|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий

Кафедра Математического обеспечения и стандартизации информационных технологий

**Задание по практической работе**

по дисциплине «Моделирование программных систем»

|  |  |
| --- | --- |
| **Выполнили:**  Студенты группыИКБО-11-22 | Андрусенко Л.Д.  Гришин А.В. |
| **Проверил:** | Образцов В.М. |

2024 г.

**Оглавление**

[Задание 3](#_Toc165137988)

[Часть 1. Создание диаграммы процесса. Изменение свойств блоков модели, её настройка и запуск 5](#_Toc165137989)

[Часть 2. Создание анимации модели 6](#_Toc165137990)

[Часть 3. Сбор статистики использования ресурсов 7](#_Toc165137991)

[Часть 4. Уточнение модели согласно ёмкости входного буфера 9](#_Toc165137992)

[Часть 5. Сбор статистики по показателям обработки запросов 10](#_Toc165137993)

[Часть 6. Добавление параметров и элементов управления 14](#_Toc165137994)

[Часть 7. Добавление гистограмм 17](#_Toc165137995)

[Часть 8. Изменение времени обработки запросов сервером 19](#_Toc165137996)

[Часть 9. Интерпретация результатов моделирования 21](#_Toc165137997)

[Вывод 22](#_Toc165137998)

# **Задание**

**Цель работы**: получение экспериментальной модели обработки запросов сервером.

**Постановка задачи:**

Построить модель работы сервера используя сети Петри. Использовать в качестве инструмента имитационного моделирования – Anylogic 8 PLE (бесплатная версия).

**Модель обработки запросов сервером**

Сервер обрабатывает запросы, поступающие с автоматизированных рабочих мест с интервалами, распределенными по показательному закону со средним значением 2 мин. Время обработки сервером одного запроса распределено по экспоненциальному закону со средним значением 3 мин. Сервер имеет входной буфер ёмкостью 5 запросов.

Сервер представляет собой однофазную систему массового обслуживания разомкнутого типа с ограниченной входной емкостью, то есть с отказами, и абсолютной надёжностью.

# **Часть 1. Создание диаграммы процесса. Изменение свойств блоков модели, её настройка и запуск**

1. Выполните Файл/Создать/Модель на панели инструментов. Появится диалоговое окно Новая модель. Задайте имя новой модели. В поле Имя модели введите Server.



Рисунок 1 – Создание новой модели

1. Создайте диаграмму процесса. Для этого в Палитре выделите Библиотеку моделирования процессов. Из неё перетащите объекты на диаграмму и соедините.

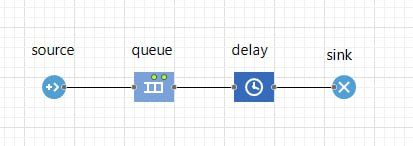


Рисунок 2 – Создание диаграммы процесса

1. Измените свойства объектов и запустите модель.

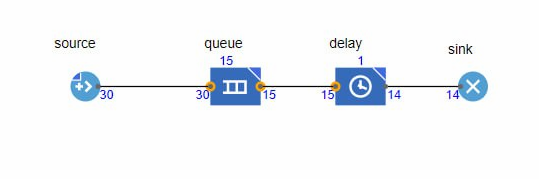


Рисунок 3 – Запуск модели

# **Часть 2. Создание анимации модели**

1. Нарисуйте прямоугольный узел, который будет обозначать на анимации сервер. Нарисуйте путь, который будет обозначать на анимации очередь к серверу.

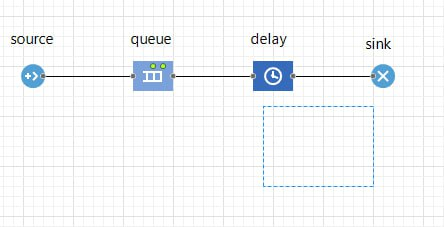


Рисунок 4 – Добавление прямоугольного узла

1. Теперь мы должны задать созданные анимационные объекты в качестве анимационных фигур объектов диаграммы нашего процесса. Задайте путь в качестве фигуры анимации очереди.

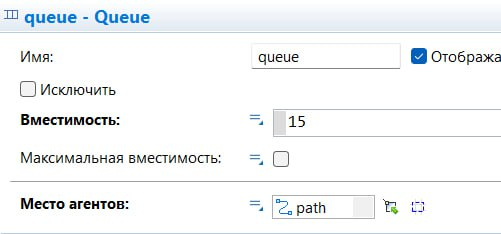


Рисунок 5 – Установка места агентов у очереди

1. Задайте прямоугольный узел в качестве фигуры анимации сервера.

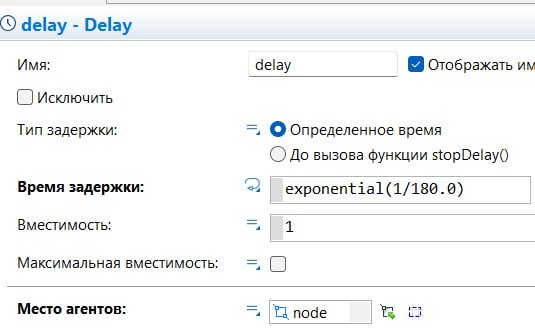


Рисунок 6 – Установка места агентов у задержки

1. Запустите модель. Вы увидите, что у модели теперь есть простейшая анимация — сервер и очередь запросов к нему. Цвет фигуры сервера будет меняться в зависимости от того, обрабатывается ли запрос в данный момент времени или нет.

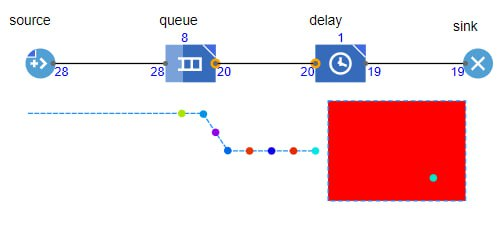


Рисунок 7 – Запуск модели

# **Часть 3. Сбор статистики использования ресурсов**

1. Добавьте диаграмму для отображения среднего коэффициента использования сервера.

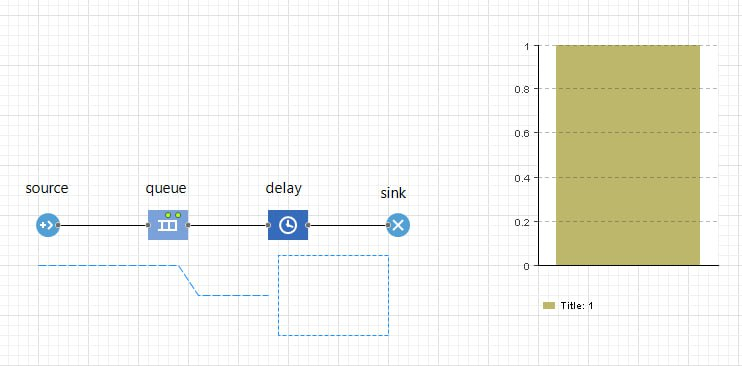


Рисунок 8 – Добавление диаграммы

1. Аналогичным образом добавьте еще одну столбиковую диаграмму для отображения средней длины очереди.

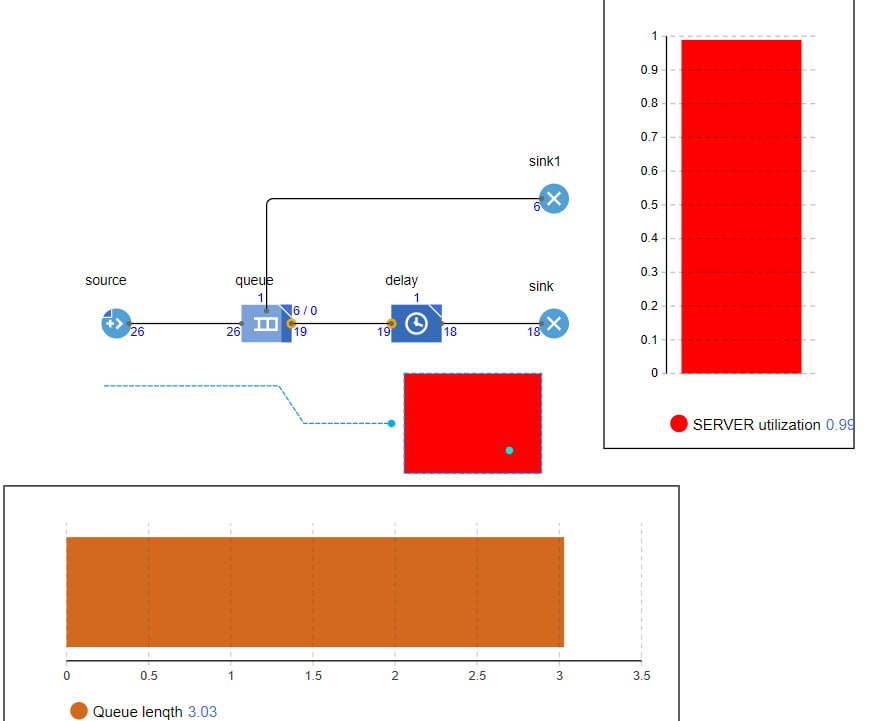


Рисунок 9 – Добавление диаграммы

1. Запустите модель с двумя столбиковыми диаграммами, установив модельное время 3600 единиц, и понаблюдайте за её работой.

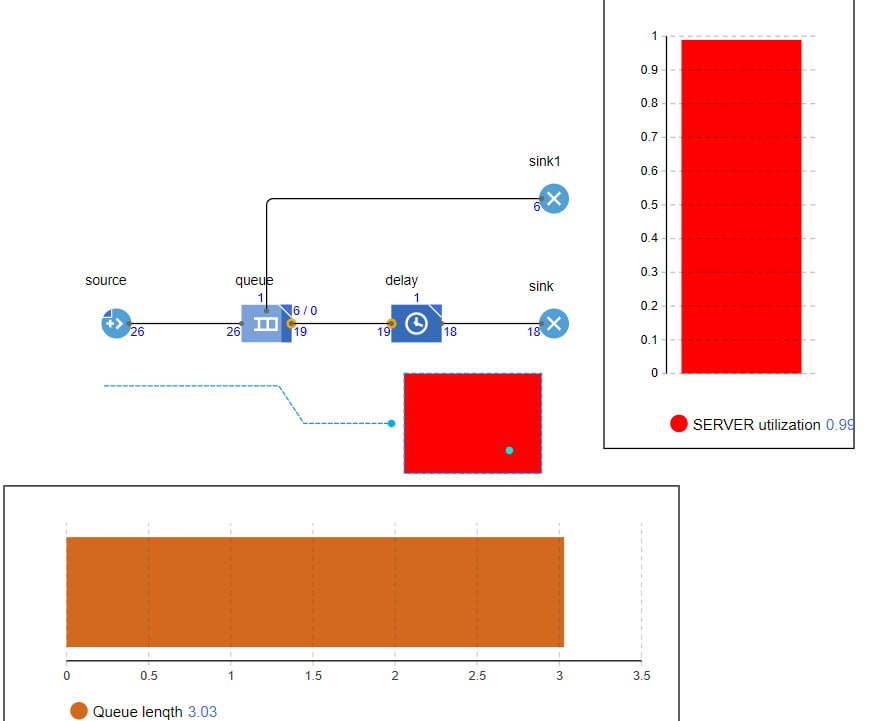


Рисунок 10 – Запуск модели

# **Часть 4. Уточнение модели согласно ёмкости входного буфера**

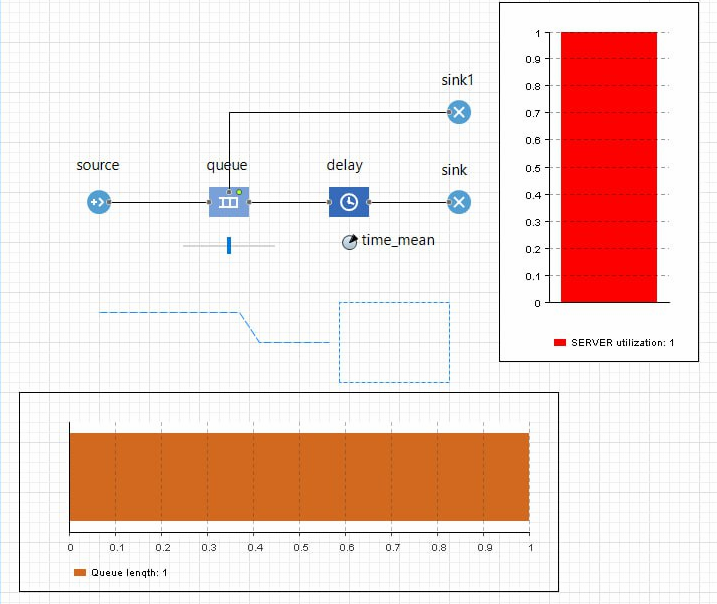
1. Для уничтожения потерянных запросов вследствие полного заполнения накопителя нужно добавить второй объект sink. Соедините порт outPreempted объекта queue с входным портом InPort блока sink1. 

Рисунок 11 – Добавление вытеснения заявок

# **Часть 5. Сбор статистики по показателям обработки запросов**

1. Для включения в запросы дополнительных полей необходимо создать нестандартный тип заявки. Это возможно двумя способами. Создадим первым способом тип заявок Inquiry. Создадим первым способом тип заявок Inquiry.



Рисунок 12 – Создание нового типа агента через создание нового Java класса

1. Появится окно c автоматически созданными параметрами нестандартного типа заявок Inquiry.

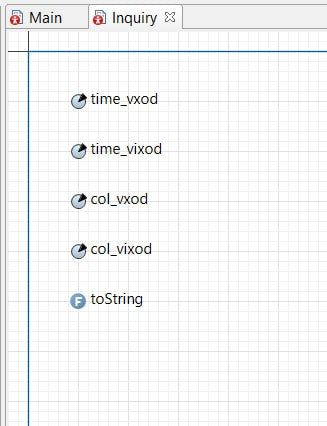


Рисунок 13 – Новый созданный класс

1. Перейдём к созданию непосредственно нестандартного типа заявки вторым способом. Выберите анимацию агента: установите 2D и выберите из выпадающего списка, например, Сообщение.

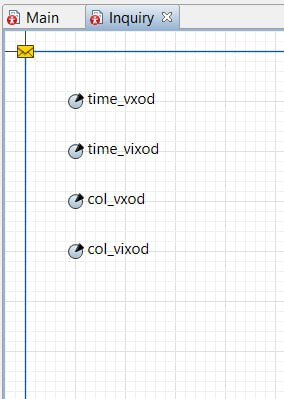


Рисунок 14 – Создание нового класса

1. Для сбора статистических данных о времени обработки запросов сервером необходимо добавить элемент статистики. Этот элемент будет запоминать значения времен для каждого запроса. На основе этого он предоставит пользователю стандартную статистическую информацию (среднее, минимальное, максимальное из измеренных значений, среднеквадратичное отклонение и т.д.).

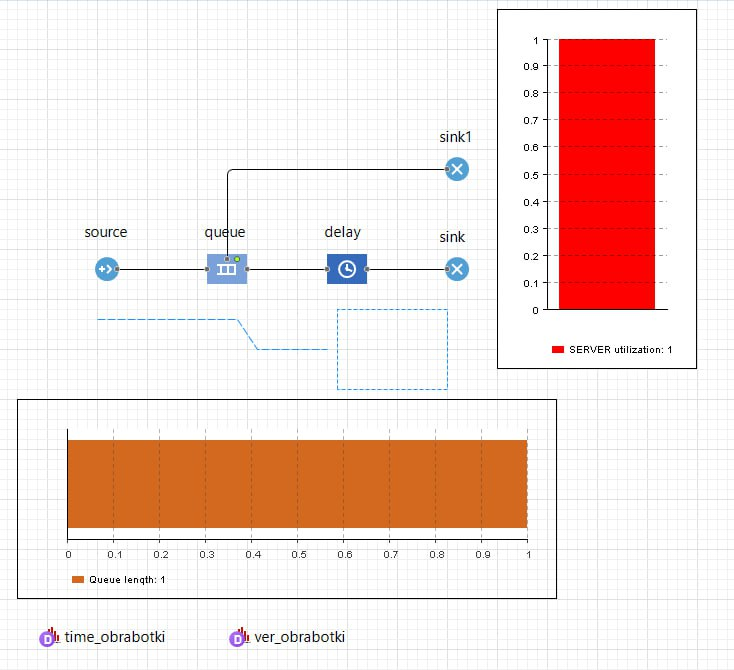


Рисунок 15 – Добавление элементов статистики

1. Измените код при входе элемента sink.

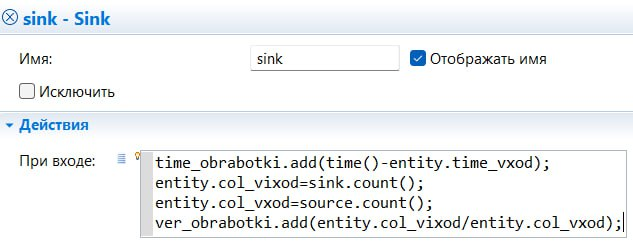


Рисунок 16 – Измененный код объекта sink

1. Обратите внимание, что вам не пришлось использовать поле time\_vixod, так как вместо него была использована функция time(), возвращающая, как вам уже известно, текущее значение модельного времени. Удалите поле time\_vixod.

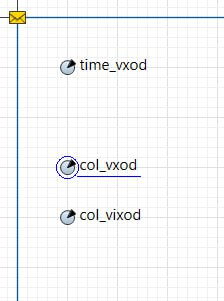


Рисунок 17 – Удалено поле time\_vixod

1. Итак, все условия постановки задачи выполнены. Чтобы наблюдать за работой модели, установите, что время остановки модели не задано. Запустите модель.

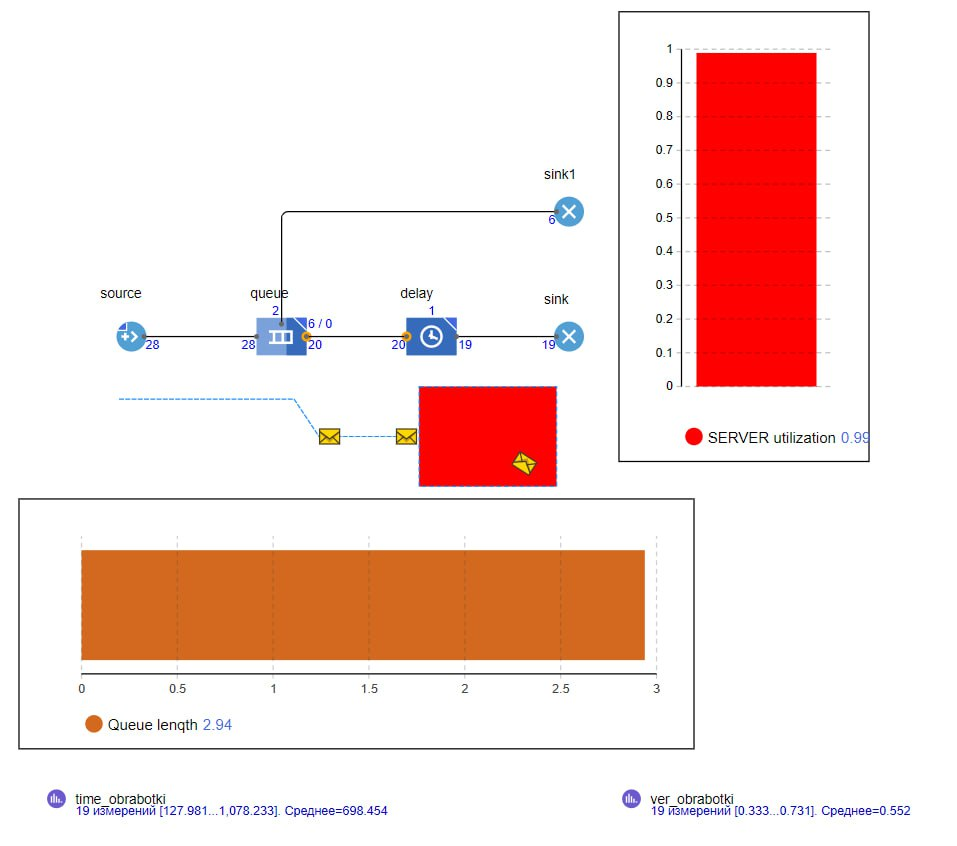


Рисунок 18 – Запуск модели

# **Часть 6. Добавление параметров и элементов управления**

1. Активный объект может иметь параметры. Параметры обычно используются для задания статических характеристик объекта. Но значения параметров при необходимости можно изменять во время работы модели. Для этого нужно написать код обработчика события, то есть действий, которые должны выполняться при изменении значения параметра. Создайте параметр time\_mean объекта delay.

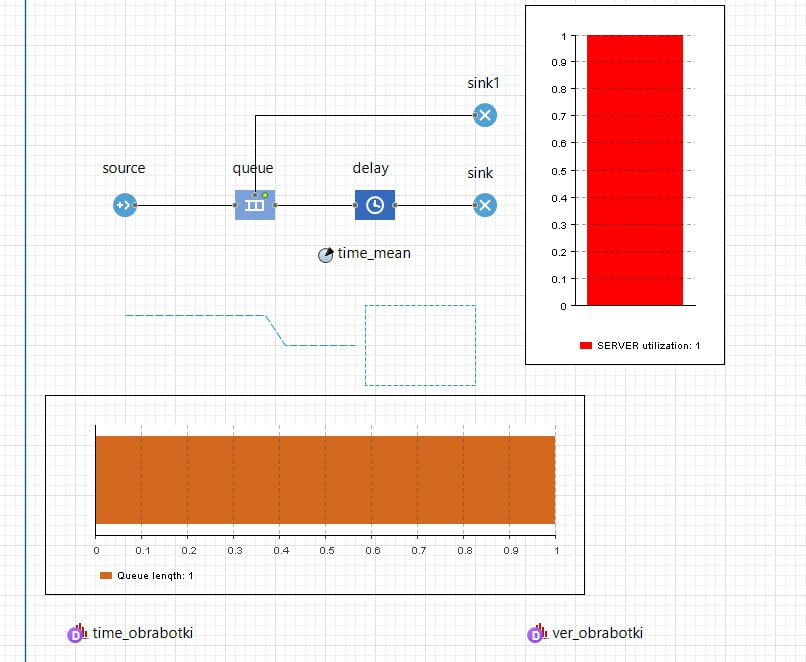


Рисунок 19 – Добавлен параметр time\_mean

1. Пусть вы хотите изменять среднее время обработки запросов time\_mean в ходе моделирования. Используйте для этого элемент управления — бегунок.

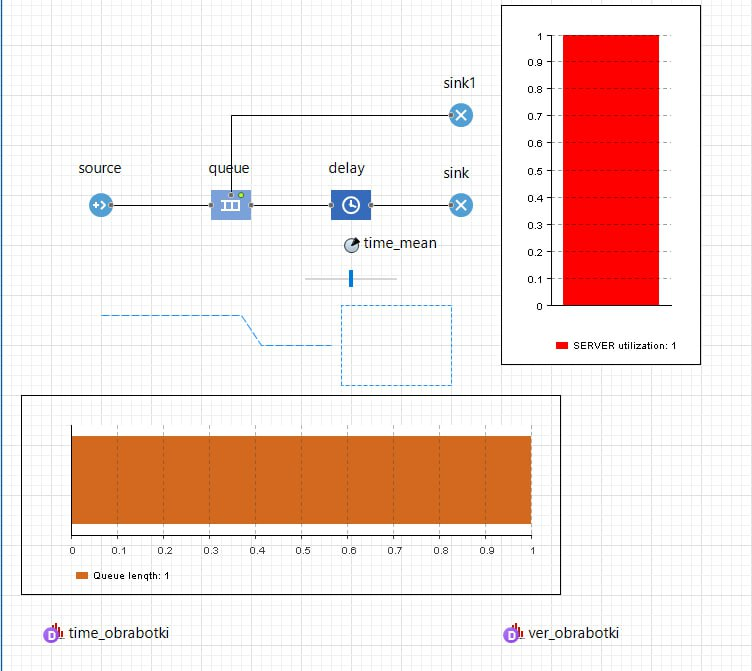


Рисунок 20 – Добавление ползунка

1. Пусть теперь вы хотите также изменять ёмкость буфера в ходе моделирования. Используйте для этого также бегунок.

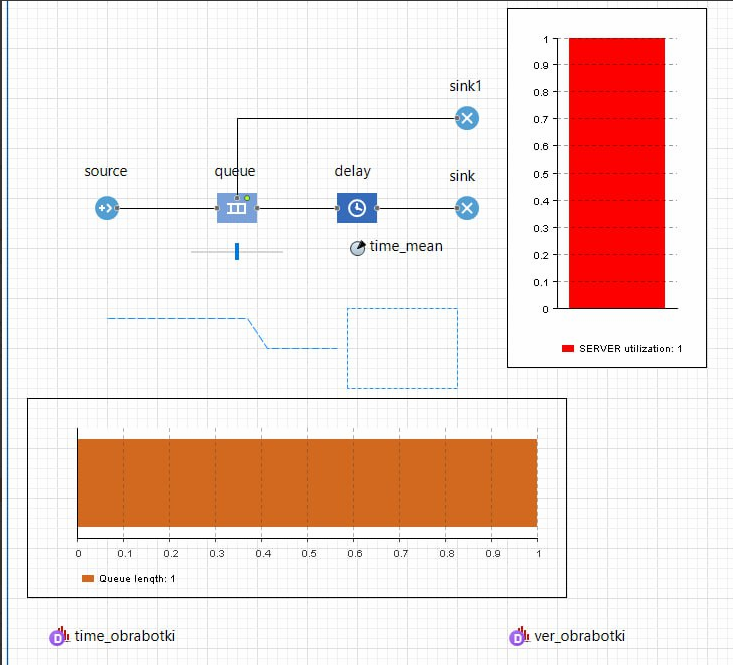


Рисунок 21 – Добавление ползунка

1. Существует и другой способ изменения свойств объектов во время выполнения модели: нужно щёлкнуть по элементу, войти в режим редактирования и ввести новое значение в одной из закладок всплывающего окна инспекта. Поэтому заранее не нужно продумывать, значения каких параметров планируется изменять, и не добавлять специальные элементы управления (например, бегунки).

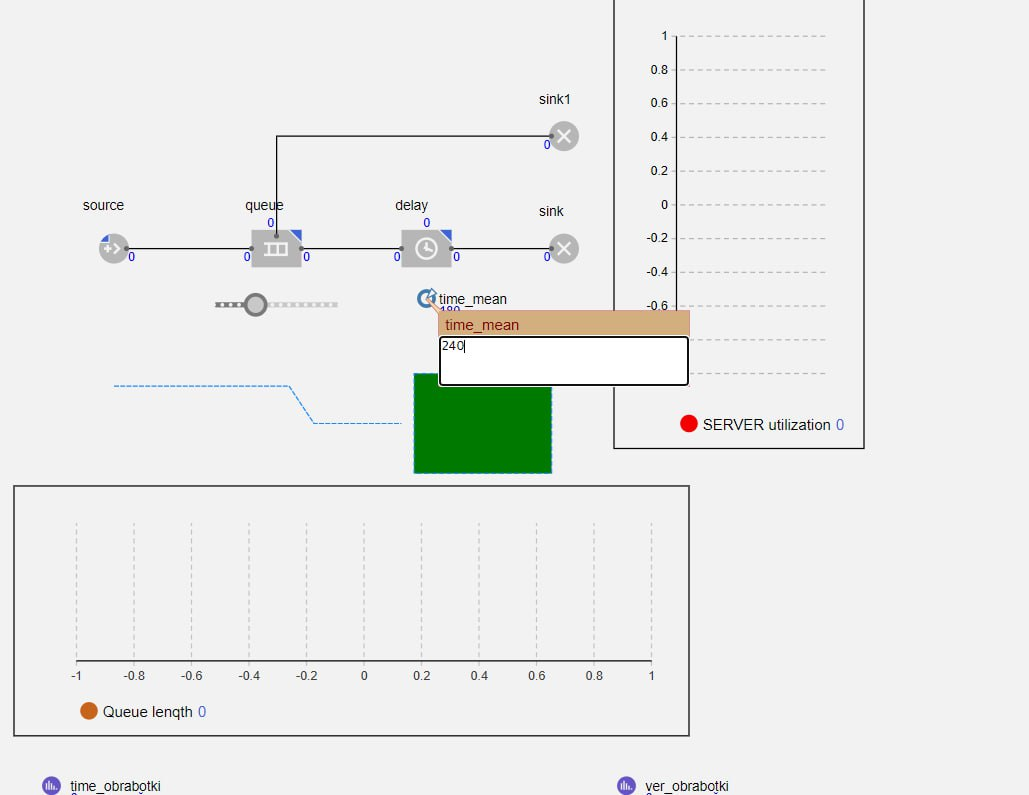


Рисунок 22 – Динамическое изменение параметра

# **Часть 7. Добавление гистограмм**

1. Теперь добавим на диаграмму нашего потока гистограмму, которая будет отображать собранную временную статистику.

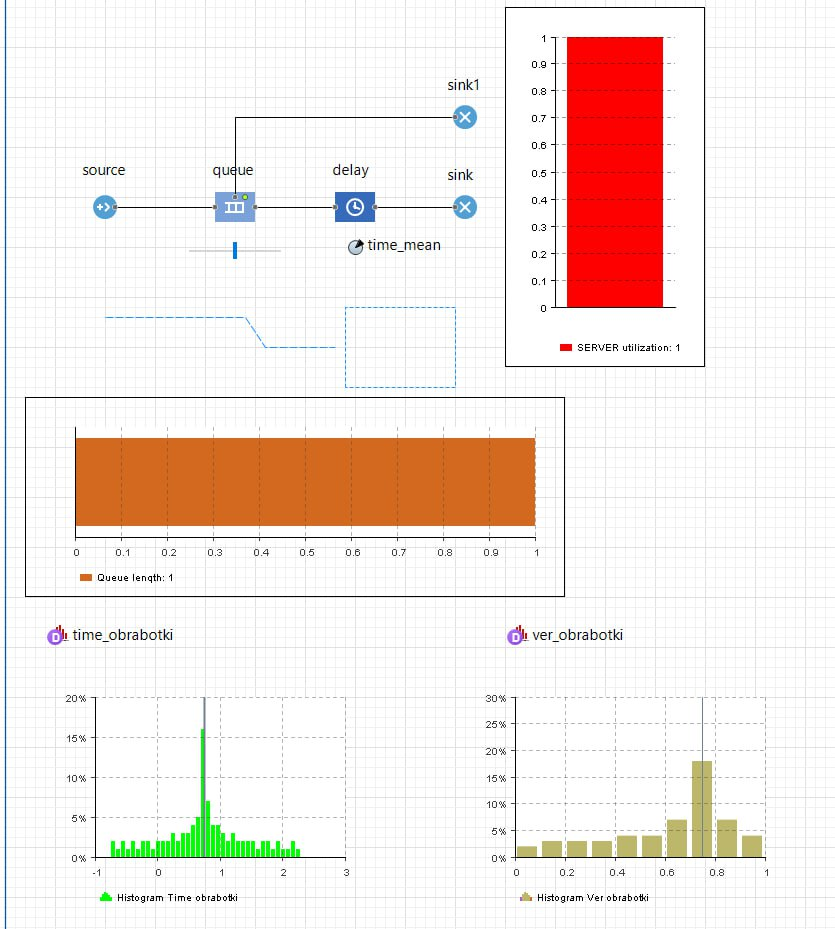


Рисунок 23 – Добавление гистограмм

1. Запустите модель.

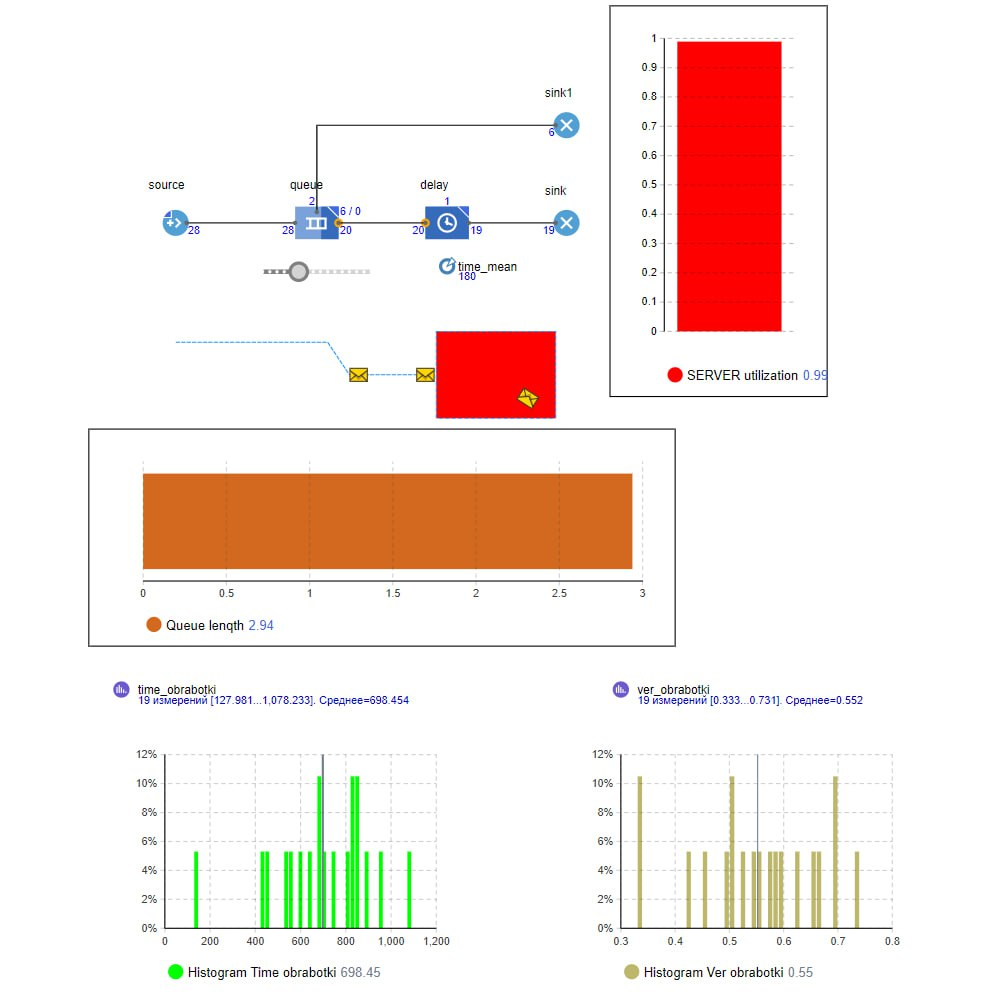


Рисунок 24 – Запуск модели

# **Часть 8. Изменение времени обработки запросов сервером**

1. Внесите в модель изменения для аналогичного расчёта времени обработки запросов. Из палитры Основная перетащите три элемента Параметр на диаграмму класса Main.

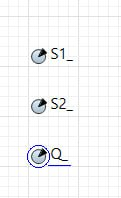


Рисунок 25 – Добавление параметров

1. Показатели моделируемой системы нужно определить в течение 3600 с, поэтому время моделирования в AnyLogic составит 3600\*9604 = 34574400 единиц модельного времени.

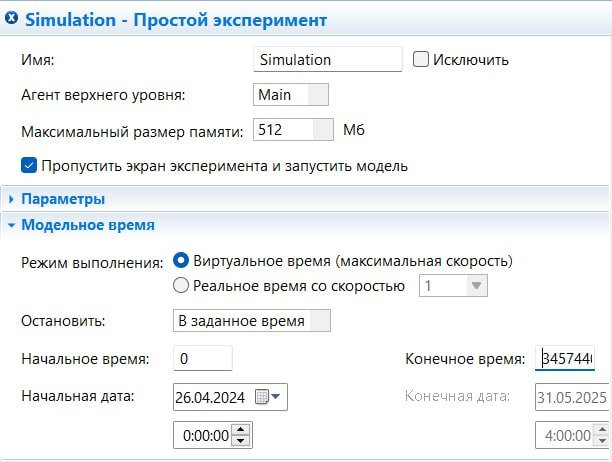


Рисунок 26 – Изменение свойств эксперимента

1. Запустите модель и дождитесь окончания моделирования.

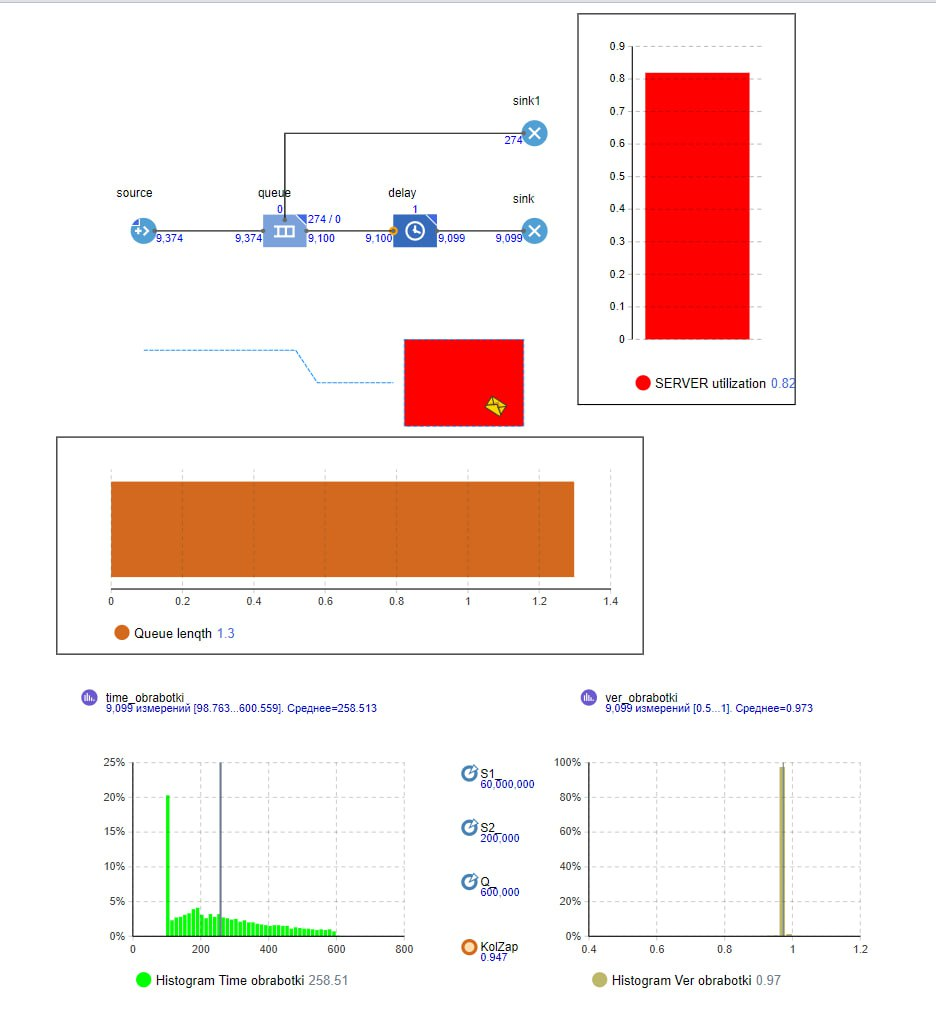


Рисунок 27 – Запуск модели

# **Часть 9. Интерпретация результатов моделирования**

1. Для проведения исследований на модели сделайте ещё несколько дополнений и изменений. Можно было бы обойтись и без них, но они улучшат эксплуатацию модели. Из палитры Основная перетащите три элемента Параметр на диаграмму агента Main.

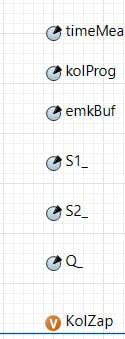


Рисунок 28 – Добавление трех параметров

1. Запустите модель.

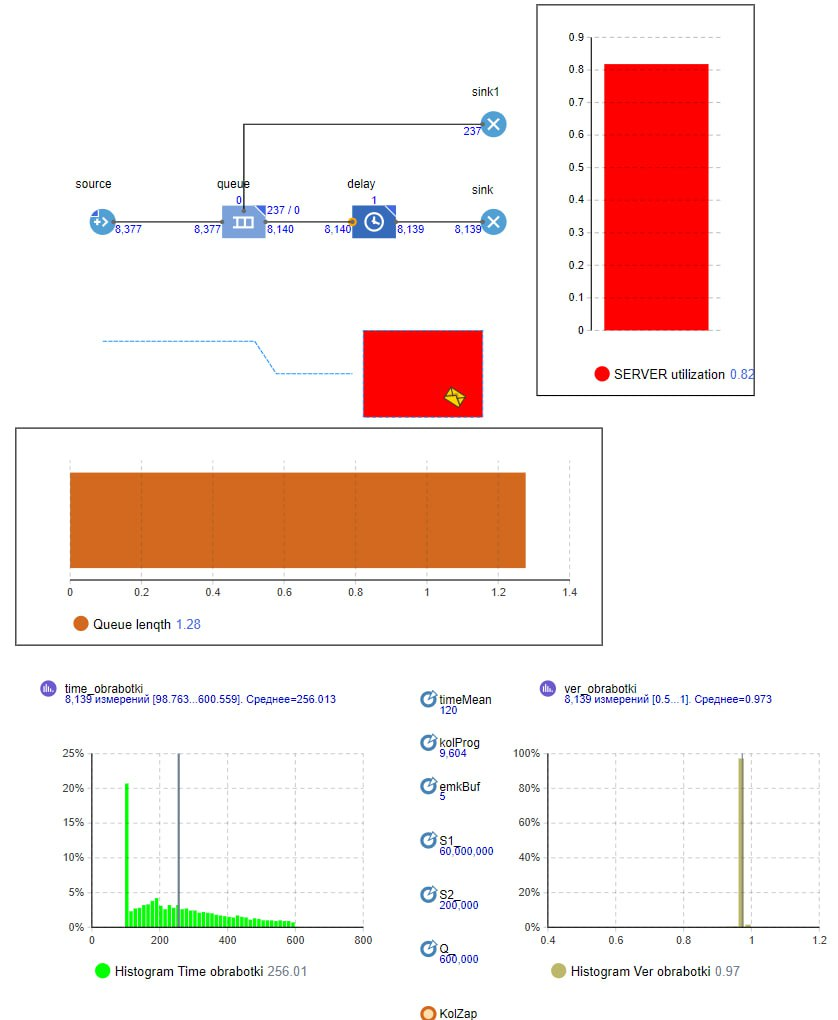


Рисунок 29 – Запуск модели

# **Вывод**

В ходе данной практической работы мы построили модель обработки запросов сер.

Теперь мы лучше понимаем, как работает однофазная система массового обслуживания разомкнутого типа с ограниченной входной емкостью, то есть с отказами, и абсолютной надёжностью.

В процессе построения данной модели мы овладели базовыми знаниями о ресурсах AnyLogic и приемах работы с ними. Так же научились описывать процессы и собирать статистику для дальнейшей работы с ней.