

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ЯДЕРНЫЙ
УНИВЕРСИТЕТ «МИФИ»

Институт Ядерной Физики и Технологий
Кафедра №40 "Физика элементарных частиц"

ОТЧЕТ
О НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ

**Разработка графического интерфейса создания
дашбордов и визуализация событий в
эксперименте BAIKAL-GVD**

Студент группы М24-112

Сафронова М.А.

Научный руководитель

д.ф.- м.н. Наумов. Д.В.

Москва 2024

ОГЛАВЛЕНИЕ

1	Введение	2
2	Эксперимент BAIKAL-GVD	4
2.1	Основная цель эксперимента	4
2.2	Байкальский нейтринный телескоп	4
3	Программный комплекс NTSim	9
3.1	Общая структура пакета	9
3.2	Генераторы первичных взаимодействий	9
3.3	Пропагаторы частиц	10
4	Полученные результаты	11
5	Разработка сервисов для работы с данными.	14
6	Заключение	17
	Список использованных источников	18

1 ВВЕДЕНИЕ

Одной из наиболее важных проблем в современной фундаментальной науке является изучение нейтрино. В частности, в последние несколько десятилетий огромное внимание уделено нейтринной астрономии. К её основным задачам относятся:

- измерение свойств и энергий нейтринных потоков;
- определение положений космических источников нейтрино;
- изучение реликтовых нейтрино и получение информации о процессах, происходивших в ранней Вселенной.

Известно, что нейтрино обладают огромной проникающей способностью (например, нейтрино энергией порядка 10 МэВ может пролететь без взаимодействия в воде расстояние порядка 10^{18} м, земной шар становится непроницаемым для нейтрино только при энергиях порядка 1000 ГэВ). Поскольку данные частицы ещё и не обладают электрическим зарядом, они представляют особую ценность в области исследования космических объектов, потому что могут долететь от источника (сверхновые звезды, черные дыры, активные ядра галактик) до Земли без каких-либо взаимодействий и сохранить информацию об этом источнике и о процессах, происходящих в космосе.

Для исследования потоков нейтрино на Земле созданы нейтринные телескопы. В силу своих свойств нейтрино не детектируются непосредственно. Схему их обнаружения можно разбить на два этапа.

- 1) Образование заряженных частиц при взаимодействии с веществом среды.

Одной из наиболее распространённых реакций в этом случае является:

$$p + \nu_{\mu} \rightarrow \mu + n \quad (1.1)$$

- 2) Образовавшиеся мюоны имеют большую энергию, поэтому при движении в плотной среде (например, в воде) их скорость превосходит скорость света в данной среде, в результате чего возникает черенковское излучение, которое фиксируется нейтринным телескопом

Поскольку сечение взаимодействия нейтрино с веществом очень мало, нейтринные детекторы должны иметь огромную площадь. Создание подобных конструкций вручную потребовало бы огромных затрат, поэтому в качестве детекторов чаще всего выступают естественные объекты. А учитывая, что нужна защита от фонового космического излучения, получается, что есть всего два варианта расположения нейтринных телескопов: под водой и под землей.

Целью данной работы является: обзор крупнейшего в мире (наряду с детектором IceCube) подводного нейтринного телескопа Байкал, ознакомление с пакетом NTSim, визуализация сгенерированных фотонов, полученных с помощью данного пакета, а также разработка пользовательских сервисов для работы со сбором и обработкой данных.

2 ЭКСПЕРИМЕНТ BAIKAL-GVD

2.1 ОСНОВНАЯ ЦЕЛЬ ЭКСПЕРИМЕНТА

Необходимость строительства детектора гигатонного объёма (Gigaton Volume Detector, сокр. GVD) в Северном полушарии появилась после того как в 2013 в эксперименте IceCube (расположенный в Южном полушарии) обнаружили нейтрино с энергией выше 1000 ТэВ. Таким образом, эксперимент Байкал является важнейшим элементом Глобальной нейтринной сети (система из детекторов IceCube, ANTARES, KM3NeT и BAIKAL-GVD) которая позволяет обнаруживать нейтрино со всей небесной сферы.

Таким образом, основной целью эксперимента BAIKAL-GVD является регистрация и исследование потоков нейтрино сверхвысоких энергий от астрофизических источников, расположенных в Северном полушарии Земли. С его помощью ученые планируют исследовать процессы с огромным выделением энергии, в том числе и те, которые происходили в ранней Вселенной. Кроме того существует вероятность найти ответ на вопрос о механизме рождения астрофизических нейтрино, энергия которых в миллиарды раз превосходит энергию солнечных нейтрино.

2.2 БАЙКАЛЬСКИЙ НЕЙТРИННЫЙ ТЕЛЕСКОП

Байкальский нейтринный телескоп расположен в южной части озера Байкал на расстоянии 3.6 км от берега на глубине примерно 1.1 км (см. 2.1)

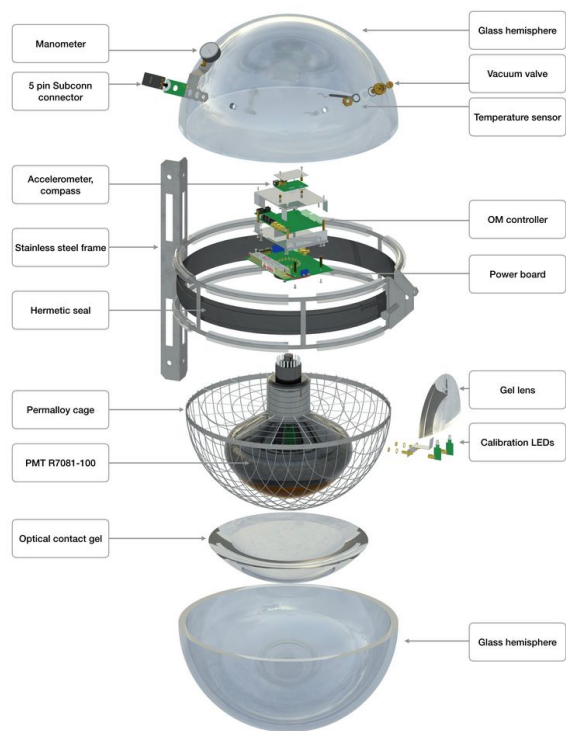


Рисунок 2.1 — Расположение нейтринного телескопа Байкал (указано звездочкой)

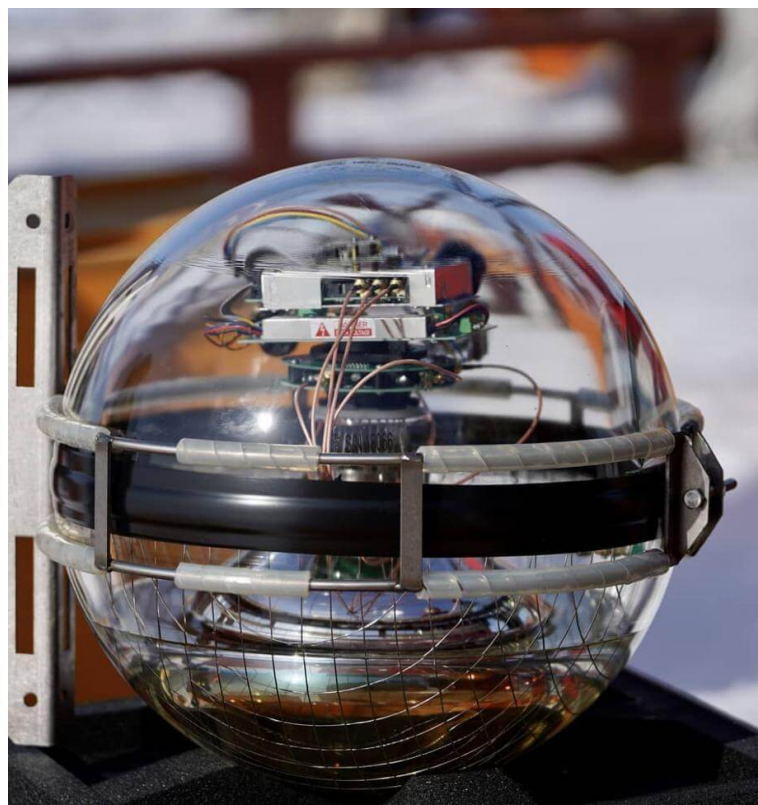
Создание больших детекторов для исследований в области нейтринной астрофизики в таком естественном водоеме как озеро Байкал имело следующие преимущества:

- 1) наличие мест в озере Байкал с глубиной более 1 км, расположенных недалеко от берега;
- 2) поглощение и рассеяние света в Байкальской воде мало. Длина поглощения света в месте дислокации детектора составляет 20 м. Длина рассеяния около 15 м;
- 3) наличие ледового покрова в течение приблизительно 8 недель существенно облегчает развертывание детектора (по сравнению с реализацией аналогичных проектов в океане).

Байкальский нейтринный телескоп имеет модульную структуру. Он состоит из большого количества вертикально расположенных струн, которые с одной стороны закрепляются на дне озера, а на поверхности удерживаются натянутыми с помощью системы буёв. Общая схема детектора представлена на рисунке 5.2.



(a)



(б)

Рисунок 2.3 — Оптические модули байкальского нейтринного телескопа: **а** схема, **б** реальный вид.

ОМ расположены на струнах, образуя семиугольный кластер, с 8 струнами, расположенными в центре и вершинах семиугольника, каждая из которых находится на расстоянии около 60 м друг от друга. Модули струн из каждой струны подключаются к кластерному центру, который, в свою очередь, соединяется с береговой станцией через электрооптический кабель. Кластерный центр отвечает за синхронизацию струн, управление запуском на уровне кластера и передачу данных на берег. Кластеры расположены на расстоянии около 300 м друг от друга, измеренном от их геометрических центров. Кластеры Baikal-GVD работают независимо друг от друга, поэтому возможно проводить однокластерный или многокластерный анализ.

С 2019 года ежегодно добавлялось два кластера, за исключением 2021 года, когда был установлен только один. Для повышения чувствительности к событиям каскада нейтрино в 2022 году в центре каждого триплета кластеров были введены межкластерные струны (ICS). Моделирование Монте-Карло показало 24%-ное увеличение обнаружения астрофизических событий для каскадов свыше 100 ТэВ . К 2023 году были установлены три ICS, каждый из которых был оснащен лазерным источником для калибровки между кластерами.

К настоящему моменту (2024 год) установлено 14 кластеров и их число в дальнейшем будет увеличиваться.

3 ПРОГРАММНЫЙ КОМПЛЕКС NTSIM

3.1 ОБЩАЯ СТРУКТУРА ПАКЕТА

NTSim представляет собой набор независимых модулей, для каждого из которых прописаны классы: Base и Factory[2]. С помощью абстрактных методов в Base определяется структура модуля, Factory необходим для интеграции реализаций каждого модуля в моделирование.

В NTSim прописаны генераторы и пропагаторы. Генераторы начинают моделирование, создавая первичные объекты, распространение в среде которых можно промоделировать с помощью пропагаторов.

Генераторы создают объект gEvent, который включает в себя такие объекты как: gHits, gPhotons, gParticles, gTracks. Вышеперечисленные объекты являются базовыми для пакета и содержат в себе информацию о типе моделируемой частицы, её энергии, времени генерации, направления и др.

3.2 ГЕНЕРАТОРЫ ПЕРВИЧНЫХ ВЗАИМОДЕЙСТВИЙ

Для инициализации моделирования используется модуль PrimaryGenerator, который реализован несколькими способами.

В данной работе использован генератор Laser, который представляет собой модуль, позволяющий моделировать световые импульсы от лазерных калибровочных установок. Лазерные калибровочные установки необходимы для временной калибровки ОМ.

Также был использован генератор IsotropicSource, позволяющий имитировать изотропное распределение.

3.3 ПРОПАГАТОРЫ ЧАСТИЦ

Пропагаторы позволяют осуществить распространение частиц в веществе, для этого данные, полученные из PrimaryGenerator, передаются на первичный пропагатор PrimaryPropagator, на выходе из которого получается объект gTracks, содержащий в себе информацию об эволюции трека частицы.

4 ПОЛУЧЕННЫЕ РЕЗУЛЬТАТЫ

Задачей данной работы была визуализация распространения фотонов в среде. Для этого использован пакет NTSim. С помощью модуля PrimaryGenerator смоделирован объект gPhotons, который содержит в себе следующие переменные:

- `_n_photons` - количество смоделированных фотонов;
- `_n_steps` - количество шагов рассеяния;
- `_position_m` - массив numpy, содержащий в себе значения координат фотона;
- `_time_ns` - массив значений времени прихода фотона в определенную точку;
- `_direction` - направление движения фотона;
- `_wavelength_nm` - длина волны фотона;
- `_progenitor` - уникальный номер частицы, породившей фотон;
- `_ta_ns` - среднее время поглощения фотона в среде;
- `_ts_ns` - среднее время рассеяния фотона в среде;

После генерации был использован пропагатор, чтобы распространить фотоны в среде, а именно в воде Байкала. На выходе был получен обновленный объект gPhotons, содержащий в себе координаты рассеяния. Количество шагов рассеяния равно пяти, то есть в массиве `_position_m` содержится пять координат для каждого из фотонов. Для визуализации пролета фотонов необходимо было получить промежуточные координаты между рассеиваниями. Для решения данной задачи использован метод интерполяции библиотеки numpy. Для плавной визуализации количество промежуточных координат между начальной и конечной точкой было выбрано равным 200.

Полученные траектории были визуализированы с помощью таких библиотек, как PyQt5 и pyqtgraph. Полученная визуализация представлена на рисунке 4.1.

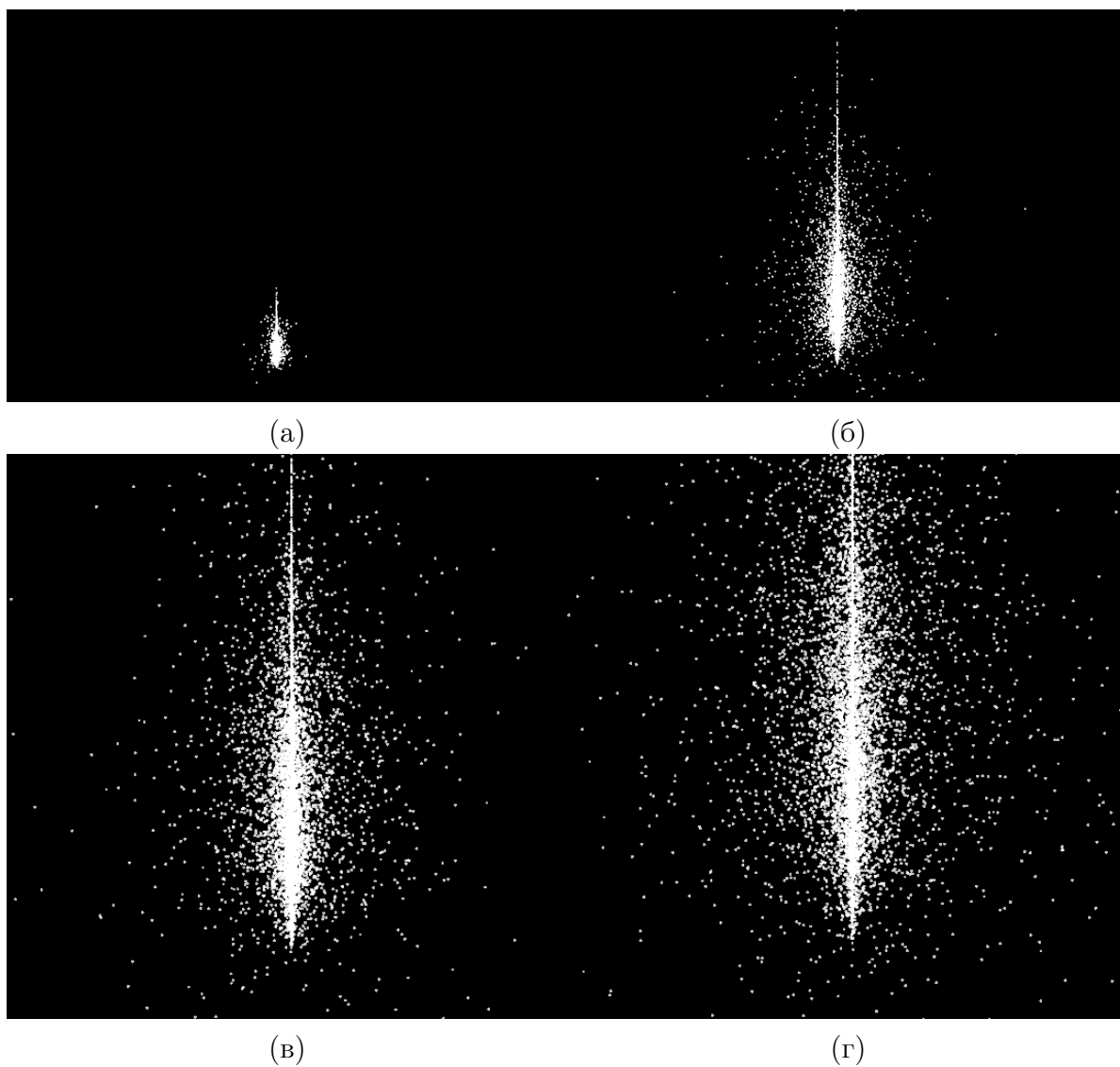
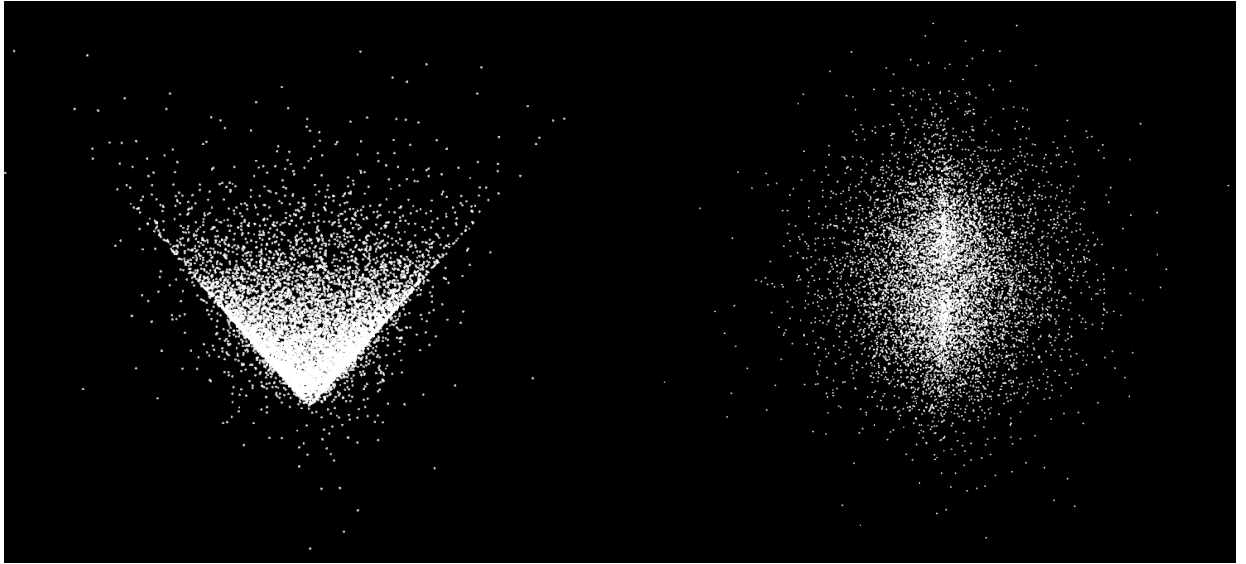


Рисунок 4.1 — Визуализация прохождения фотонов через воду.

Также были использованы возможности генератора Laser и IsotropicSource промоделировать конусообразное и изотропное распределения соответственно. Полученные результаты представлены на рисунке [4.2](#).



(a)

(б)

Рисунок 4.2 — Результат работы пропагатора: **а** конусообразное распределение, **б** имитация изотропного источника.

5 РАЗРАБОТКА СЕРВИСОВ ДЛЯ РАБОТЫ С ДАННЫМИ.

Детектор BAIKAL-GVD собирает данные пока идет время сеанса работы - рана. Для взаимодействия со всем процессом существует сервис, который позволяет клиенту взаимодействовать с раном через браузер.

Для начала работы было написано два сервиса, один из которых читал сообщения с другого. В качестве инструментов использован Flask - фреймворк для разработки веб-приложений на языке программирования python.

Для реализации данного задания написано два скрипта, каждый из которых запускает веб-сервис на локальном хосте. В один из скриптов добавлен код взаимодействия с другим сервисом с помощью сокетов. В Python модуль socket предоставляет низкоуровневый API для работы с сетевыми соединениями.

Рассмотрим создание первого сервиса. Сначала прописывается функция, которая включает в себя создание TCP - сокета, который привязывается к порту, а далее ждет подключения. При появлении подключения сокет отправляет сообщение другому сервису.

Создание второго сервиса реализовано с помощью Flask. Для связи с первым сервисом создан TCP-сокеты, данный сокет подключается к порту первого сервиса и получает сообщение частями определенного размера, после чего полученное сообщение выводится на экран.

```

1  import socket
2  from flask import Flask, jsonify, render_template
3
4
5  def run_service():
6      HOST = '127.0.0.1'
7      PORT = 5002
8
9      with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
10         s.bind((HOST, PORT))
11         s.listen()
12         print(f"Socket service запущен на {HOST}:{PORT}")
13
14         while True:
15             conn, addr = s.accept()
16             with conn:
17                 print('Socket client подключён:', addr)
18                 data_to_send = "HTTP/1.1 200 OK\r\nContent-Type:
19                 conn.sendall(data_to_send.encode())
20                 conn.close()
21
22
23
24
25  if __name__ == '__main__':
26      run_service()

```

Рисунок 5.1 — Реализация первого сервиса.

```

69 from flask import Flask, render_template_string
70 import socket
71 import threading
72 from queue import Queue
73
74 app = Flask(__name__)
75
76 messages = Queue()
77
78 HTML_TEMPLATE = """
79 <!DOCTYPE html>
80 <html>
81 <head>
82     <title>Socket Service</title>
83     <style>
84         body { font-family: Arial, sans-serif; margin: 20px; }
85         .message { padding: 10px; margin: 5px; background: #f0f0f0; border-radius: 5px; }
86     </style>
87 </head>
88 <body>
89     <h1>Полученные сообщения:</h1>
90     <div id="messages">
91         {% for msg in messages %}
92         <div class="message">{{ msg }}</div>
93         {% endfor %}
94     </div>
95 </body>
96 </html>
97 """
98
99 def socket_service():
100     host = '127.0.0.1'
101     port = 5002
102
103     with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
104         s.connect((host, port))
105         print(f"Сервер запущен на {host}:{port}")
106
107         response = b""
108
109         while True:
110             chunk = s.recv(4096)
111             if not chunk:
112                 break
113             response += chunk
114
115             headers_end = response.find(b"\r\n\r\n")
116             if headers_end == -1:
117                 raise ValueError("Invalid HTTP response")
118
119             raw_headers = response[:headers_end].decode()
120             body = response[headers_end + 4:].decode()
121
122             if body:
123                 messages.put(body)
124
125             status_line = raw_headers.split("\r\n")[0]
126             status_code = int(status_line.split()[1])
127
128             headers = {}
129             for line in raw_headers.split("\r\n")[1:]:
130                 if ":" in line:
131                     key, value = line.split(":", 1)
132                     headers[key] = value
133
134             return {
135                 "status code": status_code,
136                 "headers": headers,
137                 "body": body,
138             }
139
140 @app.route('/')
141 def index():
142     current_messages = list(messages.queue)
143     return render_template_string(HTML_TEMPLATE, messages=current_messages)
144
145 if __name__ == '__main__':
146     socket_thread = threading.Thread(target=socket_service)
147     socket_thread.daemon = True
148     socket_thread.start()
149
150     app.run(port=5000, debug=True)

```

Рисунок 5.2 — Реализация второго сервиса.

6 ЗАКЛЮЧЕНИЕ

В ходе работы было проведено ознакомление с основными нейтринными экспериментами, также были изучены принципы работы нейтринных детекторов, таких как IceCube, KM3NeT, BAIKAL-GVD.

В рамках данной научной работы был изучен и использован пакет NTSim, представляющий собой набор независимых модулей, предназначенных для моделирования частиц.

Для удобства пользования данным пакетом разрабатывается графический интерфейс, который позволит пользователю получать визуально данные из моделирования, такие как визуализация полета частицы в веществе, различные распределения и т.д. Задачей текущей научной работы стало написание скрипта, позволяющего визуализировать рассеяние фотонов в воде. Для этого использованы генераторы первичных взаимодействий и пропагаторы частиц. С помощью генераторов смоделированы световые импульсы, которые далее были распространены в веществе пропагатором. На основании полученных данных осуществлена визуализация фотонов, движущихся в определенном направлении.

Также была реализована визуализация определенных распределений: конусообразного и изотропного распределений. Для этого использовались такие генераторы пакета NTSim как Laser и IsotropicSource.

В дополнение созданы макеты сервисов, позволяющих производить взаимодействие друг с другом.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. *V. Allakhverdyan D. Naumov S. Z.* BAIKAL-GVD Neutrino Telescope: unlocking the secrets of the universe's catastrophic events // Phys. Rev. — 2024. — P. 11.
2. *A. Belyakova S. Zavyalov D. Z.* Development of the NTSim software package for modelling neutrino telescopes // Phys. Rev. — 2024. — P. 8.