|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **Компьютерные системы и сети (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.01 Информатика и вычислительная техника**

БАКАЛАВРСКАЯ ПРОГРАММА **09.03.01/03 Вычислительные машины, комплексы,**

**системы и сети**

**ОТЧЕТ ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ**

|  |  |
| --- | --- |
| Тип практики | Преддипломная практика |

|  |  |
| --- | --- |
| Название  предприятия | «НУК ИУ МГТУ им. Н.Э.Баумана |

|  |  |  |  |
| --- | --- | --- | --- |
| Студент ИУ6-83Б |  |  | А.А. Бушев |
|  |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |
| Руководитель практики |  |  | М.В.Фетисов |
|  |  | (Подпись, дата) | (И.О. Фамилия) |

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*2023 г.*

**Министерство науки и высшего образования Российской Федерации**

**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Московский государственный технический университет имени Н.Э. Баумана**

**(национальный исследовательский университет)»**

**(МГТУ им. Н.Э. Баумана)**

**ЗАДАНИЕ**

**на производственную практику**

по теме \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_адаптивный сервер моделирования\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Студент группы ИУ6-83Б

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Бушев Антон Алексеевич\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия, имя, отчество)

Направление подготовки 09.03.01 Информатика и вычислительная техника

Бакалаврская программа 09.03.01/03 Вычислительные машины, комплексы, системы и сети

Тип практики Преддипломная практика\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Название предприятия НУК ИУ МГТУ им. Н.Э. Баумана

***Техническое задание*** \_\_реализовать\_\_\_сервер\_\_\_моделирования,\_\_\_реализовать\_\_\_плагин\_\_\_для\_

\_\_подключения\_\_\_сервера\_\_\_моделирования\_\_\_к\_\_\_интегрированной\_\_\_среде\_\_\_разработке\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***Оформление отчета по практике:***

Отчет на 15-25 листах формата А4.

Перечень графического (иллюстративного) материала (чертежи, плакаты, слайды и т.п.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_нет\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата выдачи задания « 07 » февраля 2023 г.

|  |  |  |  |
| --- | --- | --- | --- |
| **Руководитель практики** |  |  | М.В. Фетисов |
|  |  | (Подпись, дата) | (И.О. Фамилия) |
| **Студент** |  |  | А.А. Бушев |
|  |  | (Подпись, дата) | (И.О. Фамилия) |

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 5](#_Toc22058)

[1 Выбор технологии и языка программирования 6](#_Toc11246)

[2 Выбор подхода разработки 6](#_Toc9620)

[3 Разработка схемы структурной информационной системы 6](#_Toc23423)

[4 Разработка диаграмм последовательностей 8](#_Toc22040)

[5 Разработка диаграмм классов предметной области 11](#_Toc17007)

[6 Разработка схемы алгоритмов модуля сцены 12](#_Toc5497)

[ЗАКЛЮЧЕНИЕ 17](#_Toc3806)

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 18](#_Toc23316)

**ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ**

ПП – программный продукт

ООП – объектно-ориентированное программирование

scriptC0 – первый скриптовый язык SIMODO

**ВВЕДЕНИЕ**

Целью преддипломной практики является описание структуры разрабатываемого продукта и реализация этапов проектирования приложения ВКРБ. В процессе работы будет выбрана технология программирования, разработан алгоритм моделирования, разработана диаграмма последовательностей и диаграмма классов предметной области.

**1 Выбор технологии и языка программирования**

В качестве технологии программирования было выбрано объектно-ориентированное программирование. ООП является основой всех современных приложений и имеет удобное и практическое применение. При использовании этого метода вся программа разбивается на объекты, с каждым из которых работа происходит по отдельности, что позволяет в будущем расширять программный продукт путем добавления новых объектов.

Для написания настольных приложений существует множество языков программирования. Наиболее популярные[1] – это С#, C++ и Python, которые являются кроссплатформенными языками.

Для разрабатываемого приложения был выбран язык C++, так как он используется в разработке адаптивной среды разработки SIMODO[2].

**2 Выбор подхода разработки**

В качестве жизненного цикла разработки была выбрана спиральная модель. Этот метод позволяет в конце каждого цикла иметь работающий продукт, который можно продемонстрировать[3]. Это дает возможность своевременно оценить и протестировать продукт, чтобы сразу вносить какие-либо правки и исправлять ошибки, не дожидаясь окончания разработки. Нахождение багов на каждом этапе позволяет избежать «волнового» исправления ошибок. А также этот метод позволяет детальнее подойти к каждому этапу разработки по отдельности.

Также при разработке было решено использовать нисходящий подход[4], реализуя сначала модули верхнего уровня (интерфейс пользователя), а после переходя к модулям нижнего уровням (логика работы программы).

**3 Разработка схемы структурной информационной системы**

Структурная схема программного продукта показывает разделение программы на её главные составляющие. На основе анализа технического задания, в разрабатываемом приложении, которое выполняет имитационное моделирование, выявлено две подсистемы:

* подсистема интеграции с адаптивной средой разработки: подсистема интеграции с адаптивной средой разработки преобразует входные данные в понятный для сервера моделирования вид и преобразует исходящие от сервера моделирования данные в приемлемый для среды разработки вид; включает в себя две подсистемы:  
  1) в подсистеме ввода вывода происходит работа с процессом, в котором работает сервер моделирования;  
  2) в подсистеме обработки сообщений происходит обработка сообщений, приходящих от сервера;
* подсистема имитационного моделирования включает в себя три подсистемы:  
  1) в подсистеме ввода вывода происходит получение и отправка данных в рамках протокола сервера моделирования;  
  2) в подсистеме обработки сообщений происходит обработка сообщений, приходящих от редактора;  
  3) подсистема интерпретации отвечает за процесс имитационного моделирования в рамках интерпретатора scriptC0; включает в себя три подсистемы:  
   а) интерпретатор отвечает за исполнение скриптов на языке scriptC0  
   б) менеджер модулей обеспечивает интерпретатор исходными текстами модулей, запрашиваемых в процессе интерпретации  
   в) модуль сцены является модулем scriptC0; отвечает за расчёт моделей в виде систем дифференциальных уравнений с помощью численных методов.

На основе выявленных подсистем была составлена структурная схема ПП, показанная на рисунке 1.

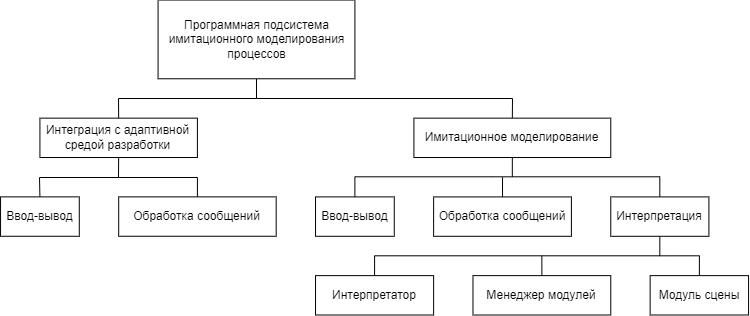


Рисунок 1 – Структурная схема информационной системы

**4 Разработка диаграмм последовательностей**

Сервер имитационного моделирования принадлежит к третьему уровню в архитектуре интегрированной среды разработки SIMODO. Сервер выполняется в отдельном процессе, с которым взаимодействует плагин расширяемого редактора для передачи исходных текстов моделей, скриптов запуска моделирования или других программных модулей. В свою очередь, сервер отправляет информацию о состоянии моделей и ходе моделирования. Серверу может понадобится дополнительная информация (например, модель использует внешние модули), поэтому он может запросить дополнительные ресурсы у плагина (например, исходные тексты внешних модулей). Запуск и остановка сервера имитационного моделирования SIMODO, запуск, временная и полная остановка моделирования должны программно контролироваться плагином расширяемого редактора.

Таким образом, можно выделить следующие сообщения, передаваемые между плагином расширяемого редактора и процессом сервера имитационного моделирования SIMODO:

* Инициализация с грамматиками языков программирования в качестве аргументов: инициализация сервера имитационного моделирования SIMODO с предоставление исходных текстов грамматик языков программирования, используемых во время имитационного моделирования.
* Запуск моделирования с основным сценарием в качестве аргумента: запуск интерпретации исходного текста основного сценария моделирования.
* Запрос внешнего ресурса по его адресу: сигнал адаптивной среде разработке о необходимости в исходном тексте модуля, который указан в основном сценарии моделирования или в уже полученных модулях.
* Отправка исходного текста ресурса: передача серверу имитационного моделирования исходного текста модуля, который был запрошен по определённому адресу.
* Установка пары ключ-значение: передача информации серверу имитационного моделирования в процессе имитационного моделирования для управления ходом имитационного моделирования.
* Сообщение информации: передача информации адаптивной среде разработки о ходе имитационного моделирования.
* Приостановка моделирования: сигнал серверу имитационного моделирования об обратимом прерывании хода имитационного моделирования.
* Возобновление моделирования: сигнал серверу имитационного моделирования о возобновлении хода имитационного моделирования.
* Остановка: сигнал серверу имитационного моделирования о необратимом прерывании хода имитационного моделирования.
* Остановка сервера: сигнал серверу имитационного моделирования о прерывании работы сервера имитационного моделирования; требуется ожидание момента завершения процесса, в котором исполняется сервер имитационного моделирования.

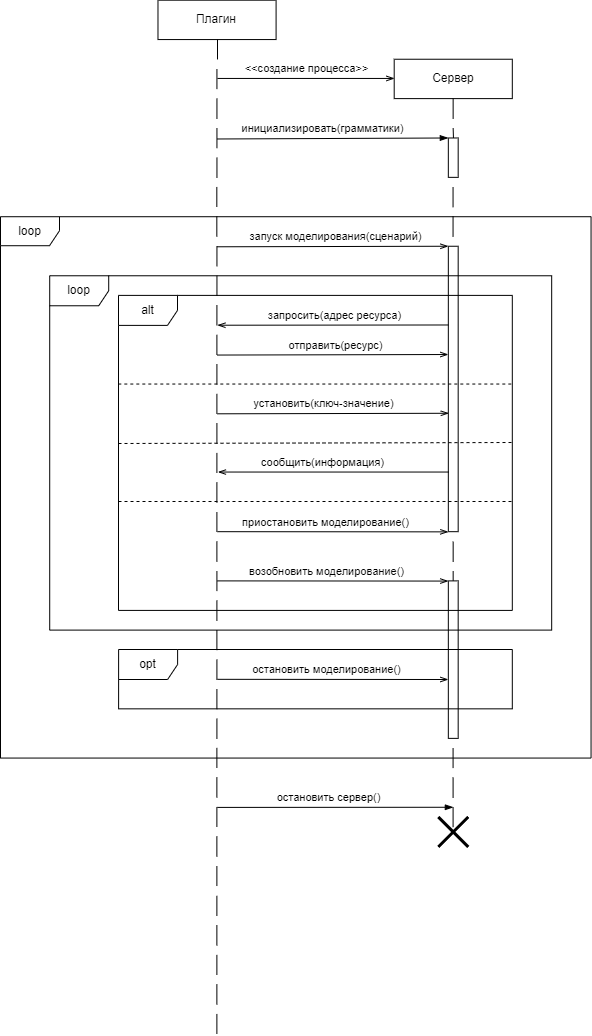


Рисунок 2 – Диаграммы последовательностей

На рисунке 2 изображена диаграмма последовательности[5] сообщений между плагином и сервером имитационного моделирования SIMODO при нормальном ходе моделирования, т.е. без возникновения ошибок при обмене сообщениями.

Основная концепция взаимодействия между процессом плагина и процессом сервера моделирования заключается в том, что после запуска процесса, на котором работает сервер моделирования, плагин инициирует запуск, управление и остановку моделирования, в свою очередь предоставляя серверу необходимые ресурсы.

В первую очередь плагин запускает процесс сервера моделирования, после чего плагин должен отправить сообщение инициализации серверу. О готовности сервера плагин может узнать отправив запрос на получение информации о состоянии сервера. В ответ на запрос сервер отправит запрашиваемую информацию стандартным сообщением данных.

Моделирование запускается на инициализированном сервере моделирования и передаётся сценарий моделирования. После этого нормальным считается следующее поведение:

* Запрос сервером моделирования дополнительных ресурсов и отправка соответствующих ресурсов плагином.
* Отправка плагину данных о ходе моделирования.
* Отправка серверу моделирования команд управления, которые обрабатываются в соответствии со сценарием моделирования.
* Временная приостановка хода моделирования и возобновления хода моделирования. Завершение хода моделирования возможно в соответствии с логикой сценария моделирования (не приведено на рисунке 2) или при получении сообщения остановки моделирования.

Сервер моделирования позволяет проводить моделирование неограниченное число раз в общем случае с разными сценариями моделирования.

**5 Разработка диаграмм классов предметной области**

После окончания разработки ПО была разработана диаграмма классов предметной области, чтобы представить созданные классы с их полями и методами. Диаграмма классов[6] сервера имитационного моделирования показана на рисунке 3. Диаграмма классов модуля сцены показана на рисунке 4.

Основные классы сервера имитационного моделирования:

* Сервер — класс, организующий конфигурацию сервера имитационного моделирования.
* СервисВводаВывода — интерфейс, обеспечивающий получение и отправку сообщений через абстрактный канал.
* ОбработчикСобытий — интерфейс, позволяющий наращивать функционал сервера созданием нового объекта.
* Интерпретатор — интерфейс, обеспечивающий ход имитационного моделирования.

Основные классы модуля сцены:

* Сцена — класс, отвечающий за управление имитационным моделированием и обеспечением режима реального времени, если он активирован.
* Актор — интерфейс участника сцены; позволяет потенциально использовать любые модели.
* Модель — класс, отвечающий за представление системы дифференциальных уравнений в понятном для сцене виде.
* Композиция — класс, отвечающий за объединение моделей в единицу, которую можно трактовать как отдельную модель; позволяет выстраивать иерархическую структуру сложных моделей.

**6 Разработка схемы алгоритмов модуля сцены**

Одной из основных подсистем приложения является модуль сцены. Алгоритм работы модуля сцены необходимо разработать так, чтобы обеспечить пересчёт всех акторов, и если активирован режим реального времени, выполнить пересчёт акторов с адекватной реальному времени задержкой.

Схемы алгоритмов[5] модуля сцены представлены на рисунках 5–6.

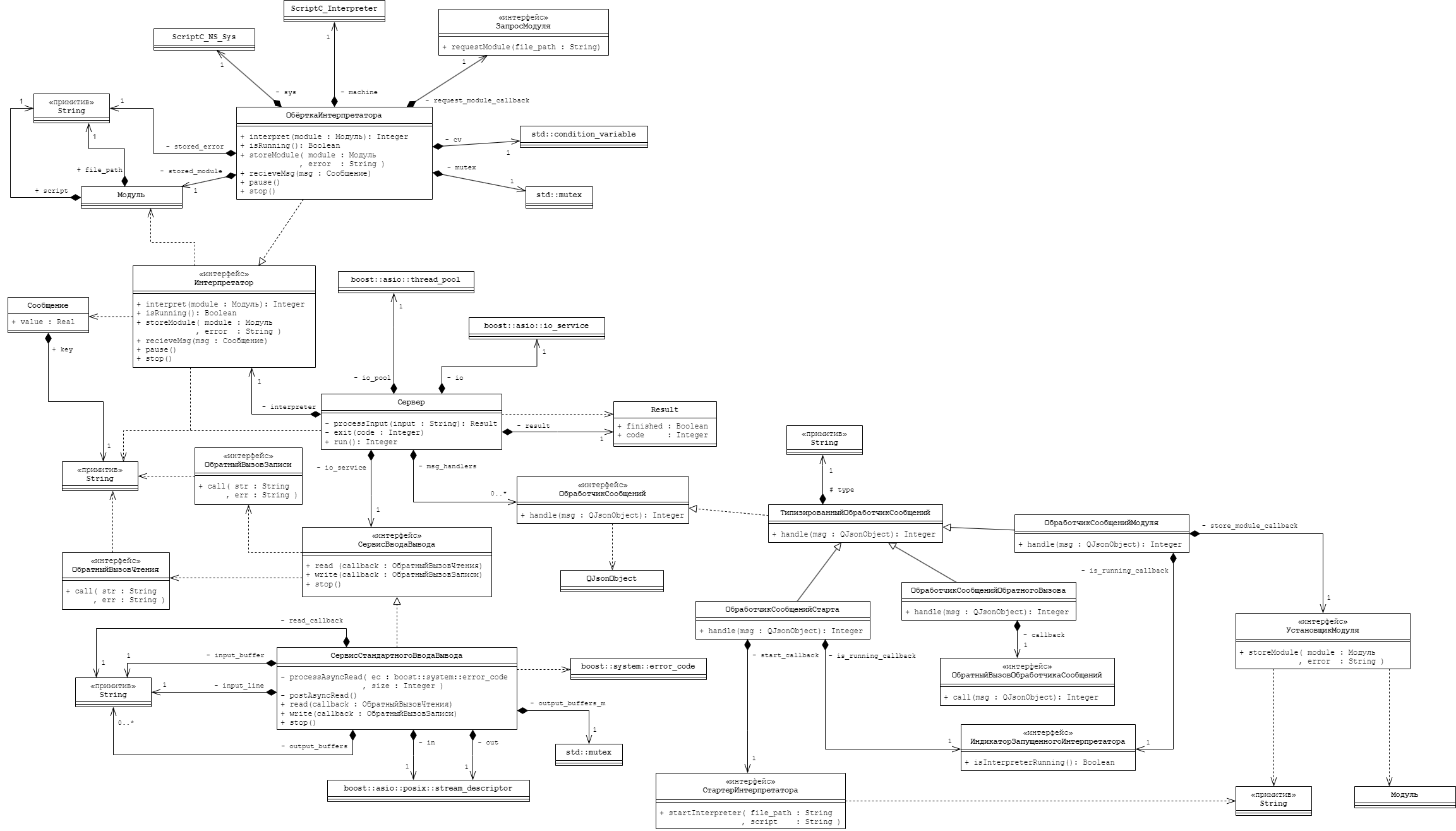


Рисунок 3 – Диаграммы классов сервера имитационного моделирования

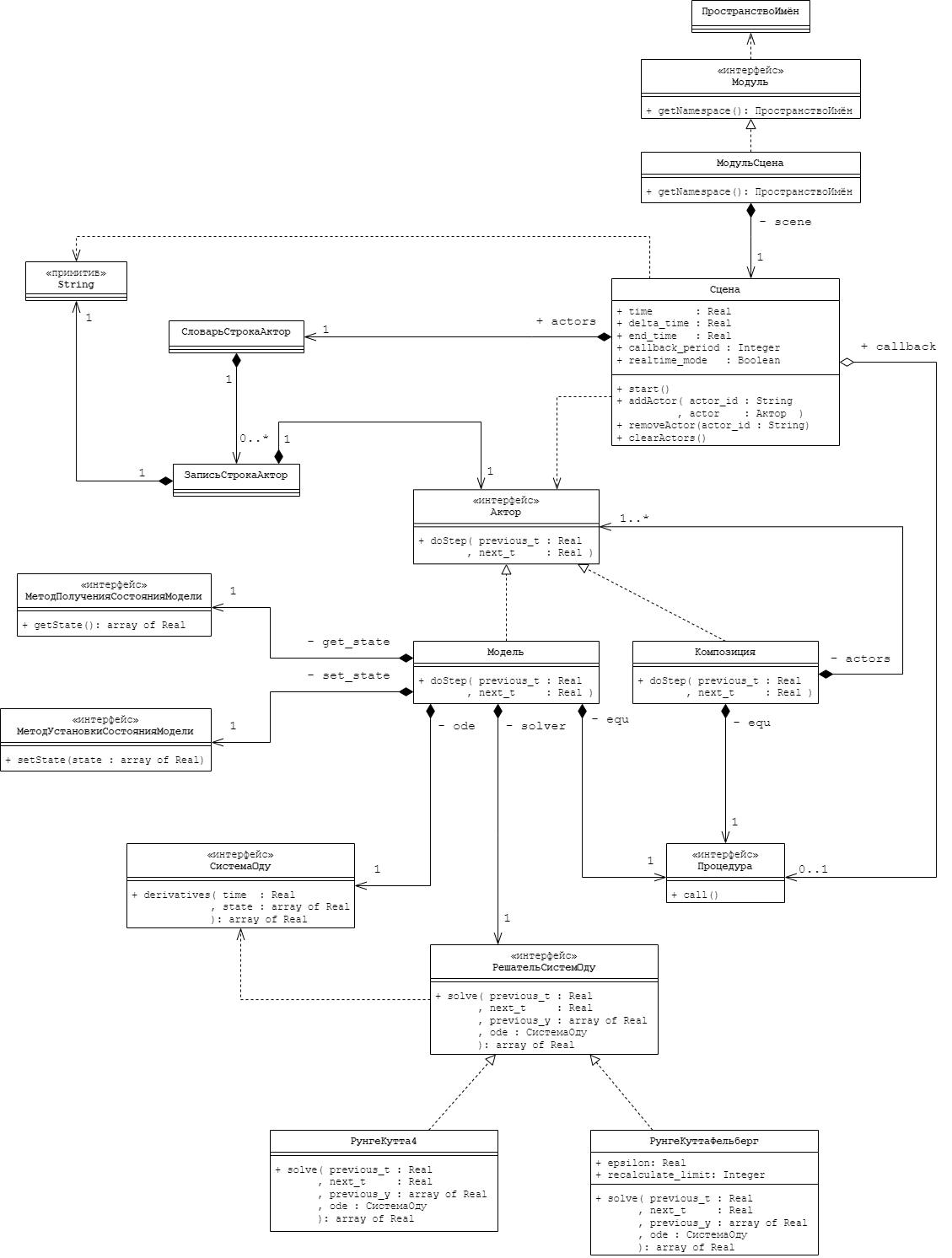


Рисунок 4 – Диаграммы классов модуля сцены

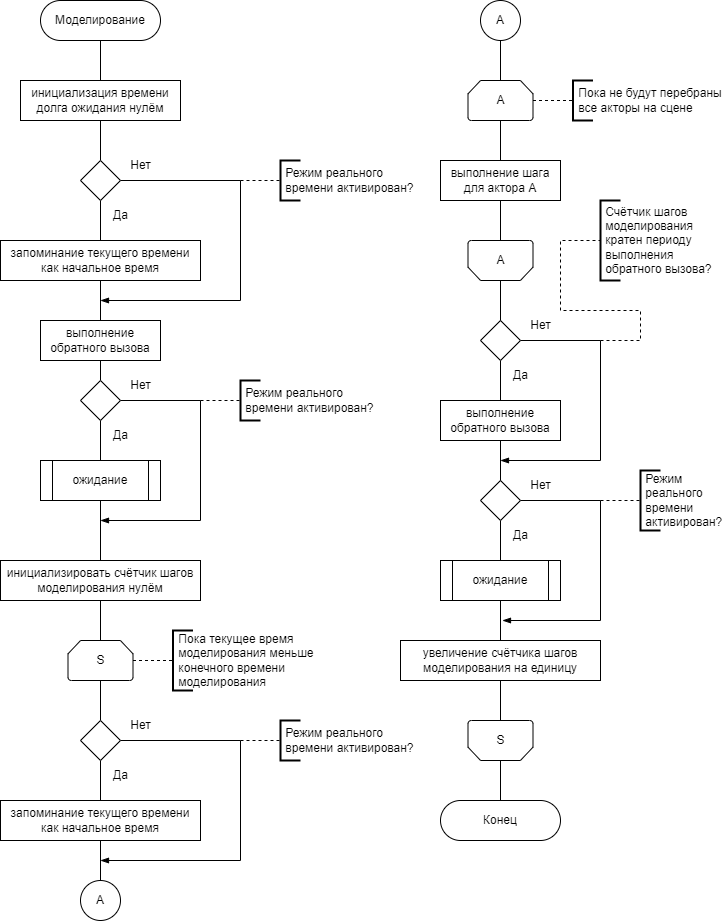


Рисунок 5 – Схемы алгоритмов модуля сцены

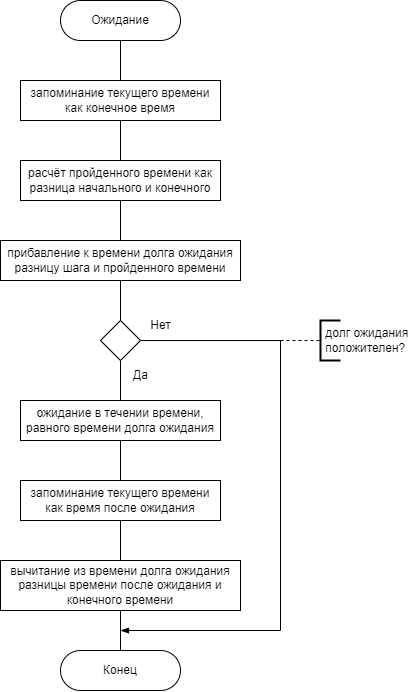


Рисунок 6 – Схемы алгоритмов модуля сцены

**ЗАКЛЮЧЕНИЕ**

В процессе прохождения преддипломной практики были описаны этапы разработки программного продукта. Создана структура приложения, разработана диаграмма последовательностей, определены основные функции приложения и разработан алгоритм моделирования, разработана диаграмма последовательностей и диаграмма классов предметной области.

**СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Десять популярных языков программирования для настольных приложений в 2021 году. [Электронный ресурс]. URL: <https://www.decipherzone.com/blog-detail/top-programming-languages-for-desktop-apps-in-2021> (дата обращения: 07.02.2023)
2. SIMODO в репозитории МГТУ им. Н.Э. Баумана. [Электронный ресурс]. URL: <https://bmstu.codes/lsx/simodo> (дата обращения: 07.02.2023).
3. Ещё раз про семь основных методологий разработки / Хабр [Электронный ресурс]. URL: <https://habr.com/ru/company/edison/blog/269789> (дата обращения: 07.02.2023)
4. Иванова, Г.С. Технология программирования: Учебник для вузов – М.: Изд-во МГТУ им. Н.Э. Баумана, 2002. 238 с.
5. IBM Developer [Электронный ресурс]. URL: <https://developer.ibm.com> (дата обращения: 07.02.2023).
6. ГОСТ 19.701­90 «ЕСПД. Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения»