .....                              | 68 |
| 4.2.5. Эмоциональные перегрузки .....  | 69 |
| 4.2.6. Нарушение правил электробезопасности .....                                  | 71 |
| 4.3. Экологическая безопасность.....   | 73 |
| 4.4. Безопасность в чрезвычайных ситуациях.....                                    | 74 |
| 4.5 Выводы .....   | 76 |
| Заключение .....   | 77 |
| Список достижений.....   | 78 |
| Список источников .....  | 79 |
| Приложение А. Диаграмма классов. ....  | 81 |
| Приложение Б. Таблицы и диаграммы для раздела «Финансовый менеджмент».....         | 82 |
| Приложение В. Программный код заголовочных файлов, используемые в библиотеке. .... | 84 |

## Введение

В наши дни средства визуализации компьютерной графики используются во многих областях для упрощения представления и понимания объектов, представления виртуальных объектов в реалистическом виде, визуализации анимации, воспроизведения реальных объектов на экране и др. Выделяют следующие области применения компьютерной графики [1]:

- Научная графика, необходимая для визуализации графиков и диаграмм, физических, математических, биохимических и других научных процессов и явлений;
- Деловая графика, которая позволяет наглядно представить различные показатели работы учреждений: отчетности и документация, бизнес-процессы, структуры и компоненты учреждений и других бизнес-объектов;
- Конструкторская графика, которая используется в работе архитекторов и инженеров для визуализации двумерных и трехмерных моделей проектируемых объектов и систем;
- Художественная и рекламная графика, используемая в компьютерных играх и мультимедии для визуализации виртуальных объектов, позволяющей пользователю представить данные объекты реалистичными;
- Компьютерная анимация, в которой расчет состояний объектов и их визуализацию производит компьютер. Таким образом при быстрой смене кадров у пользователя возникает ощущение движения объекта.

На практике функциональность перечисленных областей компьютерной графики используется в совокупности.

История компьютерной графики началась еще с 40-х годов XX-века, когда на электронных лампах ЭВМ первого поколения программисты выводили простейшие узоры и изображения. Но полученные изображения были далекими от реальности.



Создание более реалистичного и сложного изображения стало возможным только с появлением графических процессоров и в дальнейшем – шейдеров.

Изначально графические процессоры не предоставляли возможности управлять командами рендеринга и вычислениями, а только принимали на вход данные и отдавали центральному процессору выходные данные после расчетов. После изменения структуры GPU и появления шейдеров разработчики получили возможность самостоятельно контролировать расчет внутри видеокарты.

Шейдерные программы[4] делятся на три типа:

- Вершинный шейдер – шейдерная программа, которая предназначена для расчета специфической геометрической информации, которую потом будут использовать пиксельные шейдеры. Зачастую это координаты точки, направляющего вектора и информация о цвете, зависящая от положения точки в пространстве или от направляющего вектора (к примеру реализация тени объекта на обратной к источнику света стороне).

- Геометрический шейдер – шейдерная программа, которая позволяет генерировать дополнительные точки и примитивы, одновременно переопределяя информацию о текстурах и цвете. Данный шейдер существенно облегчает процесс генерации сложных объектов, так как в оперативной памяти компьютера хранятся только те точки объекта, которые передаются на вершинный шейдер.

- Фрагментный шейдер – шейдерная программа, которая обычно задает модель расчета освещения отдельно взятой точки изображения, производят выборку из текстур и математические операции над цветом и значением глубины.

Информация о визуализируемом объекте передается шейдерам именно в следующей последовательности: вершинному, геометрическому (при наличии), фрагментному.

Дальнейшая эволюция графических процессоров позволила разработчикам совершенствовать графические визуализаторы, улучшая качество изображений и увеличивая сложность объектов. Реализация собственного визуализатора организацией, направленной на создание компьютерных игр и программного обеспечения, которое должно содержать качественную графику, позволяет подстраивать функционал рендерера под разрабатываемые программные продукты.

Целью работы является реализация кроссплатформенной программной библиотеки для определения, построения и визуализации двумерных и трехмерных объектов, предоставляющей независимость программного обеспечения от аналогичных библиотек других компаний.

Для достижения поставленной цели выдвинуты следующие задачи:

1. Выполнить описание предметной области, включая объект автоматизации и основной рассматриваемый бизнес-процесс (диаграммы в нотации IDEF0).
2. Выполнить обзор существующих аналогов и провести анализ результатов.
3. Выделить роли пользователей и описать варианты использования разрабатываемой библиотеки (диаграмма в нотации UML).
4. Создать необходимые объекты программной библиотеки, описать их взаимосвязи (диаграмма классов в нотации UML).
5. Реализовать спроектированную программную библиотеку.

Дополнительно в данной работе были выполнены задания по разделам «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение» и «Социальная ответственность», позволяющие оценить необходимость проведения данной работы и реальность внедрения результатов работы в реальную деятельность организаций.

# Глава 1. Исследование предметной области

## 1.1. Описание предметной области

### 1.1.1. Описание объекта автоматизации

Объектом автоматизации является компания ООО «Интех». Данная компания оказывает следующие услуги:

- Разработка и тестирование программного обеспечения;
- Внедрение и сопровождение разработанного ПО.

Главным руководителем компании является директор. Штат сотрудников состоит из 11 человек, среди которых присутствуют дизайнеры, веб-разработчики, разработчик мобильных приложений и разработчики настольных приложений, бухгалтер. Взаимодействие сотрудников с руководством и заказчиками обеспечивает управляющий командой (team lead), занимающий пост заместителя директора компании. Организационная структура представлена на рисунке 1.1.



Рисунок 1.1 – Организационная структура компании

Компания существует около 5 лет. На данный момент ею был реализован один коммерческий проект, который находится в сопровождении.

Для реализации будущих конкурентоспособных проектов руководство компании приняло решение о реализации собственной программной библиотеки для рендеринга объектов.

### **1.1.2. Описание рассматриваемого бизнес-процесса**

В данной работе главным рассматриваемым бизнес-процессом компании является разработка программного обеспечения с использованием разрабатываемой программной библиотекой для визуализации объектов.

В наше время существует множество графических визуализаторов для реализации различных графических возможностей программного обеспечения. На вход визуализатору для рендеринга объекта разработчик передает общую информацию об объекте (цвет примитивов, расположение примитива в пространстве, тип примитива и т.д.), функции рендеринга (прозрачность объекта, отсечение невидимых линий примитива, способ соединения точек, а именно отрисовка точек, построение линий или закрашивание треугольников и т.д.), текстурные координаты (при использовании текстуры для ее дальнейшего наложения на объект). Внутренние преобразования и генерация информации происходит согласно реализованной авторами визуализатора структуре и правилами обращения к данному ПО, которые прописаны в документации продукта. После преобразования и генерации на выходе разработчик получает графическое представление объекта (изображение на экране монитора).

Входные, выходные данные и механизмы процесса рендеринга объекта проиллюстрированы на рисунке 1.2.

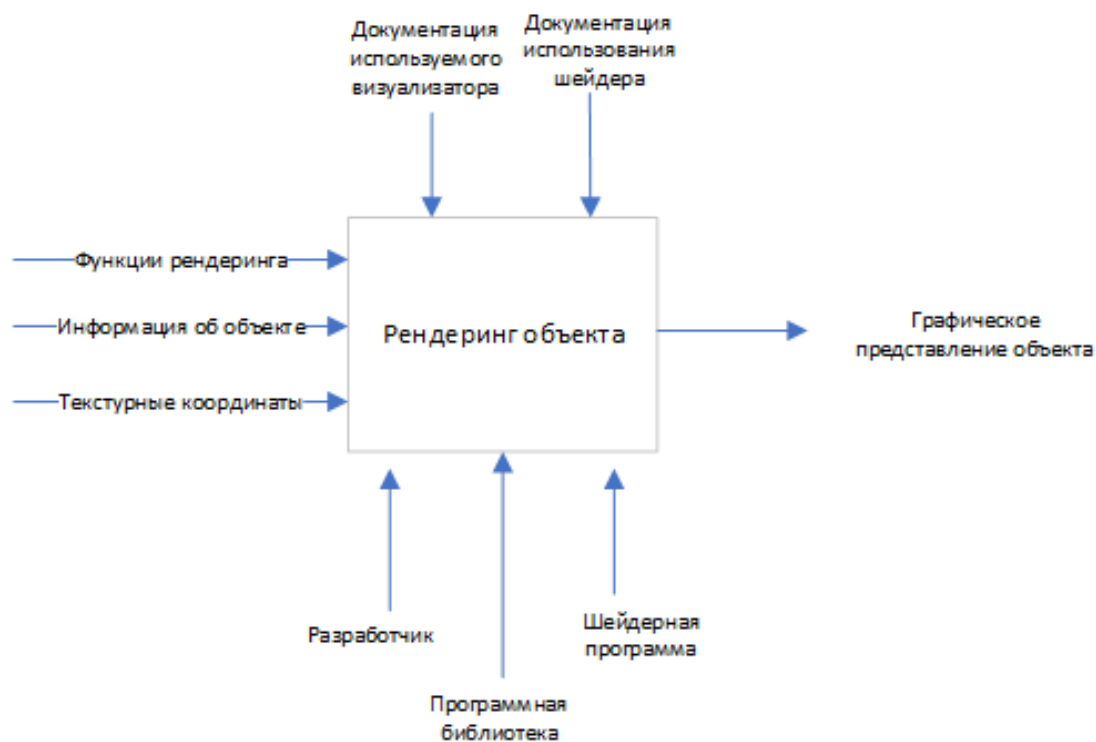


Рисунок 1.2 – Диаграмма первого уровня в нотации IDEF0 («черный ящик») для бизнес-процесса «Рендеринг объекта»

На рисунке 1.3 представлена диаграмма второго уровня декомпозирующая диаграмму первого уровня для процесса «Рендеринг объекта»

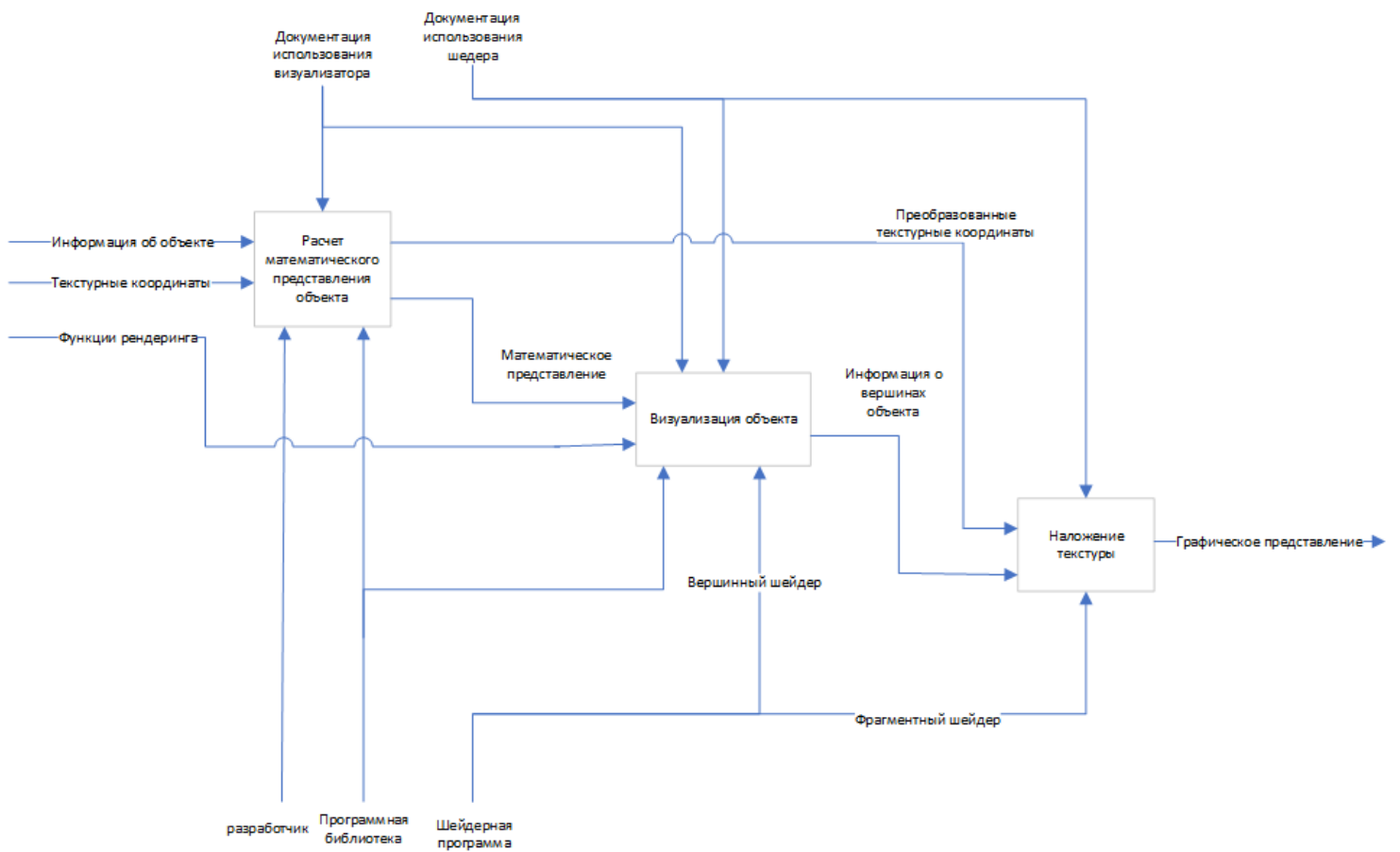


Рисунок 1.3 – Диаграмма второго уровня в нотации IDEF0 для бизнес-процесса «Рендеринг объекта»

Декомпозиция главного бизнес-процесса показывает, что процесс рендеринга разбивается на три этапа: расчет математического представления объекта, визуализация объекта и наложение текстуры

На рисунках 1.4-1.5 представлены диаграммы третьего уровня, декомпозирующие каждый из представленных этапов.

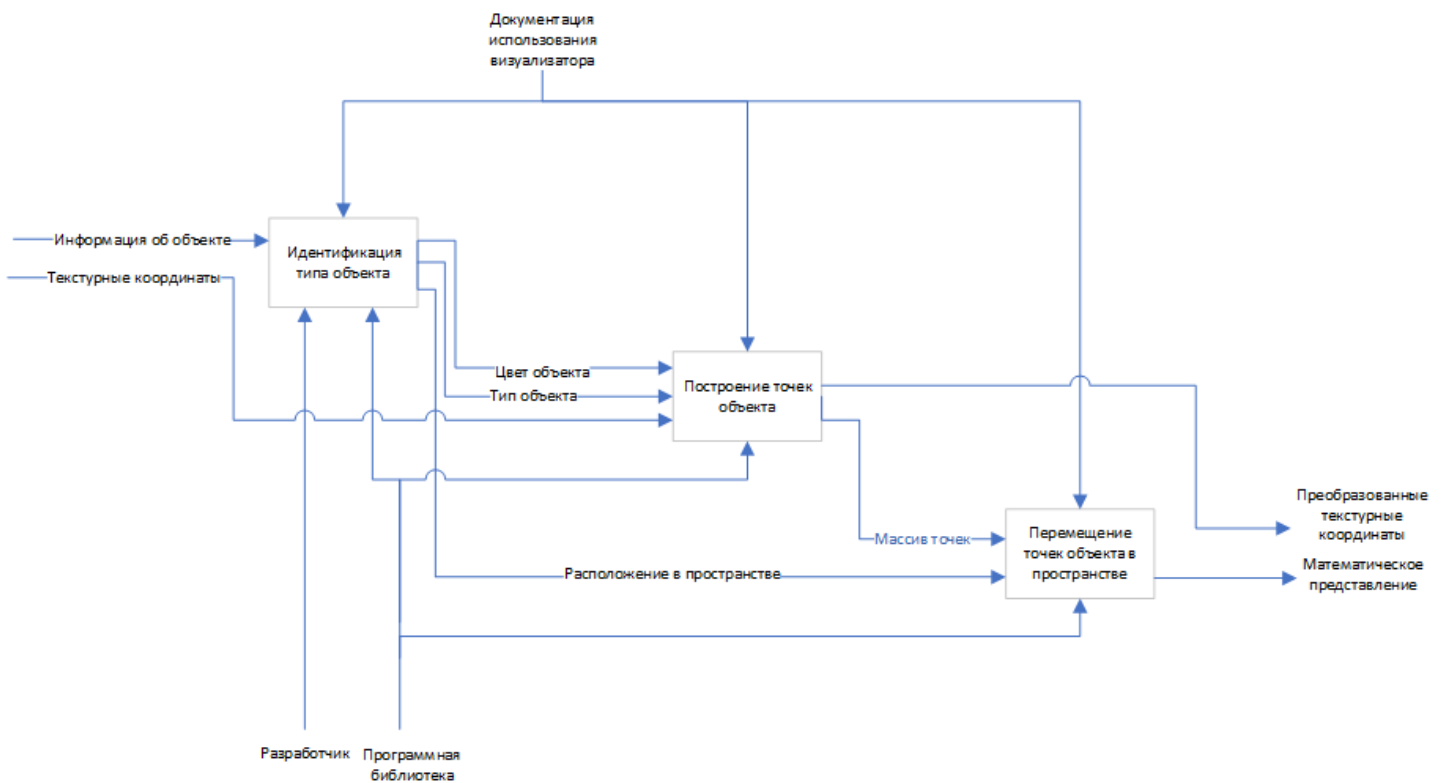


Рисунок 1.4 – Диаграмма третьего уровня в нотации IDEF0 для этапа «расчет математического представления» исследуемого бизнес-процесса

Расчет математического представления объекта начинается с указанием разработчиком типа строящегося объекта, после чего программная библиотека идентифицирует его тип. Далее рендерер строит точки относительно центра виртуального пространства в соответствии с заданным алгоритмом построения того или иного типа объекта. Текстурные координаты, если были заданы, изменяются и дополняются новыми в соответствии с типом объекта и количеством точек в объекте. Последним этапом является перемещение объекта в пространстве с помощью специальных матриц, отвечающих за поворот объекта в пространстве и его линейное перемещение.

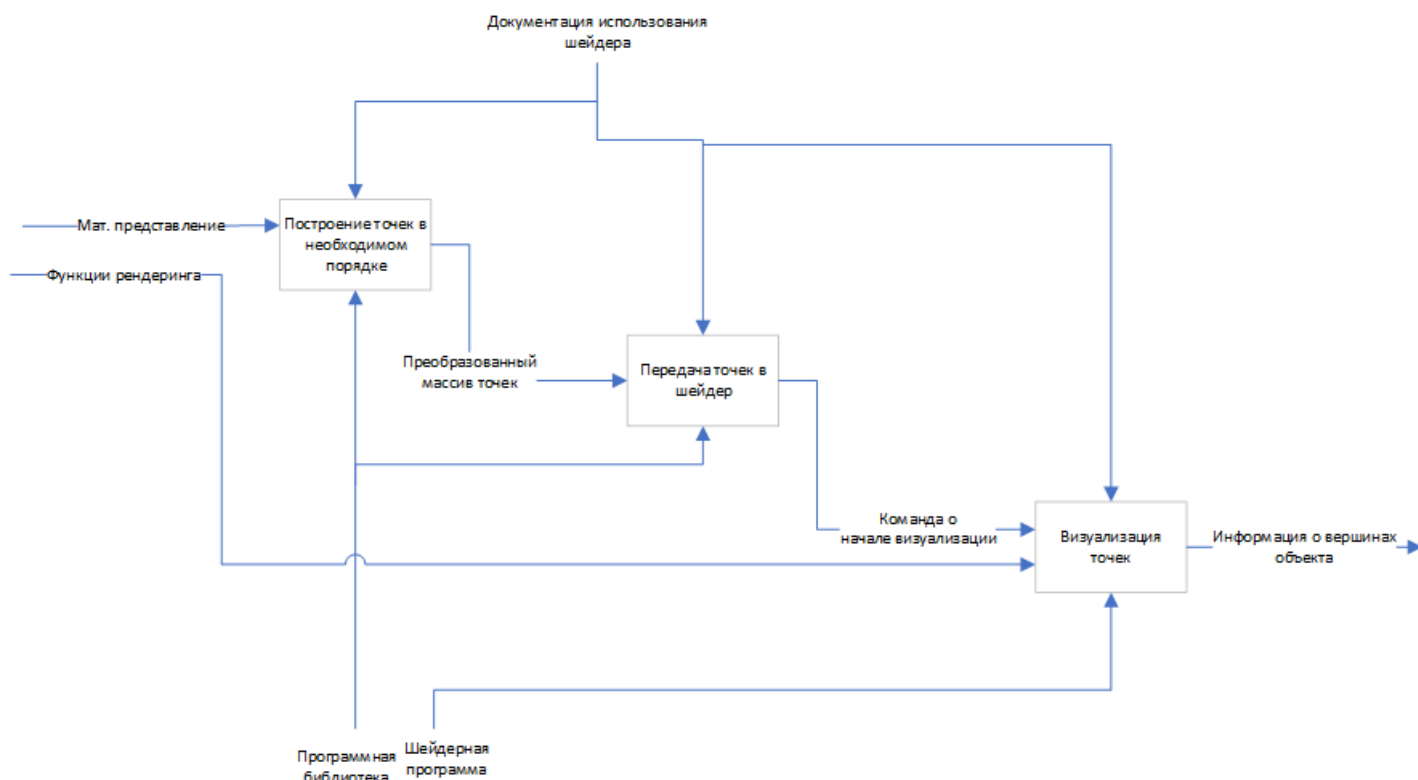


Рисунок 1.5 – Диаграмма третьего уровня в нотации IDEF0 для этапа «Визуализация объекта» исследуемого бизнес-процесса

Визуализация объекта начинается после расчета математического представления. Массив точек, рассчитанный на предыдущем этапе формируется в необходимой для визуализации видеокарты последовательности. Далее преобразованный массив точек передается в шейдерную программу, где происходят внутренние расчеты видеокарты и визуализация точек.

Последний этап рендеринга объекта – наложение текстур. При наличии текстуры и поддержке ее наложения шейдером каждая точка передает видеокarte свою текстурную координату. Данный этап является атомарным и не раскладывается на подуровень.



## **1.2. Обзор существующих аналогов**

Сегодня существует целый ряд визуализаторов и программных библиотек для рендеринга объектов. Но при этом бесплатные аналоговые продукты конкурентных компаний не предоставляют нужного функционала для реализации конкурентоспособных коммерческих проектов.

### **1.2.1. «Unreal Engine»**

Одним из лидирующих графических визуализаторов является «Unreal Engine».

Unreal Engine 4 (UE) — это набор инструментов для разработки игр, имеющий широкие возможности: от создания двухмерных игр на мобильные до AAA-проектов для консолей. Данный визуализатор имеет огромные возможности для использования и создания графики в приложениях. Но для этого необходимы хорошие знания языка программирования C++ и внутренних особенностей рендерера. Для пользователей, не знающих языков программирования, разработчики данного визуализатора реализовали скриптовый язык Blue Print, который существенно ускоряет и упрощает разработку, но при этом не предоставляет полной функциональности визуализатора.

Компания-создатель визуализатора UE «Epic» имеет большие доходы от созданных ПО и средств разработки, что является причиной бесплатного использования данного визуализатора. Лишь при превышении дохода в 3 000\$ за квартал компанией, использующей данный визуализатор, ей требуется платить 5% прибыли от продаж игры.

### **1.2.2. «Frostbite»**

Следующая рассматриваемая в качестве аналога рендереров — «Frostbite». Данное ПО состоит из визуализатора, собственного модуля обработки звука и модуля взаимодействия объектов. Одной из заявленных особенностей по словам разработчиков является оптимизация для работы на многоядерных процессорах. Данный рендерер написан на языке

программирования C#. Сегодня реализован целый ряд популярных видеоигр с использованием данного ПО.

Не смотря на перечисленные сильные стороны данного продукта, главным недостатком визуализатора является закрытое лицензирование, то есть данный программный продукт используется внутри компании EA Digital Illusions SE для собственных разработок и проектов других филиалов EA.

### **1.2.3. «Unity3D»**

Визуализатор «Unity3D» – еще один рассматриваемая аналоговый рендерер. В отличие от предыдущих данный визуализатор был разработан не для реализации игр, а для понижения порога вхождения для новичков в разработке игр и упрощения портирования готовых игр на различные платформы. Таким образом для его использования не обязательно знать программирование, что существенно увеличивает скорость написания программы с использованием Unity3D, исключая время на изучение программирования.

Цена за подписку данного ПО составляет 125\$ в месяц, что делает данный рендерер менее доступным для аудитории.

### **1.2.4. Результат обзора и анализа аналогов**

По результатам обзора и анализа аналогов разрабатываемых модулей были выявлены как достоинства программных средств, так и их недостатки.

Подведем итог по каждому из аналогов разрабатываемого ПО:

- **Unreal Engine.** Данный визуализатор является мощным средством разработки ПО, но для его использования необходимо иметь глубокие знания особенностей внутренней структуры и работы данного рендерера. Для этого необходимо иметь стаж в его использовании более чем 1-2 года. Вторым недостатком данного ПО является частичное удовлетворение функциональных потребностей компании. Таким образом отсутствует возможность дополнить функциональность, при необходимости.

- Frostbite. Мощный инструмент разработки, включающий в себя не только визуализатор, но и другие необходимые для разработки компоненты. При этом существенный недостаток «Отсутствие лицензирования» исключает возможность его использования.

- Unity3D. Данное ПО имеет ряд сильных сторон: простота в использовании, стабильность работы визуализатора, быстрое изучение функциональных возможностей инструмента разработки. Но данное ПО не предоставляет полного необходимого функционала. При этом его функционал реализован не максимально эффективно и стоимость использования данного рендера требует достаточных финансовых вложений.

Более подробное сравнение конкурентоспособности аналогов проведено во главе «Финансовый менеджмент» (таблица 3.1).

Использование готового графического визуализатора упрощает и ускоряет процесс разработки. Но при этом существует ряд недостатков использования внешнего продукта:

- Зависимость от функциональности внешнего продукта. Разработчики, использующие готовый рендерер имеют возможность реализовывать только тот функционал, который предусмотрен готовым визуализатором. Таким образом возникает зависимость разрабатываемого программного обеспечения от внешней компании.

- Низкая производительность визуализации объектов. При реализации неоптимальных алгоритмов или выбора менее производительного языка программирования готовый внешний продукт уменьшает производительность разрабатываемого программного обеспечения.

Более подробное объяснение причин низкой производительности визуализатора проиллюстрировано на рисунке 1.6.

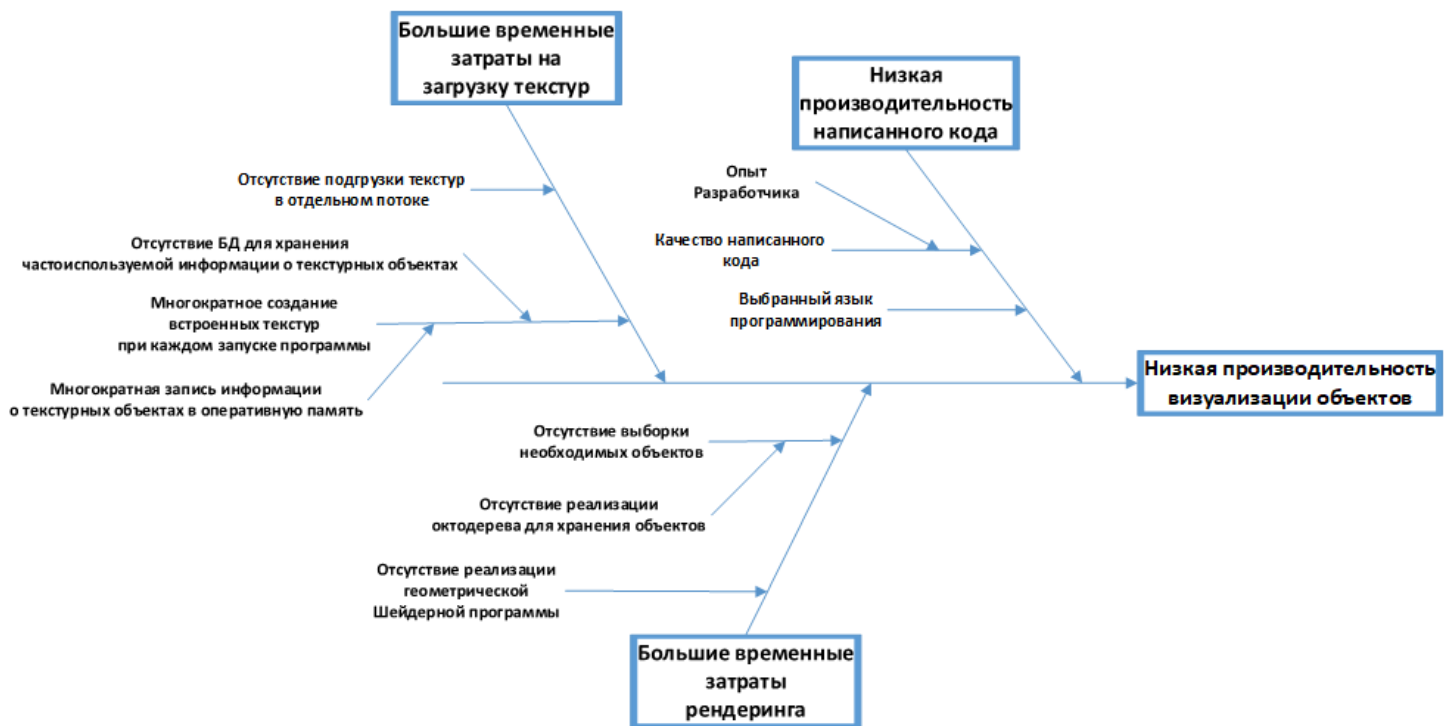


Рисунок 1.6 – Диаграмма причинно-следственных связей в нотации «FishBone»

Перечисленные недостатки являются главной причиной для реализации собственного графического визуализатора компанией, деятельность которой направлена на создание компьютерных игр, редакторов и других программных обеспечений, которым необходима качественная графика, наличие которой не влияет на производительность всего продукта.

Таким образом можно сделать вывод, что при наличии компанией перспективных проектов, требующих использования и создания качественной графики, необходима реализация собственного инструмента разработки, включающего в себя графический визуализатор. Для реализации максимально эффективных алгоритмов в компании должны присутствовать программисты, имеющие опыт в работе с графическими визуализаторами и различными языками программирования. Реализация собственного визуализатора предоставляет возможность его будущей модернизации и оптимизации, удовлетворить все функциональные потребности в будущей реализации ПО, уменьшить финансовые затраты на реализацию будущего ПО.

### 1.3. Описание выбранного средства разработки

Для реализации разрабатываемой библиотеки был выбран язык программирования C++. Данный язык программирования имеет свои недостатки и преимущества.

Недостатки:

- Сложность в изучении, что существенно уменьшает скорость разработки ПО при отсутствии знаний языка программирования;
- Прямая работа с памятью может обратить данное преимущество в риск наличия сложноуловимой ошибки.

Преимущества:

- Ресурсоэффективность разрабатываемого ПО;
- Производительность за счет возможности работы с памятью на низком уровне;
- Кроссплатформенность.

Средой разработки был выбран Qt Creator. Данная среда имеет низкий порог вхождения (легкость в использовании), удобное межпроцессорное взаимодействие (слот-сигнальную систему), активно сопровождается разработчиками, кроссплатформенность и возможность быстрого написания кода для маленьких и средних проектов, за счет наличия объемной и структурированной документации.

Для написания шейдерных программ под платформу рендеринга OpenGL используется язык программирования GLSL.

Для работы с БД был выбран SQLite. Данная СУБД является встроенной, мощной, легко интегрируемой. SQLite хорошо подходит для хранения небольших объемов данных, к которым обращается небольшое количество пользователей. В разрабатываемом проекте БД используется для хранения часто используемой информации о текстурах. Таким образом данное СУБД хорошо подходит для разрабатываемой программной библиотеки.

## Глава 2. Проектирование программной библиотеки

### 2.1. Требования к информационной системе

Для реализации программной библиотеки были выявлены требования к системе.

Нефункциональные требования:

1. Разрабатываемый продукт необходимо разработать в структуре динамической программной библиотеки. ПО должно быть легко в использовании и интеграции.
2. Необходимо реализовать кроссплатформенность библиотеки.
3. Необходимо реализовать поддержку различных систем рендеринга: Direct3D, OpenGL, Vulkan.
4. Разрабатываемая библиотека должна обладать высокой производительностью.

Функциональные требования:

1. Разработать математическую часть библиотеки, отвечающую за математическое представление всех объектов:
  - a. Реализация двумерных примитивов.
  - b. Реализация трехмерных примитивов.
  - c. Расчёт текстурных координат для фигур.
  - d. Реализация окто дерева, отвечающее за хранение всех объектов, а так же для выборки нужных объектов для рендеринга.
  - e. Реализация анимаций над объектами.
  - f. Реализация подмодуля, отвечающего за работу со шрифтами:
    - i. Подготовка информации о шрифтах.
    - ii. Создание текстурной сетки символов.
    - iii. Реализация SDF поля для дальнейшего использования подложки символа[3].
    - iv. Визуализация текста.

- v. Реализация переноса слов, выравнивания по левому, правому краям, по ширине.
  - g. Реализация камеры наблюдателя.
  - h. Реализовать подгрузку текстур в отдельном потоке.
2. Разработать графическую часть библиотеки, отвечающую за рендеринг:
- a. Возможность указания функций для рендеринга;
  - b. Реализация шейдерных программ для использования по умолчанию (для фигур, текста);
  - c. Реализация возможности подсчета количества кадров в секунду;
  - d. Возможность использования разработчиком своих шейдерных программ;
  - e. Возможность использования разработчиком своих и встроенных текстур.

## **2.2. Роли пользователей в системе. UML-моделирование**

Помимо требований для реализации ИС были выделены роли ее участников (рисунок 2.1-2.2).

В информационной системе участвуют 2 основных актора:

1. Разработчик. Данный актер является пользователем данной системы, который разрабатывает собственный продукт с использованием библиотеки для рендеринга.
2. Графический визуализатор. Визуализатор делится на 2 подмодуля:
  - a. Графический модуль. Данный модуль отвечает за визуализацию объектов.
  - b. Математический модуль. Данный модуль отвечает за определение и построение и хранение объектов.

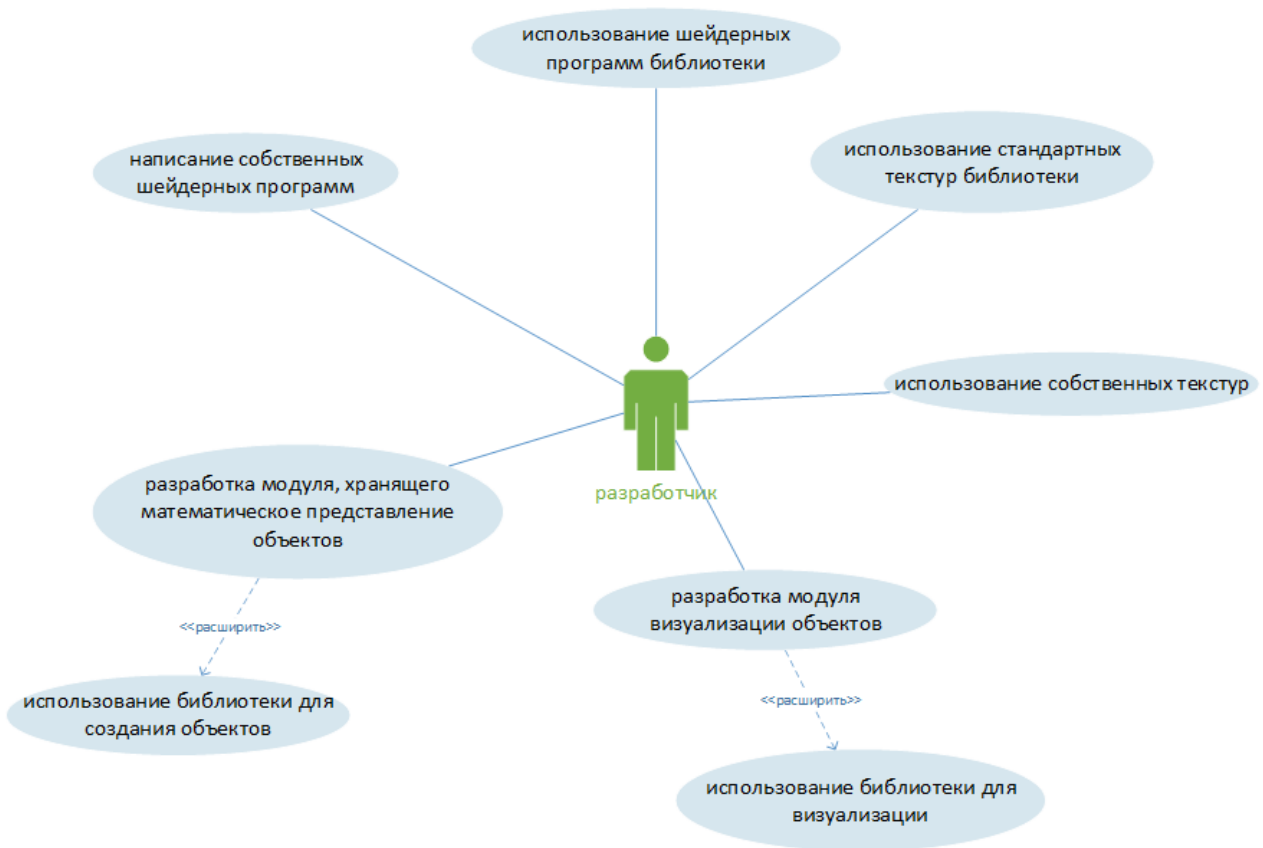


Рисунок 2.1 – Диаграмма вариантов использования информационной системы для пользователя библиотеки (разработчика)

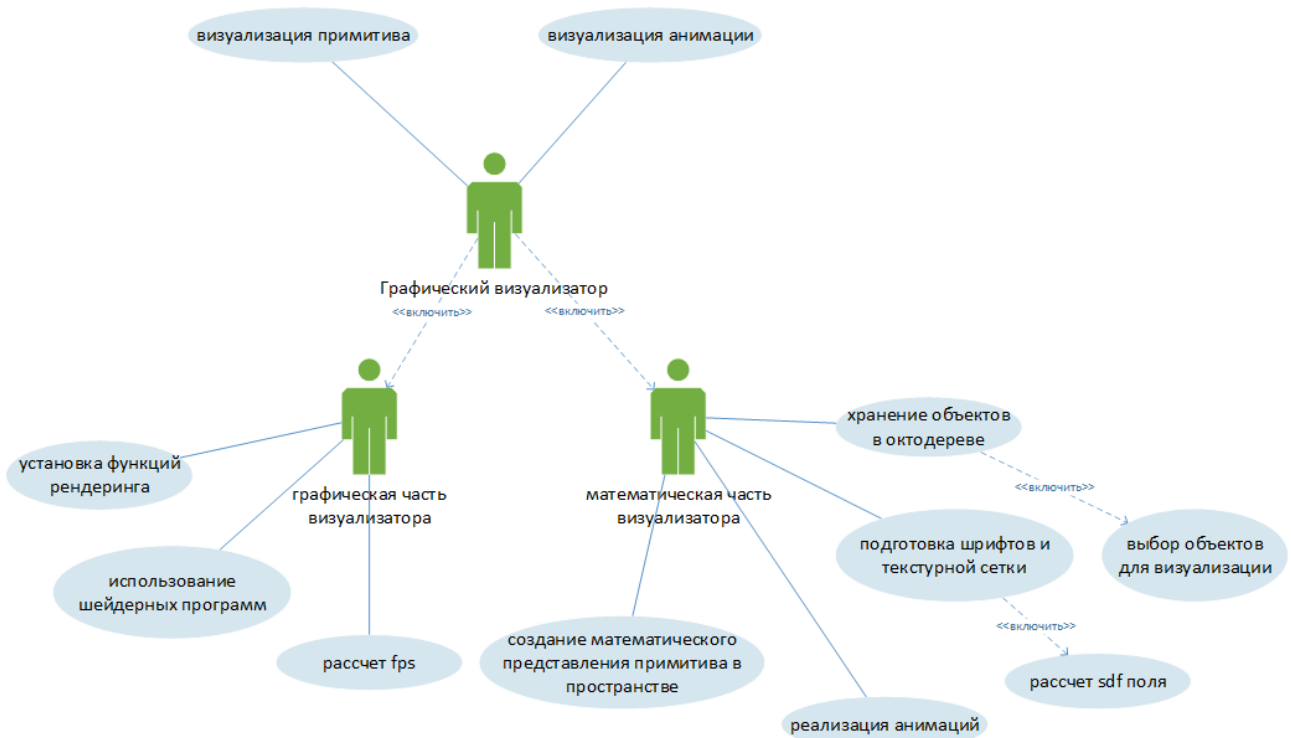


Рисунок 2.2 – Диаграмма вариантов использования информационной системы для графического визуализатора



Прежде, чем приступить к реализации продукта, необходимо досконально представлять всем участникам проекта все бизнес-процессы разработки, функциональность каждого модуля информационной системы и взаимосвязи между участниками бизнес-процессов. Для этого были спроектированы диаграммы, описывающие поведение информационной системы.

### **2.3. Проектирование потоков процессов**

Для описания процессов ИС была проектирована диаграмма событийной цепочки процессов в нотации EPC (рисунок 2.3).

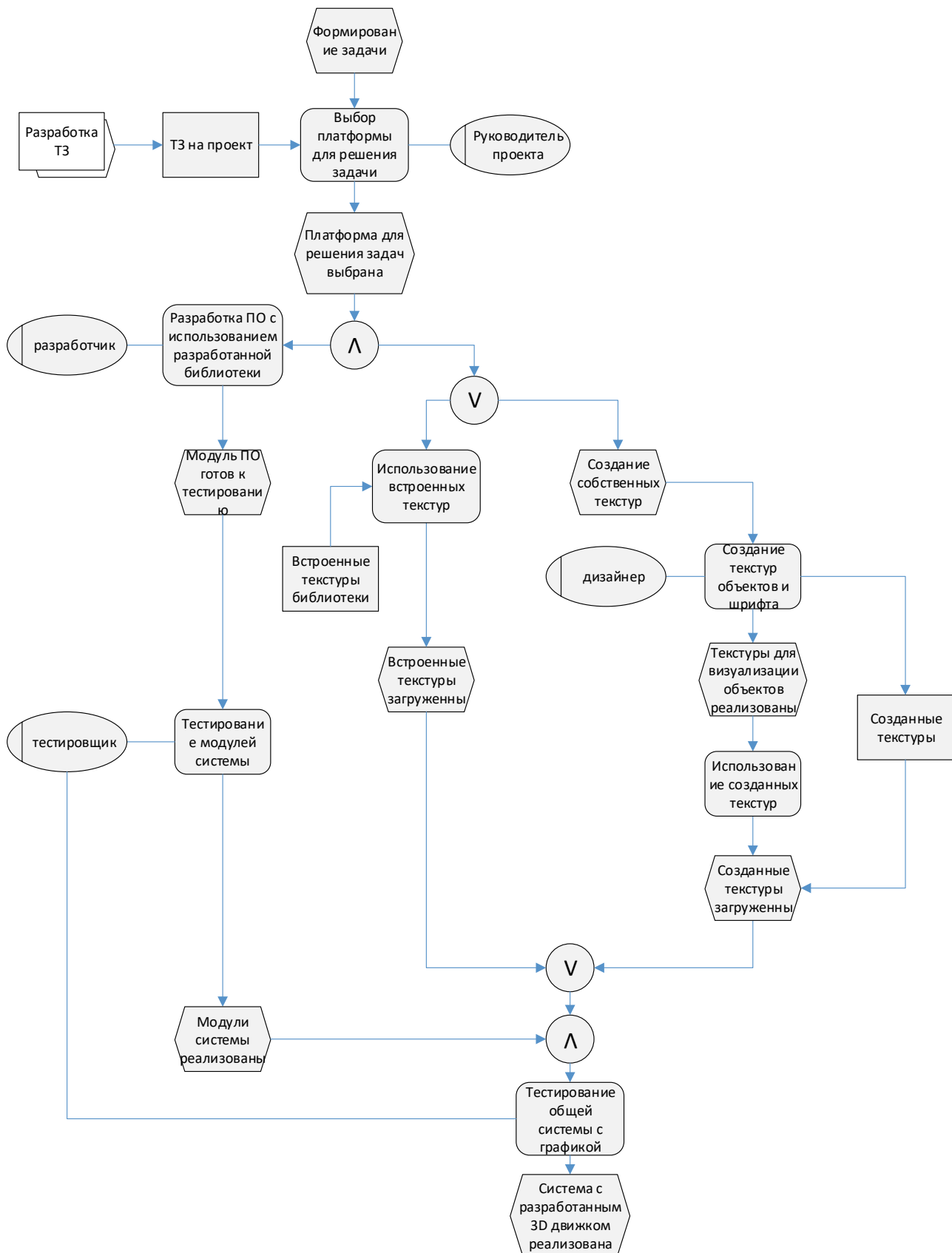


Рисунок 2.3 – Диаграмма событийной цепочки процессов в нотации «EPC»

С помощью диаграммы можно выделить 4 основных этапа разработки, которые выполняются разными участниками проекта:

управление проектом (руководитель проекта), разработка программной части (разработчик), создание текстур объектов (дизайнер) и тестирование ПО (тестировщик).

## 2.4. Реализация программной библиотеки

Разрабатываемое приложение на данном этапе имеет следующий функционал:

- Построение примитивов:
  - Квадрат;
  - Круг;
  - Куб;
  - Сфера;
  - Цилиндр;
  - Тор;
- Наложение текстур на объекты;
- Визуализация текста;
- Расчет SDF поля для визуализации позиции символа;
- Расчет FPS;
- Создание анимации:
  - Линейное перемещение;
  - Поворот объекта вокруг точки, объекта, своей оси;
  - Криволинейное перемещение;
  - Распад сложного объекта;
  - «Склеивание» сложного объекта.

Диаграмма классов разработанного ПО прикреплено в приложении А. Фрагменты заголовочных файлов написанного программного кода представлены в приложении В.

### 2.4.1. Построение примитивов

Как известно, графический процессор способен визуализировать следующие примитивы: точка, линия, треугольник. Таким образом все примитивы, которые способна строить разрабатываемая библиотека состоят из треугольников.

Разработчик способен выбрать один из способов визуализации: треугольниками, линиями и точками. В зависимости от способа массив точек, передаваемый в шейдер имеет разную структуру.

На рисунке 2.4 представлен пример построения квадрата, строящийся из линий, на рисунке 2.5 – из треугольников.

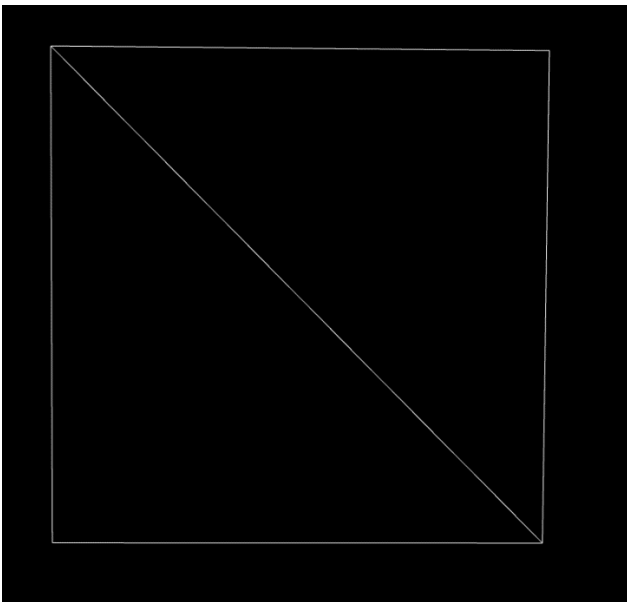


Рисунок 2.4 – Визуализация квадрата  
с помощью линий

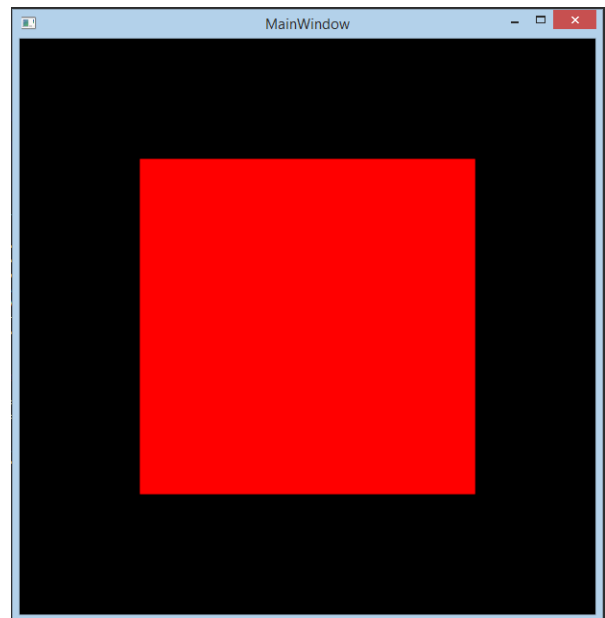


Рисунок 2.5 – Визуализация квадрата  
с помощью треугольников

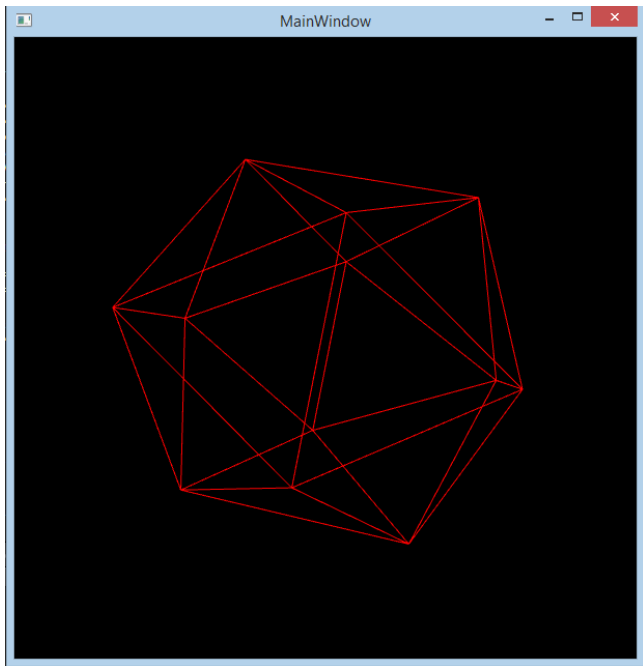


Рисунок 2.6 – Визуализация икосаэдра

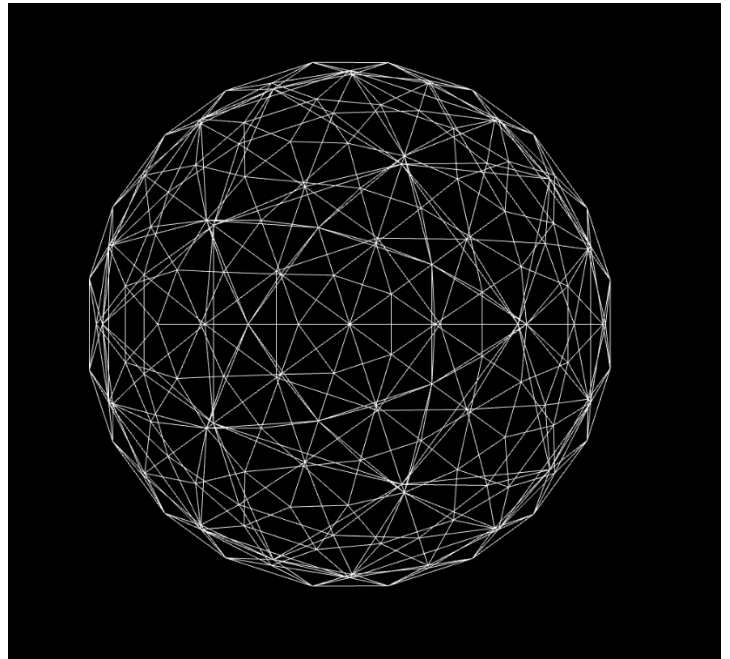


Рисунок 2.7 – Визуализация сферы

На рисунке 2.6 продемонстрирована визуализация икосаэдра. Для получения сферы используется алгоритм триангуляции, который «вытягивает» треугольники, из которых состоит каждая грань икосаэдра. На рисунке 2.7 представлена полученная сфера после двухуровневой триангуляции икосаэдра.

Помимо простой визуализации объектов для получения более реалистичного изображения используются различные шейдеры, к примеру шейдер с использованием освещения по Фонгу, что проиллюстрировано на рисунках 2.8-2.9.

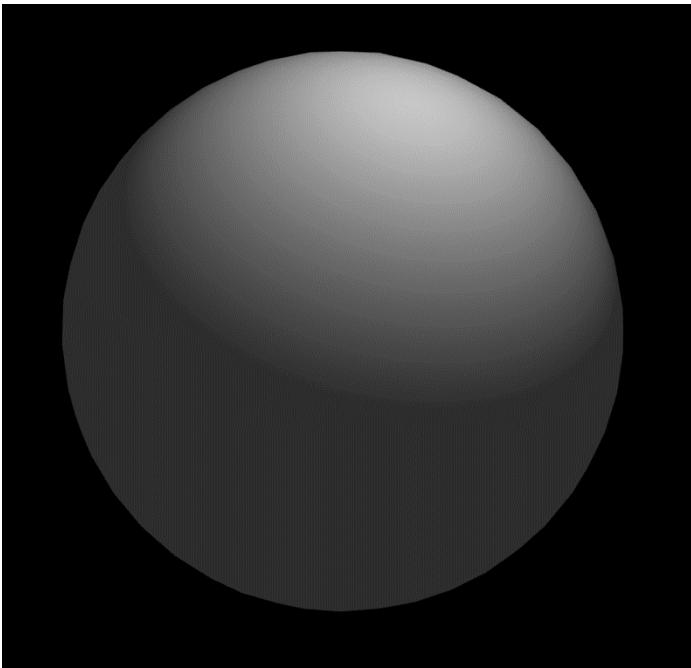


Рисунок 2.8 – Визуализация сферы

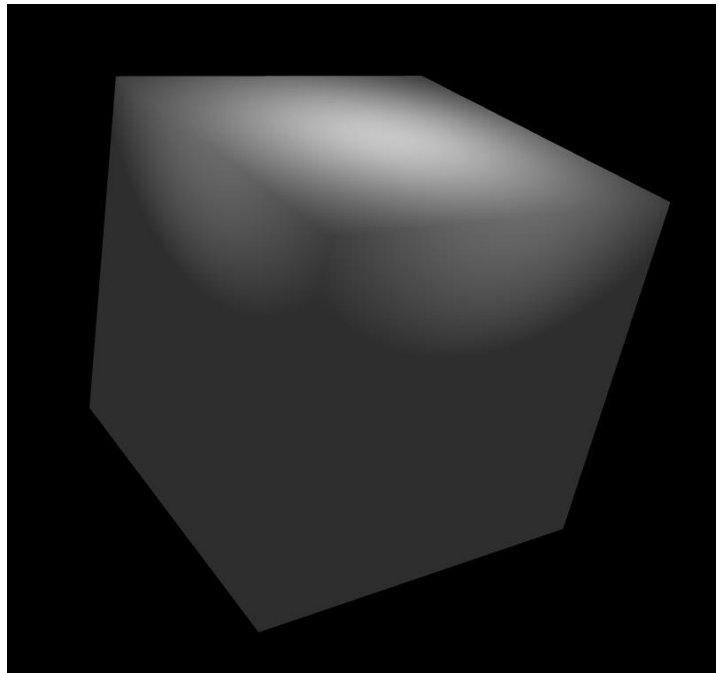


Рисунок 2.9 – Визуализация куба

Помимо построения примитивов система строит некоторые более сложных объектов, которые необходимы для разработки графического интерфейса будущих ПО. Например цилиндр и тор (рисунки 2.10-2.11) требуют для построения более сложных алгоритмов.

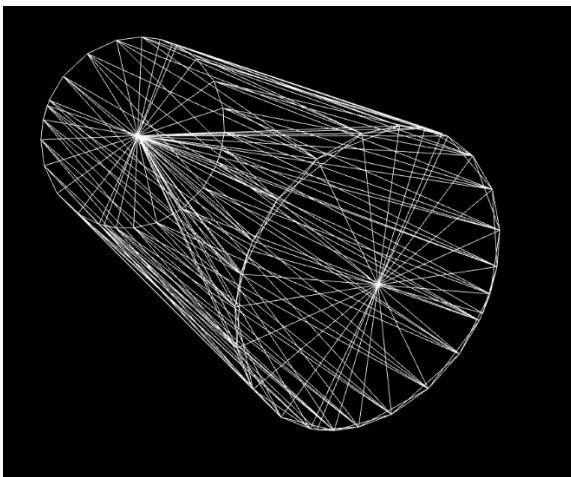


Рисунок 2.10 – Визуализация цилиндра

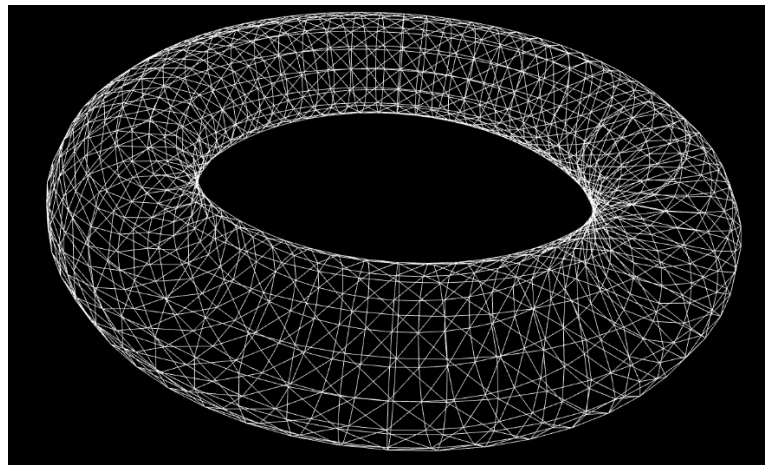


Рисунок 2.11 – Визуализация тора

#### **2.4.2. Наложение текстуры на объект**

После построения объектов программная библиотека имеет возможность накладывать на них текстурное изображение. Для этого

разработчик включает данный функционал, указывает расположение изображения, который накладывается на объекты.

На рисунках 2.12-2.13 проиллюстрированы примитивы с наложением на них текстурного изображения.

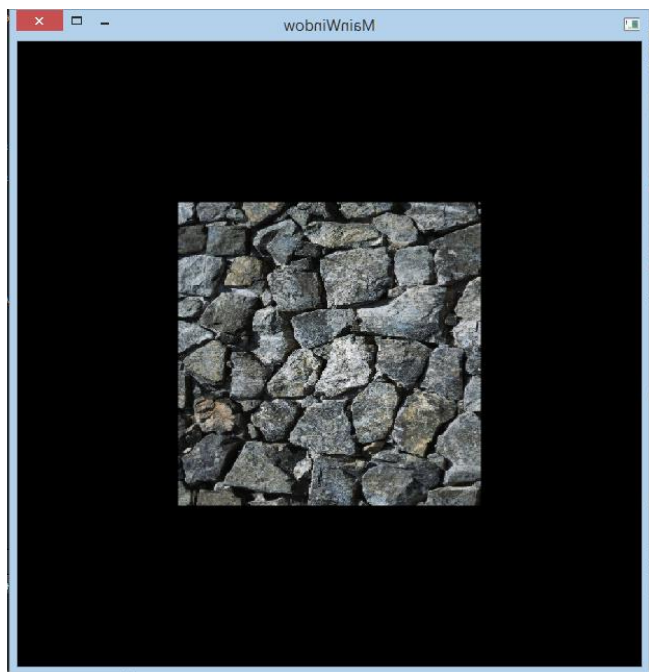


Рисунок 2.12 – Визуализация квадрата с использованием текстуры

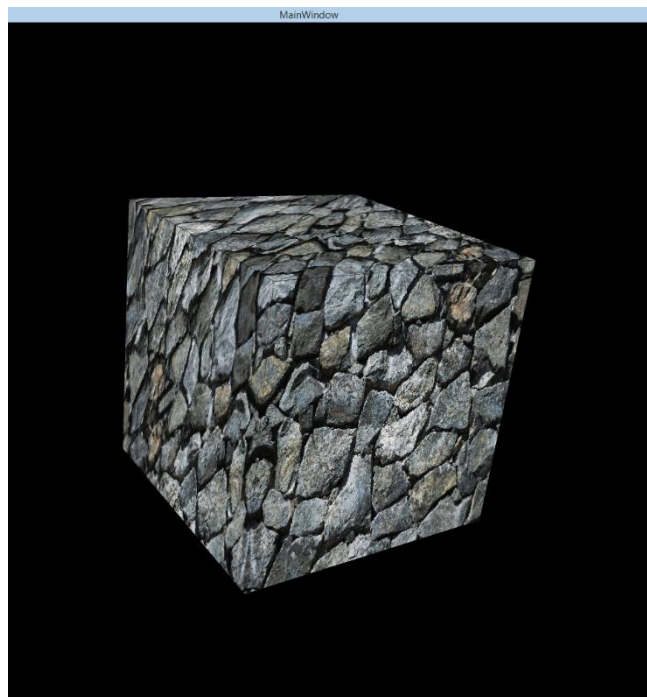


Рисунок 2.13 – визуализация куба с использованием текстуры

### 2.4.3. Создание анимации

Помимо построения и визуализации объектов в разрабатываемой библиотеке реализована возможность работы с анимацией: передвижение объектов, их поворот, приближение и отдаление камеры наблюдающего и др.

Для реализации данного функционала было разработано октодереву, позволяющее визуализировать не все объекты, существующие в пространстве, а только те, которые попадают в обзор пользовательской камеры.

Октодереву имеет такую структуру, что при попадании в ноду октодереву объектов, количество которых больше, чем максимально допустимое, нода разбивается на 8 подпространств. При пересечении грани

подпространства с объектом, данный объект помещается в отдельных массив элементов. Пример генерации объектов в пространстве и использования октодерева приведен на рисунках 2.14-2.15.



Рисунок 2.14 – генерация фигур в пространстве

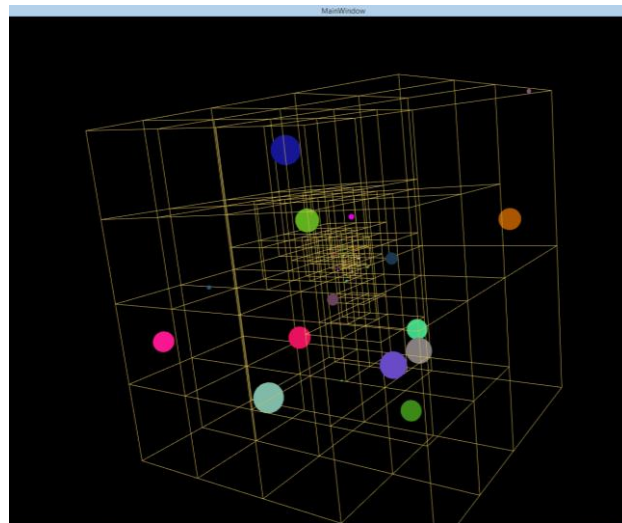


Рисунок 2.15 – использование и визуализация октодерева

В настоящее время данная библиотека ориентирована на удовлетворение функционалом будущей системы управления проектами. Для этого заранее был реализован некоторый функционал к будущему ПО. Например: анимация «распада» и «склеивания» сложных объектов.

На рисунке 2.16 представлено первоначальное расположение объектов. При выделении объекта и приближении к нему камеры, объект распадается на более мелкие объекты (рисунок 2.17)



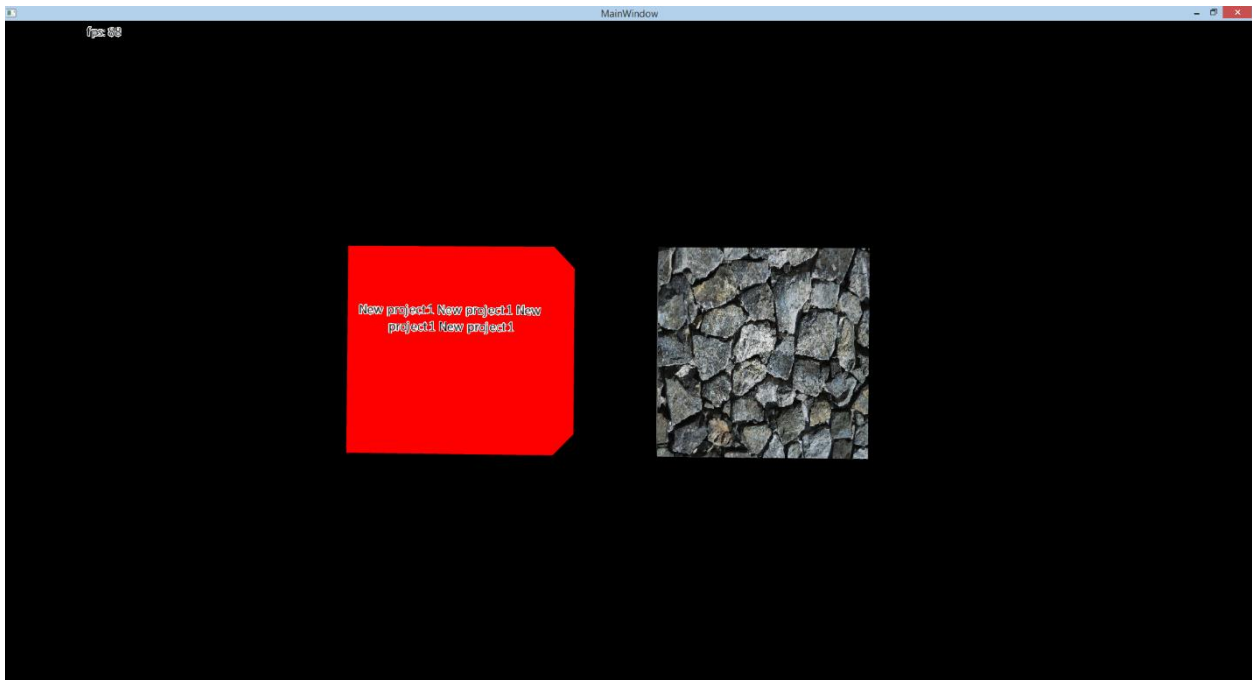


Рисунок 2.16 – первоначальное расположение объектов до «распада»

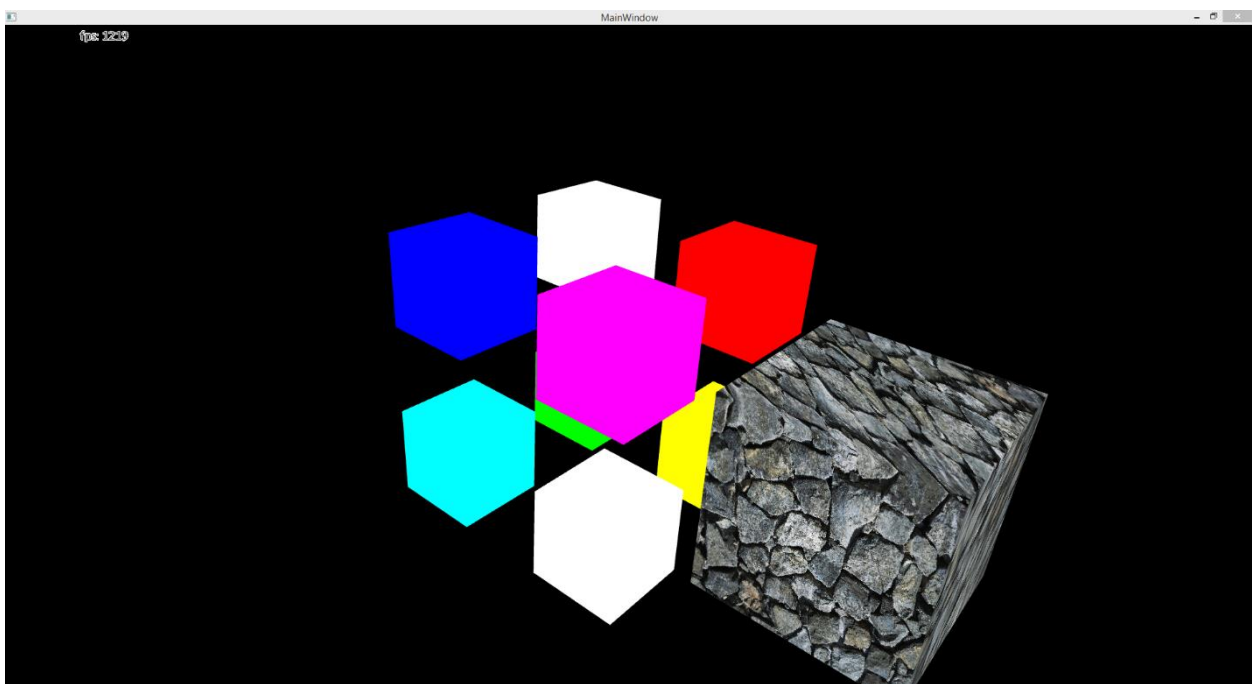


Рисунок 2.17 – распад выделенного объекта

При отдалении камеры на необходимое расстояние объект «склеивается» в первоначальное состояние.

## **Глава 3. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение**

В данном разделе проанализированы трудовые и денежные затраты, необходимые для проектирования, разработки, тестирования и сопровождения разрабатываемой кроссплатформенной программной библиотеки для визуализации объектов.

### **3.1. Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения**

#### **3.1.1. Потенциальные потребители результатов исследования**

Разрабатываемое ПО предназначено для дальнейшего внутреннего использования сотрудниками ООО «Интех». Данная библиотека позволит разработчикам с помощью визуализации примитивов, таких как точки, линии и треугольники, составлять сложные двух- и трехмерные объекты. Не смотря на то, что на рынке программных продуктов существует ряд аналоговых приложений, разработка кроссплатформенной программной библиотеки и ее дальнейшее использование для разработки будущих программных продуктов обеспечит экономическую и функциональную независимость от внешних компаний, имеющих аналоговые продукты.

#### **3.1.2 Анализ конкурентных технических решений**

Сегодня существует целый ряд визуализаторов и программных библиотек для рендеринга объектов. Но при этом бесплатные аналоговые продукты конкурентных компаний не предоставляют нужного функционала для реализации конкурентоспособных коммерческих проектов.

Выделим значимые факторы для сравнения разрабатываемой программной библиотеки с существующими аналоговыми продуктами: свободное бесплатное использование, качество ПО (оптимизированное и

ресурсоэкономное приложение), кроссплатформенность, сопровождение ПО, функциональность.

Таблица 3.1. Конкурентоспособность проекта.

| п/п | Конкуренты   | Факторы конкурентоспособности |             |                      |                  |                  | Итоговая оценка |
|-----|--|-------------------------------|-------------|----------------------|------------------|------------------|-----------------|
|     |  | Цена                          | Качество ПО | Кроссплатформенность | Сопровождение ПО | Функциональность |                 |
|     | Unreal Engine  | 3/0,528                       | 7/2,058     | 7/1,232              | 6/0,708          | 10/2,36          | 6,886           |
|     | Frostbite  | 0/0                           | 8/2,352     | 9/1,584              | 10/1,18          | 8/1,888          | 7,004           |
|     | Unity3D  | 5/0,88                        | 9/2,646     | 9/1,584              | 7/0,826          | 7/1,652          | 7,588           |
|     | Разрабатываемая программная библиотека для визуализации объектов | 10/1,76                       | 7/2,058     | 9/1,584              | 10/1,18          | 8/1,888          | 8,47            |
|     | $b_j$  | 3                             | 5           | 3                    | 2                | 4                | 17              |
|     | $w_j$  | 0,176                         | 0,294       | 0,176                | 0,118            | 0,236            | 1               |

Рассмотрим выделенные критерии у каждого из конкурентов.

Unreal Engine – визуализатор, написанный на языке C++, что делает его мощным рендерером. При этом он не имеет кроссплатформенности и слабо сопровождается по сравнению с остальными визуализаторами. Данный визуализатор является бесплатным, пока доход от разрабатываемого ПО с использованием Unreal Engine не превысит 3 000\$ за квартал. После этого компании-пользователю необходимо предоставлять 5% от доходов.

Frostbite – закрытое ПО, используемое для внутренних проектов и проектов Electronic Arts (EA). Данный визуализатор активно модернизируется для своих проектов, имеет хорошее качество кода, но меньший функционал, чем предыдущих рендерер.

Unity3D – относительно кроссплатформенный визуализатор, написанный на языках C++ и C#, что делает его более удобным в использовании. Цена лицензионного ПО – 1 500\$, либо 75\$ в месяц. Данный визуализатор обладает меньшим функционалом, чем предыдущие

конкуренты и подходит больше для реализации мобильных приложений с использованием двухмерной графики.

Разрабатываемая кроссплатформенная библиотека подразумевает собой ПО для внутреннего использования. Таким образом для организации ее использование является бесплатным. Язык программирования C++ обеспечит реализацию необходимого функционала. Так же стоит отметить, что модернизация данной библиотеки будет проходить сразу, как организации потребуется новые возможности для удовлетворения внутренних потребностей, что делает критерий «Сопровождение» максимальным.

Таким образом можно сделать вывод, что при наличии опытных сотрудников компании выгодно реализовать собственную графическую библиотеку для ее будущего использования в реализации ПО, в котором функциональность опирается на использование качественной графики и оптимизацию.

### **3.1.3. SWOT-анализ**

SWOT-анализ – один из самых часто используемых методов анализа в менеджменте и маркетинге. Данным методом дает ясное представление о текущей ситуации, а также помогает понять, какие действия необходимо предпринять для максимизации возможностей проекта и нейтрализации слабых сторон и угроз.

Целью использования SWOT-анализа для данной разработки является определение возможной эффективности и прогнозирование направлений будущего развития разрабатываемого решения.

Преимуществом SWOT-анализа является разработка связей разнообразных факторов внешней и внутренней среды разработки.

Результаты проведения SWOT-анализа представлены в приложении Б (сводная таблица SWOT-анализа), где указаны сильные и слабые стороны разработки, выявлены возможные направления будущей разработки программных модулей и рассмотрены варианты минимизации влияния угроз.

## 3.2. Планирование научно-исследовательских работ

### 3.2.1. Структура работ в рамках научного исследования

На данном этапе составляется полный список необходимых работ, назначается исполнитель и выставляется продолжительность. Результатом планирования работ является линейных график реализации проекта.

Перечень этапов работы и распределение исполнителей представлен в таблице 3.2.

Таблица 3.2. Структура работ.

| № п/п | Этапы работы  | Исполнители                           |
|-------|---|---------------------------------------|
| 1     | Выбор научного руководителя бакалаврской работы                         | Клюшов П.Д.                           |
| 2     | Составление и утверждение темы бакалаврской работы                      | Ротарь В.Г., Осипов А.В., Клюшов П.Д. |
| 3     | Постановка целей и задач  | Осипов А.В.                           |
| 4     | Разработка и утверждение ТЗ   | Ротарь В.Г., Осипов А.В., Клюшов П.Д. |
| 5     | Подбор и изучение материалов по тематике                                | Осипов А.В., Клюшов П.Д.              |
| 6     | Разработка календарного плана   | Ротарь В.Г., Осипов А.В., Клюшов П.Д. |
| 7     | Проведение анализа предметной области                                   | Клюшов П.Д.                           |
| 8     | Проектирование  | Осипов А.В., Клюшов П.Д.              |
| 9     | Разработка  | Клюшов П.Д.                           |
| 10    | Тестирование  | Осипов А.В., Клюшов П.Д.              |
| 11    | Отладка   | Клюшов П.Д.                           |
| 12    | Выполнение других частей работы (фин. менеджмент, соц. Ответственность) | Клюшов П.Д.                           |
| 13    | Подведение итогов, оформление работы                                    | Клюшов П.Д.                           |

Научный руководитель – Ротарь Виктор Григорьевич;  
Научный консультант – Осипов Артем Владимирович;  
Студент – Клюшов Павел Дмитриевич.

### 3.2.2. Определение трудоемкости выполнения работ и разработка графика проведения научного исследования

Трудовые затраты в большинстве случаев образуют основную часть стоимости разработки, поэтому важным моментом является определение трудоемкости работ каждого из участников научного исследования.

Для определения ожидаемой продолжительности работ  $t_{ож}$  с помощью экспертных оценок были использованы следующие формулы:

$$t_{ожi} = \frac{3t_{\min_i} + 2t_{\max_i}}{5},$$

где  $t_{\min}$  – минимальная продолжительность работ, дн.;

$t_{\max}$  – максимальная продолжительность работ, дн.

$$t_{ож1} = \frac{3 \cdot 90 + 2 \cdot 153}{5} = 115,2 \text{ дн.}$$

Остальные значения рассчитаны по аналогии.

Длительность этапов в рабочих днях  $T_{рд}$  вычислялась по формуле:

$$T_{рд} = t_{ож} \cdot K_{д},$$

где  $K_{д}$  – коэффициент, учитывающий дополнительное время на компенсацию и согласование работ ( $K_{д}=1,2$ ).

$T_{рд}$  равен 138,24 дн., остальные значения рассчитаны по аналогии.

Расчет продолжительности этапа в календарных днях  $T_{кд}$  производился по формуле:

$$T_{кд} = T_{рд} \cdot T_{к},$$

где  $T_{рд}$  – продолжительность выполнения этапа в рабочих днях;

$T_{к}$  – коэффициент календарности.

Коэффициент календарности рассчитывается по формуле:

$$T_{к} = \frac{T_{кал}}{T_{кал} - T_{вд} - T_{пд}}$$

где  $T_{КАЛ}$  – календарные дни ( $T_{КАЛ} = 365$ );

$T_{ВД}$  – выходные дни ( $T_{ВД} = 110$ );

$T_{ПД}$  – праздничные дни ( $T_{ПД} = 8$ ).

Коэффициент календарности  $T_K$  равен 1,48. Продолжительность проекта в календарных днях  $T_{КД}$  равна 204,6. Остальные значения рассчитаны аналогично.

Все расчеты по трудозатратам представлены в приложении Б, итоги по продолжительности этапов работы в рабочих и календарных днях являются общими трудоемкостями для каждого из участников проекта, они будут использованы для дальнейших расчетов.

### **3.2.3. Бюджет научно-технического исследования**

Для проекта по разработке программных модулей управления проектами производится оценка затрат по следующим статьям:

- материалы и покупные изделия;
- заработная плата;
- социальный налог;
- расходы на электроэнергию (без освещения);
- амортизационные отчисления;
- прочие расходы;
- аренда помещения.

Так как работа по проекту выполнялась без привлечения сторонних организаций, а также не было необходимости в командировках, расходы по соответствующим статьям отсутствуют.

#### **3.2.3.1. Расчет материальных затрат научно-технического исследования**

В материальных затратах учтены только расходы на канцелярские принадлежности и картриджи для принтера, так как все необходимые для работы над проектом материалы имелись в распоряжении исполнителей. Материалы, необходимые для выполнения данной работы, и расчет материальных затрат представлены в таблице 3.3.

Таблица 3.3. Материальные затраты

| Наименование материалов | Цена за ед., руб. | Кол-во | Сумма, руб.   |
|-------------------------|-------------------|--------|---------------|
| Бумага для принтера, А4 | 240,00            | 1 уп.  | 240,00        |
| Ручка шариковая         | 20,00             | 2 шт.  | 40,00         |
| <b>Итого:</b>           |                   |        | <b>320,00</b> |

### 3.2.3.2. Расчет амортизационных затрат

В специальное оборудование входят оборудование для рабочего места и ПО, необходимое для реализации программного обеспечения.

Таблица 3.4. Затраты на специальное оборудование.

| Наименование оборудования                              | Количество единиц оборудования | Цена за 1 ед. оборудования | Затраты, руб.  |
|--|--------------------------------|----------------------------|----------------|
| Персональный компьютер                                 | 1                              | 30 000                     | 30 000         |
| ЖК монитор   | 1                              | 4 500                      | 5 000          |
| 1 ТБ Жесткий диск Seagate 7200 BarraCuda [ST1000DM010] | 1                              | 2 999                      | 2 999          |
| Клавиатура   | 1                              | 599                        | 599            |
| Компьютерная мышь                                      | 1                              | 199                        | 199            |
| Windows 10 Pro   | 1                              | 10 799                     | 10 799         |
| Qt Creator   | 1                              | 91 800                     | 91 800         |
| <b>Итого:</b>  |                                |                            | <b>141 396</b> |



Амортизационные отчисления для рассматриваемого проекта включают в себя амортизацию используемого оборудования за время выполнения работы. Амортизационные отчисления рассчитываются по времени использования компьютера по формуле:

$$C_{AM} = \frac{N_A \cdot C_{OB}}{F_D} \cdot t_{рф} \cdot n$$

где:  $N_A$  – годовая норма амортизации;  $C_{OB}$  – цена оборудования;

$F_D$  – действительный годовой фонд рабочего времени;

$t_{рф}$  – время работы вычислительной техники;

$n$  – число задействованных единиц оборудования,  $n = 1$ .

Годовая амортизация  $N_A$  определяется как величина, обратная сроку амортизации оборудования  $C_A$ , который определяется согласно постановлению правительства РФ «О классификации основных средств, включенных в амортизационные группы». Для компьютера и периферийного оборудования, использующегося с ним примем  $C_A = 3$  года, тогда  $N_A = 0,33$ .

Примем, что ПО для разработки библиотеки Qt Creator, используется  $\frac{3}{4}$  от времени эксплуатации компьютера. Остальное время занимает использование бесплатного ПО: браузер, текстовые и графические редакторы и т.д..

Расчет затрат на амортизационные отчисления представлен в таблице 3.5.

Таблица 3.5. Затраты на амортизационные отчисления

| Наименование оборудования | Норма амортиз. оборуд., $N_A$ | Стоим. оборуд., Цоб, руб. | Факт. р/вр. оборуд., $t_{рф}$ , ч | Действ. год. фонд р/вр., $F_d$ , ч. | Аморт. отчисл., $C_{ам}$ , руб. |
|---------------------------|-------------------------------|---------------------------|-----------------------------------|-------------------------------------|---------------------------------|
| Персональный компьютер    | 0,33                          | 30 000,00                 | 1 440                             | 1 944                               | 7 333,3                         |
| ЖК монитор                | 0,33                          | 4 500,00                  | 1 440                             | 1 944                               | 1 098                           |
| Жесткий диск              | 0,33                          | 2 999                     | 1 440                             | 1 944                               | 731,76                          |
| Клавиатура                | 0,33                          | 599                       | 1 440                             | 1 944                               | 146,16                          |
| Компьютерная мышь         | 0,33                          | 199                       | 1 440                             | 1 944                               | 48,56                           |
| Windows 10 Pro            | 0,33                          | 10 799                    | 1 440                             | 1 944                               | 2 634,96                        |
| Qt Creator                | 0,33                          | 91 800                    | 1 080                             | 1 944                               | 16 830                          |
| <b>Итого:</b>             |                               |                           |                                   |                                     | <b>28822,74</b>                 |

### 3.2.3.3. Расчет заработной платы

Расчет основной заработной платы выполняется на основе трудоемкости выполнения каждого этапа и величины месячного оклада исполнителя. Месячный оклад (МО) научного руководителя, занимающего должность доцента и имеющего степень кандидата технических наук, составляет 33 664,00 руб./мес., МО научного консультанта составляет 50 000 руб./мес., МО исполнителя, являющегося студентом, составляет 12 500 руб./мес.

Таблица 3.6. Расчет годового фонда рабочего времени.

| Показатели рабочего времени                           | Дни |
|---|-----|
| Календарные дни                                       | 365 |
| Нерабочие дни (праздники, выходные)                   | 66  |
| Потери рабочего времени (отпуск, невыходы по болезни) | 56  |
| Действительных годовых фонд рабочего времени          | 243 |

$$Здн = \frac{Зм * М}{Fд}$$

Зм – месячный оклад работника, руб

М – количество месяцев работы без отпуска в течение года:

- при отпуске в 24 раб. дня М =11,2 месяца, 5-дневная неделя;
- при отпуске в 48 раб. дней М=10,4 месяца, 6-дневная неделя.

Fд – действительный годовых фонд рабочего времени персонала, раб. дн.

Кпр – премиальный коэффициент (0,3);

Кд – коэффициент доплат и надбавок (0,2-0,5);

Кр – районный коэффициент (для Томска 1,3);

Тр – продолжительность работ, выполняемых работником, раб. Дни

$$Зосн = Здн * Тр * (1 + Кпр + Кд) * Кр$$

$$Зосн = Здн * Тр * (1 + Кпр + Кд) * Кр$$

Таблица 3.7. Расчет основной заработной платы.

| Исполнители             | Здн, руб | Кпр | Кд  | Кр  | Тр    | Зосн       |
|-------------------------|----------|-----|-----|-----|-------|------------|
| Студент                 | 576,13   | 0,3 | 0,2 | 1,3 | 99,7  | 112 008,31 |
| Научный<br>руководитель | 1 551,59 | 0,3 | 0,4 | 1,3 | 2,4   | 8 229,63   |
| Научный<br>консультант  | 2 304,53 | 0,3 | 0,4 | 1,3 | 35,36 | 180 088,88 |
| Итого:                  |          |     |     |     |       | 300 326,82 |

Исходя из таблицы 3.7 рассчитаем внебюджетные фонды (страховые отчисления) для работников ООО «Интех»: студент, научный консультант:

$$\Phi = \frac{(\text{Здоп} + \text{Зосн}) * 30\%}{100\%}$$

Для учреждений осуществляющих образовательную и научную деятельность в 2019г. водится пониженная ставка 28% (п. 6 ч. 1 ст. 58 Закона 212-ФЗ).

Зосн – основная заработная плата;

Здоп – дополнительная заработная плата исполнителей темы.

$$\text{Здоп} = \frac{\text{Зосн} * 12\%}{100\%}$$

Таблица 3.8. Расчет страховых отчислений.

| Исполнители          | Дополнительная<br>заработная плата | Отчисления<br>во<br>внебюджетные фонды |
|----------------------|------------------------------------|--|
| Научный руководитель | 987,56                             | 2 580,81                               |
| Научный консультант  | 21 610,67                          | 60 509,87                              |
| Студент              | 13 440,99                          | 37 634,79                              |
| Итого                | 36039,22                           | 100 725,47                             |

#### 1.2.3.4. Расчет затрат на электроэнергию

Данный вид расходов включает в себя затраты на электроэнергию при работе оборудования, а именно компьютера. Затраты на электроэнергию при работе оборудования  $C_{\text{эл.об.}}$  рассчитываются по формуле:

$$C_{\text{эл.об.}} = P_{\text{об}} \cdot Ц_{\text{э}} \cdot t_{\text{об}},$$

где  $P_{\text{об}}$  – мощность, потребляемая оборудованием, кВт;

$Ц_{\text{э}}$  – тарифная цена за 1 кВт·час;  $t_{\text{об}}$  – время работы оборудования, час.

Мощность  $P_{\text{об}}$ , потребляемая оборудованием, определяется по формуле:

$$P_{\text{об}} = P_{\text{ном.}} \cdot K_{\text{с}},$$

где  $P_{\text{ном.}}$  – номинальная мощность оборудования, кВт;

$K_{\text{с}}$  – коэффициент загрузки (для технологического оборудования малой мощности  $K_{\text{с}} = 1$ ).

Номинальная мощность персонального компьютера составляет 0,3 кВт.

Тарифная цена за 1 кВт час = 2,39 руб./кВт час.

Время работы оборудования  $t_{\text{об}}$  для исполнителя вычисляется на основе данных таблицы 3.8, где указаны трудозатраты проекта:

$$t_{\text{об}} = T_{\text{рд}} \cdot K_t,$$

где  $K_t \leq 1$  – коэффициент использования оборудования по времени, равный отношению времени его работы в процессе выполнения проекта к  $T_{\text{рд}}$ .

Из расчета, что продолжительность рабочего дня равна 8 часов, а работа выполнялась 204,6 рабочих дней, получим, что общее время выполнения проекта составляет 1 636,8 часа.

Так как работа на компьютере проводилась по 7 часов в день из 8, то  $K_t = 0,88$ . Тогда из 1 636,8 часов, потраченных исполнителем на осуществление проекта, 1 440 часов были проведены за компьютером.

Затраты на электроэнергию при работе оборудования сведены в таблицу 3.9.

Таблица 3.9. Затраты на электроэнергию для технологических целей

| Наименование оборудования | Время работы оборудования<br>t <sub>об</sub> , час | Потребляемая мощность<br>P <sub>об</sub> , кВт | Затраты<br>Э <sub>об</sub> , руб. |
|---------------------------|--|--|-----------------------------------|
| Персональный компьютер    | 1 440  | 0,3  | 1 032,5                           |

### 3.2.3.5. Формирование бюджета затрат научно-исследовательского проекта

Прежде чем приступать к формированию бюджета необходимо рассчитать накладные расходы:

$$НР = \frac{(З_{мат} + З_{со} + З_{осн} + З_{доп} + \Phi) * 16\%}{100\%}$$

Где З<sub>мат</sub> – материальные затраты;

З<sub>со</sub> – затраты на специальное оборудование;

З<sub>осн</sub> – затраты на основную зар. плату;

З<sub>доп</sub> – затраты на дополнительную зар. плату;

Φ – страховые взносы.

Таблица 3.10. Смета затрат на разработку проекта

| Статья затрат                        | Сумма, руб. | Удельный вес, % |
|--------------------------------------|-------------|-----------------|
| Материалы и покупные изделия         | 320,00      | 0,047           |
| Затраты на спец. оборудование        | 141 396     | 21,02           |
| Затраты на основную зар. плату       | 300 326,82  | 44,66           |
| Затраты на дополнительную зар. плату | 36 039,22   | 5,358           |
| Страховые взносы                     | 100 725,47  | 14,98           |
| Расходы на электроэнергию            | 1 032,5     | 0,153           |
| Накладные расходы                    | 92 609,2    | 13,772          |
| Итого:                               | 672 449,21  | 100             |

Общая себестоимость проекта получилась равной 672 449,21 рубля.

### **3.3. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования**

Определение эффективности происходит на основе расчета интегрального показателя эффективности научного исследования. Его нахождение связано с определением двух средневзвешенных величин: финансовой эффективности и ресурсоэффективности.

Интегральный показатель финансовой эффективности научного исследования получают в ходе оценки бюджета затрат трех (или более) вариантов исполнения научного исследования. Для этого наибольший интегральный показатель реализации технической задачи принимается за базу расчета (как знаменатель), с которым соотносятся финансовые значения по всем вариантам исполнения.

Интегральный финансовый показатель разработки определяется как:

$$I_{\text{финр}}^{\text{исп.}i} = \frac{\Phi_{pi}}{\Phi_{\text{max}}},$$

где  $I_{\text{финр}}$  – интегральный финансовый показатель разработки;

$\Phi_{pi}$  – стоимость  $i$ -го варианта исполнения;

$\Phi_{\text{max}}$  – максимальная стоимость исполнения научно-исследовательского проекта (в т.ч. аналоги).

Полученная величина интегрального финансового показателя разработки отражает соответствующее численное увеличение бюджета затрат разработки в размах (значение больше единицы), либо соответствующее численное удешевление стоимости разработки в размах (значение меньше единицы, но больше нуля).

Интегральный показатель ресурсоэффективности вариантов исполнения объекта исследования можно определить следующим образом:

$$I_{pi} = \sum a_i \cdot b_i,$$

где  $I_{pi}$  – интегральный показатель ресурсоэффективности для  $i$ -го варианта исполнения разработки;

$a_i$  – весовой коэффициент  $i$ -го варианта исполнения разработки;

$b_i$ , – бальная оценка  $i$ -го варианта исполнения разработки, устанавливается экспертным путем по выбранной шкале оценивания;

$n$  – число параметров сравнения.

Расчет интегрального показателя ресурсоэффективности приведен в таблице 3.11.

Таблица 3.11. Сравнительная оценка характеристик вариантов исполнения проекта

| Объект исследования/Критерии  | Весовой коэффициент параметра | И1 | И2 | И3 |
|---|-------------------------------|----|----|----|
| Удобство эксплуатации (соответствует требованиям потребителей) в      | 0,1                           | 3  | 5  | 3  |
| Оптимизированные и ресурсоэффективные алгоритмы (производительность)ч | 0,4                           | 4  | 4  | 5  |
| Надежность  | 0,3                           | 5  | 3  | 4  |
| Кроссплатформенность  | 0,2                           | 4  | 5  | 4  |
| <b>Итого</b>  | 1                             |    |    |    |

Тогда значение интегрального показателя для каждого использования будет:

$$I_{p-исп1} = 4,2$$

$$I_{p-исп2} = 4,4$$

$$I_{p-исп3} = 4,3$$

Интегральный показатель эффективности вариантов исполнения разработки ( $I_{исп}$ ) определяется на основании интегрального показателя ресурсоэффективности и интегрального финансового показателя по формуле:



$$I_{исп.i} = \frac{I_{p-испi}}{I_{финр}}$$

Сравнение интегрального показателя эффективности вариантов исполнения разработки позволит определить сравнительную эффективность проекта и выбрать наиболее целесообразный вариант из предложенных. Сравнительная эффективность проекта ( $\mathcal{E}_{cp}$ ):

$$\mathcal{E}_{cp} = \frac{I_{исп.i}}{I_{исп.i+1}}$$

В таблице 3.12 приведены результаты сравнения эффективности разработки.

Таблица 3.12. Сравнительная эффективность разработки

| № п/п | Показатели  | Исп.1 | Исп.2 | Исп.3 |
|-------|---|-------|-------|-------|
| 1     | Интегральный финансовый показатель разработки           | 1     | 1     | 1     |
| 2     | Интегральный показатель ресурсоэффективности разработки | 4.2   | 4.4   | 4.3   |
| 3     | Интегральный показатель эффективности                   | 4.2   | 4,4   | 4,3   |
| 4     | Сравнительная эффективность вариантов исполнения        | 0,95  | 1,02  | 1,02  |

Из полученной таблицы видно, что наиболее эффективный вариант решения поставленной задачи с позиции финансовой и ресурсной эффективности является исполнение 1.

### 3.4 Вывод

Подводя итог, можно сделать следующий вывод: разрабатываемое ПО будет использоваться внутри организации для дальнейшего создания коммерческих программных продуктов, включающие в себя использование качественной графики.

Подразумевается, что в дальнейшем данный проект будет модифицироваться и расширять свой функционал, из-за чего дальнейшая себестоимость и длительность разработки увеличатся. Но дальнейшая модификация станет возможной после реализации первоначальной части проекта (основного функционала программной библиотеки: реализация под платформу рендеринга OpenGL и DirectX) и после ее окупаемости первыми коммерческими проектами.

## Глава 4. Социальная ответственность.

### Введение

Трудовая деятельность разработчика программных систем связана с воздействием производственных факторов различного характера. Для предупреждения вредного воздействия и сохранения здоровья работника предусмотрен ряд мер по обеспечению безопасности трудовой деятельности.

В данном разделе проведен анализ вредных и опасных факторов труда, определен комплекс мер организационного, правового, технического и режимного характера, который должен способствовать снижению возможности возникновения негативных последствий работы разработчика.

Выпускная квалификационная работа по разработке программной библиотеки для определения, построения и визуализации трехмерных и двумерных объектов выполнялась в ходе преддипломной практики в компании ООО «Интех». Проектируемое рабочее место представляет собой офисное помещение, в котором будет работать разработчик.

Характеристика помещения:

- ширина рабочего помещения – 5 м, длина – 15 м, высота – 5 м;
- площадь помещения – 75 м<sup>2</sup>;
- объем помещения – 375 м<sup>3</sup>;
- в помещении установлен кондиционер, имеется естественная вентиляция – вытяжное вентиляционное отверстие, дверь, окна, щели;
- в помещении установлено искусственное освещение, имеется естественное освещение.

В данном помещении оборудовано одиннадцать рабочих мест, максимальное количество сотрудников в одну смену – 11. В среднем на одного сотрудника приходится 6,8 м<sup>2</sup> площади и около 34 м<sup>3</sup> объема помещения. Данное размещение сотрудников удовлетворяет санитарным нормам, согласно которым на одного работника должно приходиться не

менее 6 м<sup>2</sup> площади и 24 м<sup>3</sup> объема рабочего помещения, с учетом максимального числа одновременно работающих в смену.

#### **4.1. Правовые и организационные вопросы обеспечения безопасности**

При организации рабочего места с ПК необходимо учитывать требования безопасности, промышленных санитарных норм, эргономики и технической эстетики.

Рабочее место должно быть организовано с учетом требований ГОСТ 12.2.032-78 «ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования» и СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы».

Согласно требованиям при организации работы с ПК должны выполняться следующие условия:

- площадь на одно рабочее место пользователя с ПК должна составлять не менее 6 м<sup>2</sup>;
- конструкция рабочей мебели должна обеспечивать возможность индивидуальной регулировки соответственно росту пользователя и создавать удобную позу для работы;
- ПК и, соответственно, рабочее место должно располагаться так, чтобы свет падал сбоку, лучше слева;
- расстояние от ПК до стен должно быть не менее 1 м, по возможности следует избегать расположения рабочих мест в углах помещения либо лицом к стене;
- ПК лучше установить так, чтобы, подняв глаза от экрана, можно было увидеть какой-нибудь удаленный предмет в помещении или на улице, таким образом, предоставляя эффективный способ разгрузки зрительного аппарата;
- окна в помещениях с ПК должны быть оборудованы регулируемыми устройствами – жалюзи, занавески, внешние козырьки;

- монитор, клавиатура и корпус компьютера должны находиться прямо перед работником;
- высота рабочего стола с клавиатурой должна составлять 680-800 мм над уровнем стола;
- высота экрана над полом – 900-1280 мм, монитор должен находиться на расстоянии 600-700 мм от работника на 20 градусов ниже уровня глаз;
- рабочее кресло должно иметь мягкое сиденье и спинку, с регулировкой сиденья по высоте с удобной опорой для поясницы
- положение тела пользователя относительно монитора должно соответствовать направлению просмотра под прямым углом 90 градусов или под углом 75 градусов.

В соответствии с Трудовым кодексом РФ 197-ФЗ предусмотрена рациональная организация труда в течение смены, согласно которой:

- длительность рабочей смены должна быть не более 8 часов;
- должны быть установлены два регламентируемых перерыва - не менее 20 минут после 1-2 часов работы или не менее 30 минут после 2 часов работы;
- обеденный перерыв должен быть не менее 40 минут, может быть скользящим в течение рабочей смены.

Также, Трудовым кодексом закреплён обязательный предварительный медицинский осмотр при приеме на работу и периодические медицинские осмотры.

Каждый сотрудник должен пройти инструктаж по технике безопасности перед приемом на работу и в дальнейшем, должен быть пройден инструктаж по электробезопасности и охране труда. Каждому работнику обязательно должна быть предоставлена рабочая инструкция, с описанием входящих в его должность функций и рабочих моментов, а также конкретным описанием границ ответственности.

При выполнении ВКР на представленном рабочем месте нарушения правовых и организационных норм не было, рабочее место оборудовано согласно санитарным и эргономическим нормам, организация рабочего времени согласно регламентированным нормам.

## 4.2. Производственная безопасность

Таблица 4.1. Вредные факторы производства.

| Факторы<br>(ГОСТ 12.0.003-2015)                 | Этапы работ    |                  | Нормативные<br>документы  |
|---|----------------|------------------|---|
|   | Разрабо<br>тка | Эксплуа<br>тация |   |
| 1.повышенный уровень электромагнитных излучений | +              | +                | ГОСТ 12.1.006-84 ССБТ. Электромагнитные поля радиочастот [6].   |
| 2.отклонение показателей микроклимата           | +              | +                | СанПиН 2.2.4.548–96. Гигиенические требования к микроклимату производственных помещений [7].  |
| 3.недостаточная освещенность рабочей зоны;      | +              | +                | СП 52.13330.2016. Естественное и искусственное освещение [8].   |
| 4.повышенный уровень шума на рабочем месте;     | +              | +                | ГОСТ 12.1.003-2014 ССБТ. Шум. Общие требования безопасности [9].  |
| 5.эмоциональные перегрузки;                     | +              | +                | Р 2.2.2006–05. Гигиена труда. Руководство по гигиенической оценке факторов рабочей среды и трудового процесса. Критерии и классификация условий труда [10]. |
| 6.нарушение правил электробезопасности          | +              | +                | ГОСТ 12.1.030-81 ССБТ. Электробезопасность. Защитное заземление. Зануление [11].  |

При анализе работы химические и биологические факторы не оказывают существенного влияния на состояние здоровья разработчиков

программных систем, поэтому рассмотрим только физические и психофизиологические факторы.

Вредными и опасными производственными факторами являются:

- повышенный уровень электромагнитных излучений;
- отклонение показателей микроклимата;
- недостаточная освещенность рабочей зоны;
- повышенный уровень шума на рабочем месте;
- монотонность труда;
- эмоциональные перегрузки;
- нарушение правил электробезопасности.

Проведем анализ всех вышеперечисленных факторов и определим соответствие рабочего места установленным санитарным нормам.

#### **4.2.1. Повышенный уровень электромагнитных излучений**

Главным рабочим аппаратом разработчика программных систем является персональный компьютер (ПК), который в период работы подвергает работника вредному электромагнитному излучению. Электромагнитное излучение ПК сложное по спектральному составу, изменяется в диапазоне частот от 0 Гц до 1000 МГц. Такое излучение состоит из электрической (Е) и магнитной (Н) составляющих.

Норма допустимых уровней напряженности полей и излучений регламентируются СанПиН 2.2.4.1191-03 и нормами Госкомсанэпиднадзора «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы» (СанПиН 2.2.4.1340-03). По установленным нормам время пребывания работника в рабочей зоне вычисляется по формуле:

$$T = (50/E) - 2.$$

Так, например, при напряженности до 5 кВ/м присутствие работника в рабочей зоне разрешается в течение 8 часов, а при напряженности 20-25 кВ/м время присутствия работника в рабочей зоне сокращается до 10 минут.

В предполагаемом рабочем месте уровень напряженности электрических полей не превышает значения 4 кВ/м, при котором разрешенное время пребывания в рабочей зоне может составлять до 10,5 часов. Рабочая смена длится 8 часов, следовательно, уровень электромагнитных излучений на рабочем месте в норме. Таким образом можно сделать вывод, что сегодня мониторы почти не оказывают сильного излучения на организм человека. При этом мониторы вредно влияют на зрение человека.

#### **4.2.2. Отклонение показателей микроклимата**

Под микроклиматом рабочего помещения понимают климат внутренней среды помещения, в котором находятся сотрудники в течение рабочего времени. Микроклимат определяется совокупностью показателей, действующих на организм работника, - температуры воздуха и поверхностей, относительной влажности воздуха, скорости движения воздуха и интенсивности теплового облучения.

Нормативные показатели микроклимата регламентируются СанПиН 2.2.4.548-96 «Гигиенические требования к микроклимату производственных помещений». Санитарные нормы устанавливают оптимальные и допустимые значения показателей в рабочей зоне, что позволяет создавать благоприятные условия работы, соответствующие физиологическим потребностям организма человека. Для поддержания и доведения микроклиматических показателей до нормативных значений проводятся мероприятия, которые обязательно должны включаться в комплексные планы предприятий по охране труда.

Работа, выполняемая разработчиком программных систем, относится к категории Ia, так как она является мало подвижной и мало интенсивной, проводится сидя с минимальными физическими напряжениями. В таблицах 4.2 и 4.3 представлены оптимальные и допустимые значения показателей микроклимата на рабочих местах для данной категории.



Таблица 4.2. Оптимальные величины показателей микроклимата на рабочих местах производственных помещений (СанПиН 2.2.4.548-96 [7])

| Период года | Температура воздуха, °С | Температура поверхностей, °С | Относительная влажность воздуха, % | Скорость движения воздуха, м/с |
|-------------|-------------------------|------------------------------|------------------------------------|--------------------------------|
| Холодный    | 22-24                   | 21-25                        | 60-40                              | 0,1                            |
| Теплый      | 23-25                   | 22-26                        | 60-40                              | 0,1                            |

Таблица 4.3. Допустимые величины показателей микроклимата на рабочих местах производственных помещений (СанПиН 2.2.4.548-96)

| Период года | Температура воздуха, °С           |                                   | Температура поверхностей, °С | Относительная влажность воздуха, % | Скорость движения воздуха, м/с                                      |   |
|-------------|-----------------------------------|-----------------------------------|------------------------------|------------------------------------|---|---|
|             | диапазон ниже оптимальных величин | диапазон выше оптимальных величин |                              |                                    | для диапазона температур воздуха ниже оптимальных величин, не более | для диапазона температур воздуха выше оптимальных величин, не более |
| Холодный    | 20,0 - 21,9                       | 24,1 - 25,0                       | 19,0 - 26,0                  | 15 – 75                            | 0,1   | 0,1   |
| Теплый      | 21,0 - 22,9                       | 25,1 - 28,0                       | 20,0 - 29,0                  | 15 – 75                            | 0,1   | 0,2   |

Значения показателей, полученные при измерении на рабочем месте:

- температура воздуха 23,5 °С – оптимальное значение;
- температура поверхностей 22 °С – оптимальное значение;
- относительная влажность воздуха 65% – допустимое значение;
- скорость движения воздуха 0,1 м/с – оптимальное значение.

Все измеренные показатели удовлетворяют санитарным нормам для рабочих помещений.

#### 4.2.3. Недостаточная освещенность рабочей зоны

Низкая освещенность рабочей зоны губительно влияет на органы зрения работников, снижает зрительную работоспособность, а также влияет на настроение и общее самочувствие работников, определяет эффективность

выполнения работы. Нерациональная организация освещения является одной из причин травматизма на рабочем месте, так как ухудшение видимости объектов и неадекватное восприятие наблюдаемых предметов может быть спровоцировано плохо освещенными опасными зонами, слепящими источниками света, световыми бликами, резкими тенями, а также пульсацией световых источников.

В помещениях для работы с персональными компьютерами должно быть естественное и искусственное освещение. Нормативные показатели естественного, искусственного и совмещенного освещения в соответствии с СанПиН 2.2.1/2.1.1.1278-03[8] представлены в таблице 4.4.

Таблица 4.4. Нормируемые показатели естественного, искусственного и совмещенного освещения (СанПиН 2.2.1/2.1.1.1278-03)

| Помещения                                 | Рабочая поверхность и плоскость нормирования КЕО и освещенности и высота плоскости над полом, м | Естественное освещение                    |                       | Совмещенное освещение                     |                       | Искусственное освещение       |            |                     |                                     |  |
|---|---|---|-----------------------|---|-----------------------|-------------------------------|------------|---------------------|-------------------------------------|--|
|   |   | КЕО ен, %                                 |                       | КЕО ен, %                                 |                       | освещенность, лк              |            |                     | показатель дисконтности М, не более | коэффициент пульсации освещенности, Кп, % не более |
|   |   | при верхнем или комбинированном освещении | при боковом освещении | при верхнем или комбинированном освещении | при боковом освещении | при комбинированном освещении |            | при общем освещении |                                     |  |
|   |   |   |                       |   |                       | все го                        | от общ его |                     |                                     |  |
| Помещения для работы с дисплеями залы ЭВМ | Г-0,8<br>Экран монитора<br>:<br>В-1,2   | 3,5<br>-                                  | 1,2<br>-              | 2,1<br>-                                  | 0,7<br>-              | 500<br>-                      | 300<br>-   | 400<br>200          | 15<br>-                             | 10<br>-  |

На представленном рабочем месте использовано сочетание естественного и искусственного освещения, то есть освещение смешанного типа.

Естественным освещением помещение обеспечивается за счет оконных проемов, освещение должно быть с левой стороны от работника.

Для искусственного освещения помещений с персональными компьютерами рекомендовано применение светильников типа ЛПО 2х36. Расположение светильников рекомендуется линиями, так, чтобы при разном положении ПК светильники были параллельно линии зрения работника. Защитный угол светильников должен быть не менее 40 градусов.

Рассматривая представленное рабочее место, установим, что естественное освещение в помещении осуществляется через три оконных проема размером 2х1,5 метра в наружной стене. Искусственный свет в помещении представлен 10 светильниками типа ЛПО 36, расположенными линиями, что дает непрерывное и равномерное освещение. В каждом светильнике установлено 4 люминесцентные лампы типа ЛБ-40.

Проведем расчет освещенности рабочего места. Исходными данными являются размеры помещения 5х15х5 м, световой поток используемых ламп равен 900 лк. Стены и потолок в помещении имеют отделку белого цвета, пол серого цвета, следовательно, индексы отражения для потолка и стен равны 80, для пола – 30.

Так как должность разработчика предполагает длительные монотонные операции с высоким уровнем зрительной работы, то есть различение объектов, размером от 3 до 5 мм, такая работа класса III, необходимо принять за норму освещенности рабочего места от 300 до 500 лк.

Коэффициент запаса, показывающий поправку на запыленность источников освещения, примем 1,2, так как запыленность значительно меньше 1 мг/м<sup>3</sup>.

Определяем индекс помещения по формуле:

$$I_{\text{П}} = \frac{S}{((h_1 - h_2) \times (a + b))}$$

где  $I_{\text{П}}$  – индекс помещения;  $S$  – площадь;  $h_1$  – высота потолков;  $h_2$  – высота рабочего стола;  $a$  – длина помещения;  $b$  – ширина помещения.

Для представленного рабочего места рассчитаем:

$$I_{\Pi} = \frac{75}{((5 - 0,8) \times (5 + 5))} = 1,78$$

По полученному индексу помещения определим, что коэффициент использования помещения  $U$  равен 76.

Проведем расчет освещенности по следующей формуле:

$$E = \frac{K_{\text{СВ}} \times K_{\text{Л}} \times \text{СП}_{\text{Л}} \times U}{S \times k_3 \times 100}$$

где  $K_{\text{СВ}}$  – количество светильников;  $K_{\text{Л}}$  – количество лампочек в светильнике;  $\text{СП}_{\text{Л}}$  – световой поток лампочки;  $U$  – коэффициент использования;  $S$  – площадь;  $k_3$  – коэффициент запаса.

Для представленного рабочего места получим:

$$E = \frac{10 \times 4 \times 900 \times 76}{75 \times 1,2 \times 100} = 304$$

Получено значение освещенности в 304 лк, следовательно, освещение рабочего места соответствует нормативным значениям.

#### **4.2.4. Повышенный уровень шума на рабочем месте**

Шумом называется совокупность различных звуков, возникающих в процессе производства и неблагоприятно воздействующих на организм работника, так как под воздействием шума нарушаются физиологические функции, уменьшается концентрация внимания, проявляется усталость и напряжение. Таким образом, шум уменьшает работоспособность и снижает производительность работника.

Для различных категорий рабочих помещений нормативные уровни шума регламентируются ГОСТ 12.1.003-83[9]. Помещения для работы с ПК не могут граничить с помещениями с повышенным уровнем шума. При выполнении работы на ПК уровень шума на рабочем месте не должен превышать 50 дБА. Оборудование, превышающее нормативный уровень шума должны находиться вне помещения для работы с ПК.

В представленном рабочем помещении основными источниками шума являются персональные компьютеры, оргтехника и кондиционер. С учетом

максимального числа работников в смену уровень шума равен 58 дБА, что немного превышает нормативное значение.

Для снижения уровня шума можно:

- использовать звукопоглощающие материалы для обшивки стен;
- использовать звукопоглощающие экраны и перегородки;
- проверить настройки оборудования, так как при некорректных настройках уровень шума от оборудования может увеличиваться, например, из-за перегрева;
- использовать беруши;
- использование бесшумных персональных компьютеров нового поколения.

#### **4.2.5. Эмоциональные перегрузки**

Работа разработчика программных систем является умственным трудом. Умственный труд подразделяется на три типа: сенсорный, сенсомоторный и логический. Труд разработчика является логическим и заключается в приеме информации, ее переработке и выработке решения. При такой работе большое значение имеет процесс мышления, выбор оптимального решения из ряда логических вариантов, создание нового подхода к решению задачи. Поиск решения связан с опытом, знаниями, особенностями нервной системы человека.

Умственный труд является напряженным, от чего страдают зрительные и слуховые анализаторы, центральная нервная система, в особенности высшие психические функции – память, мышление и воображение.

К факторам возникновения эмоциональных перегрузок можно отнести: длительное эмоциональное напряжение, хроническую усталость, хроническое нарушение режимов труда и отдыха, социальные перемены, значимые жизненные трудности и так далее.

Вследствие влияния таких факторов, у работника начинаются проявления последствий: снижение интереса к работе и работоспособности, проявление раздражительности и конфликтности, повышение количества ошибок в работе, психоэмоциональные сдвиги.

К мероприятиям по профилактике и снижению эмоциональных перегрузок можно отнести:

- умственные тренировки и повышение квалификации;
- умеренную и постоянную производственную нагрузку;
- улучшение культуры труда и быта, выработку силы воли;
- правильное трудовое, психологическое и эстетическое воспитание;
- развитие умения отвлекаться от того, что вызвало стрессовое состояние;
- повышение интереса к работе;
- создание условий для возникновения положительных эмоций;
- оптимальную организацию отдыха.

Помимо эмоциональных перегрузок связанных с умственным трудом на здоровье офисного работника (в частности программиста) так же влияет монотонность труда.

Монотонным трудом называется однообразие трудовых операций или производственной обстановки, то есть объективные факторы трудовой деятельности. Монотонность является достаточно серьезным негативным фактором, так как многие виды работы требуют от работника длительного выполнения однообразных действий или непрерывной и устойчивой концентрации внимания в условиях дефицита сенсорных нагрузок.

В условиях монотонной работы, а именно с проведением сидячего образа жизни, держа руки постоянно в одном положении и не изменяя направление взгляда, с организмом человека могут произойти следующие изменения:

- изменение функционального состояния центральной нервной системы;
- удлинение латентного периода зрительно моторных реакций;
- снижение уровня бодрствования;
- нарушение автоматизма деятельности;
- нарушение способности к переключениям;
- изменение биологических ритмов.

Так как работа разработчика программных систем связана только с работой на ПК, она является монотонной. Такая работа требует непрерывной концентрации внимания на протяжении длительного времени и является однообразной.

Для предупреждения и снижения уровня монотонности труда можно проводить следующие мероприятия:

- введение рационального режима труда и отдыха;
- правильная организация ритма и темпа работы, чередование операций;
- кратковременные частые перерывы в работе;
- организация физических упражнений в течение регламентированных перерывов;
- чередование работы со сменой положений стоя – сидя;
- организация специальных помещений психологической разгрузки и отдыха.

#### **4.2.6. Нарушение правил электробезопасности**

Источниками электрической опасности являются электрические сети, электрифицированное оборудование и инструмент, вычислительная и организационная техника, работающая на электричестве. В связи с большим количеством электрических приборов и вычислительных машин на представленном рабочем месте, электробезопасность является важной составляющей производственной безопасности.

При работе с электрифицированными приборами необходимо соблюдать технику безопасности, которая представляет собой систему мероприятий и технических средств, направленных на предотвращение воздействий на работников вредных и опасных факторов.

В рабочем помещении может происходить накопление статического электричества, его разряды не представляют опасности для работников, но могут привести к проблемам с вычислительными машинами. Чтобы снизить величины зарядов статического электричества покрытие полов в помещении выполняется из однослойного линолеума.

Опасность поражения электрическим током являются серьезной потенциальной проблемой, так как человеческие органы чувств не могут обнаружить наличие электрического напряжения на расстоянии.

Риск поражения электрическим током возрастает при следующих условиях: повышенная влажность, когда относительная влажность воздуха выше 75 %; высокая температура воздуха и поверхностей, более 35 °С; наличие токопроводящей пыли и токопроводящих полов; возможность одновременного соприкосновения к заземленным металлическим элементам и металлическим корпусом электрооборудования.

Работа может проводиться исключительно в помещениях, исключающих повышенную опасность, однако, есть риск возникновения опасности другого рода:

- при прикосновении к токоведущим частям (во время ремонта ПК);
- при прикосновении к нетоковедущим частям, которые оказались под напряжением (при нарушении изоляции);
- при соприкосновении с полом или стенами, оказавшимися под напряжением (при нарушении электрической сети);
- при коротком замыкании в высоковольтных блоках.

Представленное место работы не относится к помещениям повышенной опасности электропоражения. В помещении используются



приборы, потребляющие напряжение 220 В переменного тока с частотой 50 Гц. Для предотвращения возникновения опасных ситуаций обязательны следующие меры предосторожности:

- перед началом рабочей смены необходимо убедиться, что выключатели и розетки закреплены и не имеют оголенных токоведущих частей;
- при обнаружении неисправности оборудования и приборов, необходимо сообщить ответственному лицу, не делая никаких самостоятельных исправлений;
- запрещено загромождать рабочее место лишними предметами.

### **4.3. Экологическая безопасность**

Человечество всегда оказывает влияние на окружающую среду, на текущий момент это влияние имеет катастрофические масштабы, так как воздействие человека и его потребление ресурсов перешло на тот уровень, когда планета не способна воспроизвести столько ресурсов, сколько потребляет человечество. Такое отношение к Земле привело к дефициту экосистем и экологическому кризису.

Наука не стоит на месте, развивается и представляет новые способы предотвращения и исправления экологических проблем. Защита окружающей среды требует полного перехода к безотходным и малоотходным производствам и технологиям, к правильной утилизации отходов. Для этого необходим комплекс технологических и организационных мероприятий, основанных на использовании современных научных достижений.

Утилизация компьютерной и организационной техники ограничено законодательно, так как в производстве такой техники используется большое количество материалов, способных нанести большой вред окружающей среде. Утилизация компьютерного оборудования происходит через обязательное извлечение компонент, их сортировку и последующую отправку для повторного использования. Такая утилизация обязательно

производится на оборудованных полигонах с привлечением квалифицированного персонала.

Люминесцентные лампы являются одним из самых распространенным источником загрязнения ртутью, так как при неправильной утилизации ламп ртуть, находящаяся в них, попадает в землю, что очень опасно для планеты и для жизни людей. Правильной утилизацией люминесцентных ламп является передача лицензированным компаниям для переработки и вторичного использования сырья в качестве материала для производств.

Утилизация мусорных отходов, таких как бумажная макулатура, отходы от канцелярских принадлежностей, отходы от продуктов питания, личной гигиены, производится через сбор, обязательную сортировку и утилизацию. Отходы, которые можно использовать повторно, например, макулатуру, после сортировки отправляют на переработку через компании, занимающиеся сбором таких отходов.

Используя такую систему утилизации отходов работы можно реально уменьшить свое воздействие на окружающую среду, а также на собственное здоровье, так как качественная утилизация отходов исключает отравление опасными веществами и попадание тяжелых металлов в организмы.

#### **4.4. Безопасность в чрезвычайных ситуациях**

Чрезвычайной ситуацией (ЧС) называется обстановка на определенной территории, сложившаяся в результате аварии, опасного природного явления, катастрофы или другого бедствия, которая может повлечь за собой человеческие жертвы, ущерб здоровью людей или окружающей среде, значительные материальные потери и нарушение условий жизнедеятельности людей. ЧС для представленного рабочего помещения является пожар. Данная ЧС может произойти в случае не соблюдения мер пожаробезопасности, нарушения техники использования электрических приборов и ПК, нарушениях разводки электрических сетей и ряда других причин.

Рабочее помещение, представленное для выполнения ВКР, согласно ГОСТ 12.1.030-81 ССБТ [11], можно отнести к категории В (пожароопасное).

В качестве возможных причин возникновения пожара можно указать следующие причины:

- короткое замыкание;
- опасная перегрузка сетей, которая ведет за собой сильный нагрев токоведущих частей и загорание изоляции;
- пуск оборудования после некорректного и неквалифицированного ремонта.

Для предотвращения ЧС необходимо соблюдать правила пожарной безопасности, чтобы обеспечить состояние защищенности работников и имущества от пожара.

Для защиты от коротких замыканий и перегрузок необходимо правильно выбирать, устанавливать и использовать электрические сети и средства автоматизации.

Для предупреждения возникновения пожаров необходимо исключить образование горючей среды, следить за применением при строительстве и отделке зданий негорючих или трудно сгораемых материалов.

Необходимо проводить следующие пожарно-профилактические мероприятия:

- организационные мероприятия, касающиеся технического процесса с учетом пожарной безопасности объекта (инструктаж персонала, обучение правилам техники безопасности, издание инструкций, плакатов, планов эвакуации);
- эксплуатационные мероприятия, рассматривающие эксплуатацию используемого оборудования (соблюдение эксплуатационных норм оборудования, обеспечение свободного подхода к оборудованию, поддержание исправности изоляции проводников);
- технические и конструктивные мероприятия, связанные с правильным размещением и монтажом электрооборудования и отопительных

приборов (соблюдение противопожарных мероприятий при устройстве электропроводок, оборудования, систем отопления, вентиляции и освещения).

Для повышения устойчивости рабочего помещения к ЧС необходимо произвести установку систем противопожарной сигнализации, реагирующих на дым и другие продукты горения, установку огнетушителей. Также, два раза в год проводить учебные тревоги для отработки действий при пожаре.

В представленном рабочем помещении при входе представлен план эвакуации, установлена система противопожарной сигнализации. Помещение оборудовано углекислотными огнетушителями типа ОУ-2 в количестве 2 штук на одну рабочую зону. В зоне досягаемости работниками находится электрощит, с помощью которого можно полностью обесточить рабочее помещение.

В случае возникновения возгорания, необходимо вызвать пожарную службу по телефону 101 и сообщить место возникновения ЧС, предпринять меры по эвакуации работников в соответствии с планом эвакуации. При отсутствии прямых угроз здоровью и жизни произвести попытку тушения возникшего возгорания имеющимися углекислотными огнетушителями. В случае потери контроля над пожаром, необходимо эвакуироваться вслед за сотрудниками по плану эвакуации и ждать приезда специалистов пожарной службы.

#### **4.5 Выводы**

В заключение раздела можно сделать выводы о том, что грубых нарушений по организации работы при выполнении ВКР не обнаружено, все требования и нормы безопасности соблюдены. Организационные вопросы по обеспечению необходимых рабочих условий имеют под собой законодательное подтверждение и не нарушают законодательный регламент.

## Заключение

При выполнении выпускной квалификационной работы была реализована программная библиотека под платформу Windows и Linux, а также для платформы рендеринга OpenGL.

В ходе выполнения работы было проведено исследование предметной области, которое включает в себя описание предметной области, моделирование бизнес-процессов в нотации IDEF0, обзор и анализ существующих аналогов графических визуализаторов.

В ходе реализации программной библиотеки был создан необходимый функционал для создания трехмерных и двухмерных объектов, манипуляции над ними и их визуализации. Помимо этого был реализован функционал для работы с текстурами, текстом, созданием анимации объектов.

В результате выполнения выпускной квалификационной работы были закреплены и углублены теоретические знания, получены практические навыки проектирования и моделирования программной библиотеки, программирования на языках C++, GLSL. Полученные навыки удовлетворяют описанным ранее планируемым результатам обучения по профилю специальности «Программная инженерия».

## Список достижений

### Статьи:

1. Черкашин А.Ю., Ключов П.Д., Ворожейкин В.А., Марчуков А.В. Национальный стандарт передачи данных для нефтегазовой промышленности России [Электронный ресурс] // Конкурс научно-технических печатных работ молодых ученых и специалистов: сборник трудов 71-ой Международной молодежной научной конференции «Нефть и газ», Москва, 2017 г – С. 405-414. – Режим доступа: [http://neftegaz.gubkin.ru/site/assets/files/1221/sbornik\\_trudov.pdf](http://neftegaz.gubkin.ru/site/assets/files/1221/sbornik_trudov.pdf)
2. Ключов П.Д., Ворожейкин В.А., Соколова В.В. Разработка приложения для судейства спортивных соревнований // Молодежь и современные информационные технологии // XVI Международная конференция студентов, аспирантов и молодых ученых, Томск, 3-7 Декабря 2018. – Томск: ТПУ, 2018 – С. 399-400.

### Дипломы:

1. Диплом III степени на XVI Международная конференция студентов, аспирантов и молодых ученых «Разработка приложения для судейства спортивных соревнований» (2018 г.).

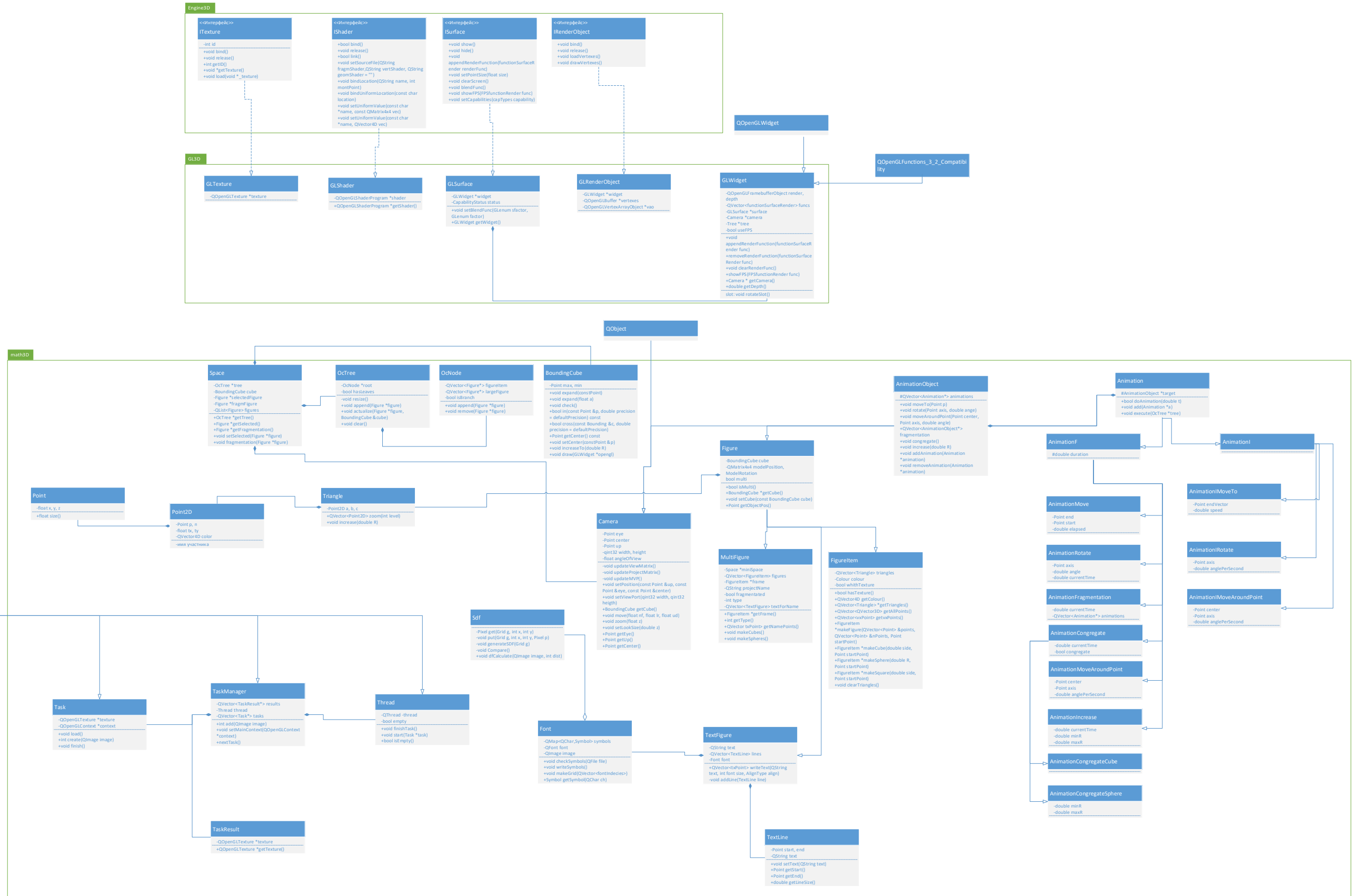
## Список источников

- [1] «Графический движок» 2019. [Электронный ресурс]. URL: <https://dic.academic.ru/dic.nsf/ruwiki/145202>. (Дата обращения: 08 04 2019).
- [2] «learnopengl. Урок 1.1 — OpenGL» 2019. [Электронный ресурс]. URL: <https://habr.com/ru/post/310790/>. (Дата обращения: 08 04 2019).
- [3] «Основы Signed Distance Field в 2D» 2019. [Электронный ресурс]. URL: <https://habr.com/ru/post/438316/>. (Дата обращения: 08 04 2019).
- [4] «Что такое шейдеры? Просто о сложном для начинающих» 2019. [Электронный ресурс]. URL: <https://coremission.net/gamedev/chto-takoe-sheidery/>. (Дата обращения: 08 04 2019).
- [5] «Введение в октодеревья» 2019. [Электронный ресурс]. URL: <https://habr.com/ru/post/334990/>. (Дата обращения: 07 04 2019).
- [6] «ГОСТ 12.1.006-84 ССБТ. Электромагнитные поля радиочастот» [Электронный ресурс]. URL: <http://docs.cntd.ru/document/5200272> (дата обращения 12.05.2019)
- [7] «СанПиН 2.2.4.548–96. Гигиенические требования к микроклимату производственных помещений» [Электронный ресурс]. URL: <http://docs.cntd.ru/document/901704046> (дата обращения 12.05.2019)
- [8] «СП 52.13330.2016. Естественное и искусственное освещение» [Электронный ресурс]. URL: <http://docs.cntd.ru/document/456054197> (дата обращения 12.05.2019)
- [9] «ГОСТ 12.1.003-2014 ССБТ. Шум. Общие требования безопасности» [Электронный ресурс]. URL: <http://docs.cntd.ru/document/5200291> (дата обращения 12.05.2019)
- [10] «Р 2.2.2006–05. Гигиена труда. Руководство по гигиенической оценке факторов рабочей среды и трудового процесса. Критерии и классификация условий труда» [Электронный ресурс]. URL: <http://docs.cntd.ru/document/1200040973> (дата обращения 12.05.2019)

[11] «ГОСТ 12.1.030-81 ССБТ. Электробезопасность. Защитное заземление. Зануление» [Электронный ресурс]. URL: <http://docs.cntd.ru/document/5200289> (дата обращения 12.05.2019)



# Приложение А. Диаграмма классов.

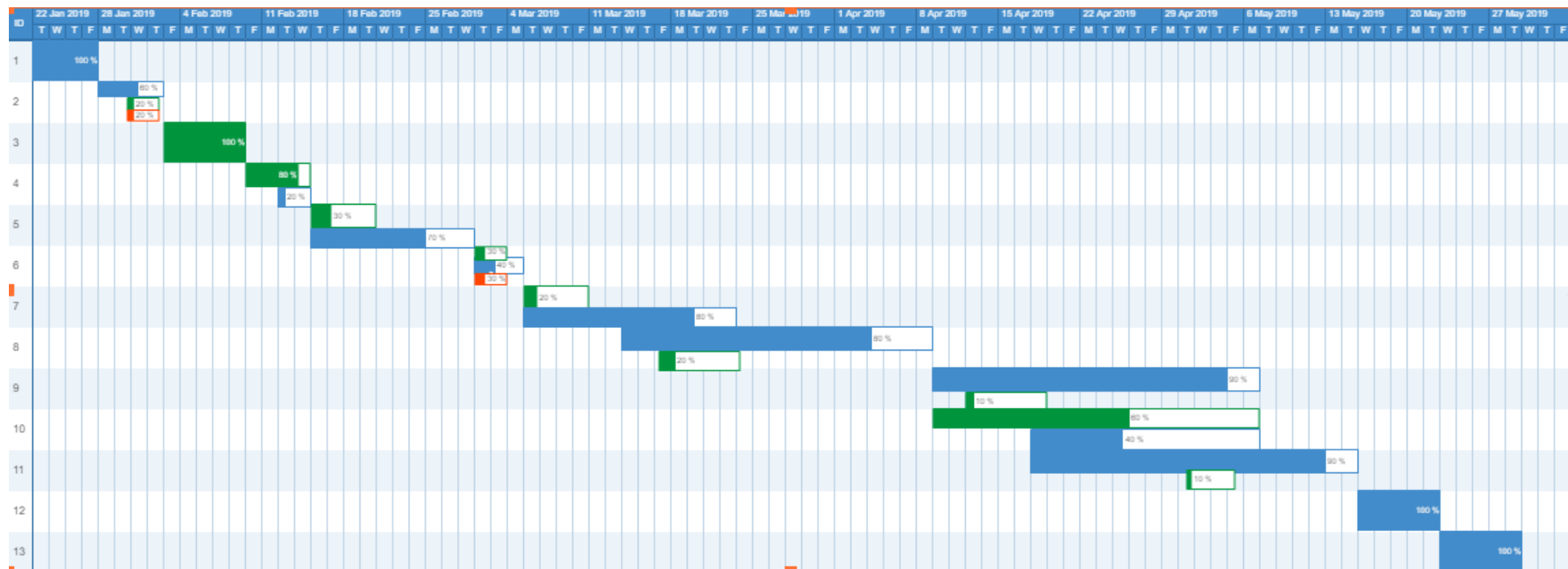


## Приложение Б. Таблицы и диаграммы для раздела «Финансовый менеджмент»

Таблица 1. Сводная таблица SWOT-анализа

|  |  |  |
|--|--|--|
|  | <p><b>Сильные стороны проекта:</b></p> <ul style="list-style-type: none"> <li>• Реализация более оптимальных и ресурсоемких алгоритмов и решений по сравнению с аналогами;</li> <li>• Использование программной библиотеки внутри организации для реализации будущих проектов</li> </ul> | <p><b>Слабые стороны проекта:</b></p> <ul style="list-style-type: none"> <li>• Дополнительные временные и финансовые затраты на реализацию, тестирование и сопровождение программной библиотеки</li> </ul>   |
| <p><b>Возможности:</b></p> <ul style="list-style-type: none"> <li>• Экономическая и функциональная независимость компании от аналоговых продуктов конкурентов</li> </ul>                         | <p>Реализация и будущая модификация оптимизированной кроссплатформенной библиотеки позволит создавать собственные программные обеспечения без функциональных ограничений и финансовых затрат на ее использование.</p>  | <p>Реализация собственной программной библиотеки для дальнейшего создания будущих ПО с целью коммерческой выгоды окупится раньше, чем использование платных аналоговых продуктов.</p>  |
| <p><b>Угрозы:</b></p> <ul style="list-style-type: none"> <li>• Наличие более оптимизированных и функциональных библиотек у конкурентов, позволяющие реализовать более качественное ПО</li> </ul> | <p>Анализ аналоговых библиотек позволит реализовать собственное ПО, не уступающее и конкурирующее по оптимизации и качеству.</p>   | <p>На основе угроз и слабых сторон можно сделать вывод, что существует риск реализации продукта, в котором будут не соответствовать ее качество и оптимизация будущей программной библиотеки с затратами на реализацию. Во избежание данного риска необходимо анализировать конкурентов и иметь в команде разработчиков опытных руководителей команды.</p> |

## Диаграмма Ганта для бизнес процесса «Реализация программной библиотеки»



Примечание:



## Приложение В. Программный код заголовочных файлов, использующиеся в библиотеке.

### Заголовочный файл класса OcTree (октодеревя)

```
class OcNode
{
    BoundingBox boundingCube;

    OcNode *LNU = nullptr;

    OcNode *LND = nullptr;

    OcNode *LFU = nullptr;

    OcNode *LFD = nullptr;

    OcNode *RNU = nullptr;

    OcNode *RND = nullptr;

    OcNode *RFU = nullptr;

    OcNode *RFD = nullptr;

    bool isBranch = false;

    QVector<Figure*> items;

    QVector<Figure*> largeItems;

    const quint32 maxNodeSize = 3;

public:

    OcNode(const BoundingBox &cube);

    ~OcNode()
    {
        delete LNU;

        delete LND;

        delete LFU;

        delete LFD;

        delete RNU;

        delete RND;

        delete RFU;

        delete RFD;
    }
}
```

```

    bool append(Figure *figure, double precision = defaultPrecision);

    bool remove(Figure *figure, double precision = defaultPrecision);

    void clear() { items.clear(); largeItems.clear(); }

    bool get(const BoundingBox &cube, QVector<Figure *> &_items, double
precision = defaultPrecision) const;

    OcNode *lookup(Figure *figure, double precision = defaultPrecision);

    void debug(quint32 level = 0, const QString &tab="", const QString
&postFix = "") const;

    const BoundingBox &getBoundingBox() const { return boundingCube; }

    void draw(GLWidget *widget);

    QVector<Figure*> getAllFigures();

protected:

    bool split(double precision = defaultPrecision);

};

class OcTree

{

    OcNode *root;

    bool hasLeaves;

    void resize();

    bool checkLarge(Figure figure);

public:

    OcTree();

    OcTree(const BoundingBox &cube);

    ~OcTree() { delete root; }

    bool append(Figure *figure, double precision = defaultPrecision) { return
this->root->append(figure, precision); }

    bool remove(Figure *figure, double precision = defaultPrecision) { return
this->root->remove(figure, precision); }

    bool get(const BoundingBox &cube, QVector<Figure*> &items, double
precision = defaultPrecision) const { return root->get(cube, items, precision); }

    bool actualize(Figure *figure, const BoundingBox &newCube, double
precision = defaultPrecision);

    void clear();

```

```

void draw(GLWidget *widget) { root->draw(widget); }

QVector<Figure*> getAllFigures() { return root->getAllFigures(); }

};

```

## Заголовочный файл класса Figure (родительский класс фигур)

```

class Figure: public AnimationObject
{
protected:
    QMatrix4x4 modelPosition, modelRotation;

    BoundingBox cube;

    bool _multi = false;

public:
    Figure(): AnimationObject () { }

    bool isMulty() { return _multi; }

    BoundingBox *getBoundingBox() { return &cube; }

    void setBoundingBox(const BoundingBox &_cube) { cube = _cube; }

    QMatrix4x4 getPosModel() { return modelPosition; }

    QMatrix4x4 getRotModel() { return modelRotation; }

    Point getObjectPosition() const override { return cube.getCenter(); }

protected:
    virtual QVector<AnimationObject *> fragmentation() override {};

    virtual void congregate() override {};

    virtual void removeMinispace() override {};

    virtual void increase(double R) override {};

    void rotate(const Point &axis, double angle) override {
modelRotation.rotate(angle, axis.toVector3D()); }

    void moveTo(const Point &p) override;

    void moveAroundPoint(const Point &O, const Point &axis, double angle)
override;

    void moveAroundObject(const Point &O, const Point &axis, double angle,
double _Rorbit) override;

    void actualize(OcTree *tree) override;

};

```

## Заголовочный файл класса Camera (пользовательская камера):

```
class Camera : public QObject, public AnimationObject
{
    qint32 width, height;

    Point up, eye, center;

    float angleOfView;

    struct
    {
        mutable QMatrix4x4 view;

        mutable QMatrix4x4 projection;

        mutable QMatrix4x4 viewport;

        mutable QMatrix4x4 mvp;

        mutable QMatrix4x4 pmvp;

        mutable QMatrix4x4 imvp;

    } matrixes;

    mutable struct
    {
        bool view;

        bool projection;

        bool mvp;

    } flags;

    void updateViewMatrix() const;

    void updateProjectionMatrix() const;

    void updateMVP() const;

public:

    Camera();

    Camera(const Point &_up, const Point &_eye, const Point &_center, const
double &_angleOfView);

    virtual ~Camera() { animationList.clear(); }

    void setPosition(const Point &_up, const Point &_eye, const Point
&_center);
```

```

void setViewPort(qint32 _width, qint32 _height);

BoundingCube getBoundingCube() const;

const QMatrix4x4 &getView() const { if(flags.view) updateViewMatrix();
return this->matrixes.view; }

const QMatrix4x4 &getProjection() const { if(flags.projection)
updateProjectionMatrix(); return this->matrixes.projection; }

const QMatrix4x4 &getMVP() const { if(flags.mvp) updateMVP(); return this-
>matrixes.mvp; }

const QMatrix4x4 &getPMVP() const { if(flags.mvp) updateMVP(); return
this->matrixes.pmvp; }

const QMatrix4x4 &getIMVP() const { if(flags.mvp) updateMVP(); return
this->matrixes.imvp; }

QVector3D project(const QVector3D &p) const { return QVector4D(getPMVP() *
QVector4D(p.x(), p.y(), p.z(), 1.0f)).toVector3DAffine(); }

QVector3D unproject(const QPoint &p, float depth) const { return
QVector4D(getIMVP() * QVector4D(p.x(), p.y(), 2.0f*depth - 1.0f,
1.0f)).toVector3DAffine(); }

void move(float nf, float lr, float ud);

void zoom(float z);

void zoomToObject(float z, Point O);

void rotateUpDnAroundObject(float a, Point O);

void rotateUpDnAroundCenter(float a);

void rotateLRAroundObject(float a, Point O);

void rotateLRAroundCenter(float a);

void rotateUpDn(float a);

void rotateAroundUp(float a);

void rotateAroundLook(float a);

void setLookSize(double sz) { Point look = getLook().normalize() * sz;
center = eye + look; flags.view = true; flags.mvp = true; }

Point getLook() const { return (center - eye); }

Point getUp() const { return (up - eye); }

const Point &getEye() const { return eye; }

const Point &getCenter() const { return center; }

float getAngleOfView(){ return angleOfView; }

```



```
void moveTo(const Point &p) override;

void rotate(const Point &axis, double angle) override;

void moveAroundPoint(const Point &O, const Point &axis, double angle)
override;

void moveAroundObject(const Point &O, const Point &axis, double angle,
double _ROrbit) override {}

Point getObjectPosition() const override { return getCenter(); }

void actualize(Octree *) override {}

QVector<AnimationObject*> fragmentation() override {}

void congregate() override {};

void increase(double R) override {}

void removeMinispace() override {}

};
```