**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ**

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ**

**«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук

Образовательная программа бакалавриата «Программная инженерия»

|  |  |  |
| --- | --- | --- |
| СОГЛАСОВАНО  Преподаватель департамента программной инженерии ФКН, кандидат компьютерных наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_С.А.Виденин  \_\_\_\_\_\_\_\_\_\_\_10\_марта 2024 г. |  | УТВЕРЖДАЮ  Академический руководитель образовательной программы «Программная инженерия»  профессор департамента программной инженерии, кандидат технических наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н.А. Павлочев  \_\_\_\_\_\_\_\_\_\_\_\_10\_марта\_2024 г. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | ***Подп. и дата*** |  | | ***Инв. № дубл.*** |  | | ***Взам. Инв. №*** |  | | ***Подп. и дата*** |  | | ***Инв. № подл.*** |  | | **ВЫСОКОПРОИЗВОДИТЕЛЬНЫЙ НАСТРАИВАЕМЫЙ HTTP СЕРВЕР**  **Руководство оператора**  **ЛИСТ УТВЕРЖДЕНИЯ**  **RU.17701729.04.04-01 34 01-1-ЛУ**  Исполнитель  студент группы БПИ214  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/ Е.К.Фортов/  \_\_\_\_\_\_\_\_\_\_\_\_\_10\_марта\_2024 г. | |
|  |  |

УТВЕРЖДЕН

RU.17701729.04.04-01 ТЗ 01-1-ЛУ

|  |  |
| --- | --- |
| ***Подп. и дата*** |  |
| ***Инв. № дубл.*** |  |
| ***Взам. инв. №*** |  |
| ***Подп. и дата*** |  |
| ***Инв. № подл*** |  |

**ВЫСОКОПРОИЗВОДИТЕЛЬНЫЙ НАСТРАИВАЕМЫЙ HTTP СЕРВЕР**

**Руководство оператора  
  
RU.17701729.04.04-01 34 01-1-ЛУ**

**Листов 11**

Оглавление

[1. НАЗНАЧЕНИЕ ПРОГРАММЫ 3](#_Toc72106850)

[1.1. Функциональное назначение 3](#_Toc72106851)

[1.2. Эксплуатационное назначение 3](#_Toc72106852)

[1.2. Состав функций 3](#_Toc72106852)

[2. УСЛОВИЯ ВЫПОЛНЕНИЯ ПРОГРАММЫ](#_Toc72106853) 5

[2.1.](#_Toc72106854) [Минимальный состав аппаратурных средств](#_Toc72273824)5

[2.2.](#_Toc72106855) [Минимальный состав программных средств](#_Toc72273825)5

[2.3.](#_Toc72106855) [Требования к персоналу (пользователю)](#_Toc72273826)5

[3.](#_Toc72106856) [ВЫПОЛНЕНИЕ ПРОГРАММЫ](#_Toc72273827)6

[3.1.](#_Toc72106857) [Загрузка программы](#_Toc72273828)6

[3.2.](#_Toc72106858) [Запуск программы](#_Toc72273829)6

[3.3.](#_Toc72106858) [Выполнение программы](#_Toc72273830)6

[3.3.1. Создание сервера](#_Toc72273831) 6

[3.3.2. Добавление эндпоинтов к серверу](#_Toc72273831) 6

[3.3.3. Запуск сервера](#_Toc72273831) 6

[3.3.4. Остановка сервера (необязательно)](#_Toc72273831) 6

[3.3.5. Наследование отдельных компонентов сервера](#_Toc72273831) 7

[4.](#_Toc72106859) [СООБЩЕНИЯ ОПЕРАТОРУ](#_Toc72273852)9

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ](#_Toc72106890) 10

# 1. НАЗНАЧЕНИЕ [Error: Reference source not found](#_Toc72106850)

## 1.1. Функциональное назначение

Функциональным назначением программы является предоставление программисту возможности быстро и удобно проектировать REST API на языке С++, не погружаясь в детали реализации REST API. Данный продукт является отличным решением для команд разработки, пишущих на С++, так как позволит им не менять стек разработки, в том числе и ЯП, при появлении требования в необходимости наличия REST API для каких бы то ни было целей.

## 1.2. Эксплуатационное назначение

Многие существующие IT продукты написаны на С++. Этот ЯП славится своей производительностью и универсальностью, однако многие более новые языки (например, Джава, С#, Го) позволяют разрабатывать такие же продукты в разы быстрее.

Зачастую командам невозможно поменять стек разработки по разным причинам (например, слишком большое наследние, функционал которого нельзя перенести на современный технологический стек с точки зрения бизнес-value и затраченного на разработку времени). Чтобы не оказаться вне рынка с текущим «устаревшим» продуктом, его разработчикам приходится имплементировать современные features на старом технологическом стеке. Для одной из таких features, а именно REST API, которое присутствует во многих промышленных системах, и предназначена данная библиотека. Она призвана значительно сокращать время на разработку REST API и добиваться наибольшего значения соотношения «бизнес-value / затраченное на разработку время».

Для подключения данной библиотеки достаточно импортировать один заголовочный файл, который, в свою очередь, будет подключать другие заголовочные файлы. Настраивать зависимости будет система автоматизации сборки проектов CMake. Такой способ подключения является наиболее современным и простым, и именно поэтому многие разработчики придерживаются такого подхода при написании собственных библиотек, фреймворков и модулей.

## 1.3. Состав функций

Программа должна давать пользователю возможность выполнять следующие функции:

* инстанцировать объект http сервера
* базово конфигурировать http сервер
* наследовать класс http сервера под свои нужды
* создавать status line http ответа из готовых шаблонов: определять методы GET и POST, код возвращаемого значения
* создавать заголовки http ответа из готовых шаблонов: content-type, content-length и т. д.
* создавать тела http ответа из готовых шаблонов, отдельных html файлов
* настраивать кастомное логирование с разными уровнями в отдельный файл
* настраивать кастомное логирование с разными уровнями в syslog
* кешировать http ответы
* обрабатывать ошибки
* читать комментарии в коде сервера, которых будет достаточно для использования всех возможностей сервера
* подключать сервер через .hpp файл с отдельной папкой (где будут все остальные файлы-зависимости лежать), настройка необходимых зависимостей идет через готовый CmakeLists.txt

# 2. [Error: Reference source not found](#_Toc72106853) [УСЛОВИЯ ВЫПОЛНЕНИЯ ПРОГРАММЫ](#_Toc72106853)

## 2.1. Минимальный состав аппаратурных средств

Для бесперебойной работы программного продукта требуется компьютер с:

* Объемом свободной встроенной памяти не меньше 100 МБ,
* Объёмом оперативной памяти не меньше 1 ГБ.

## 2.2. Минимальный состав программных средств

Для бесперебойной работы программного продукта требуется компьютер с:

1. Установленной версией компилятора gcc - 14, clang — 3.4
2. Операционной системой со стабильной сборкой, выпущенной не позднее 2015 года
3. Необходимыми библиотеками (все, которые перечислены в include-тегах)

## 2.3. Требования к персоналу (пользователю)

Необходимое количество персонала – 1 человек.

Необходимая квалификация персонала – пользователь.

# 3. ВЫПОЛНЕНИЕ ПРОГРАММЫ

В данном разделе описан пример работы с программой.

## 3.1. Загрузка программы

Для загрузки данного ПО надо скачать файл ServeMe.hpp и обеспечить наличие подключенных к проекту необходимых библиотек (перечислены в include-тегах .hpp-файла).

## 3.2. Запуск программы

Для подключения HTTP-сервера к основной программе необходимо прописать #include «profiler.hpp» (см. рис. 1):



## 3.3. Выполнение программы

## 3.3.1. Создание сервера

Для создания сервера необходимо проинстанцировать объект класса RESTAPIAPP из namespace Utils. Необходимо указать порт, на котором будет в дальнейшем запущен сервер.



**3.3.2. Добавление эндпоинтов к серверу**

Для создания эндпоинтов нужно указать название эндпоинта, ответ в виде raw text или файла (полный путь), а также метод - «GET» или «POST».

**3.3.3. Запуск сервера**

Непосредственный старт сервера:

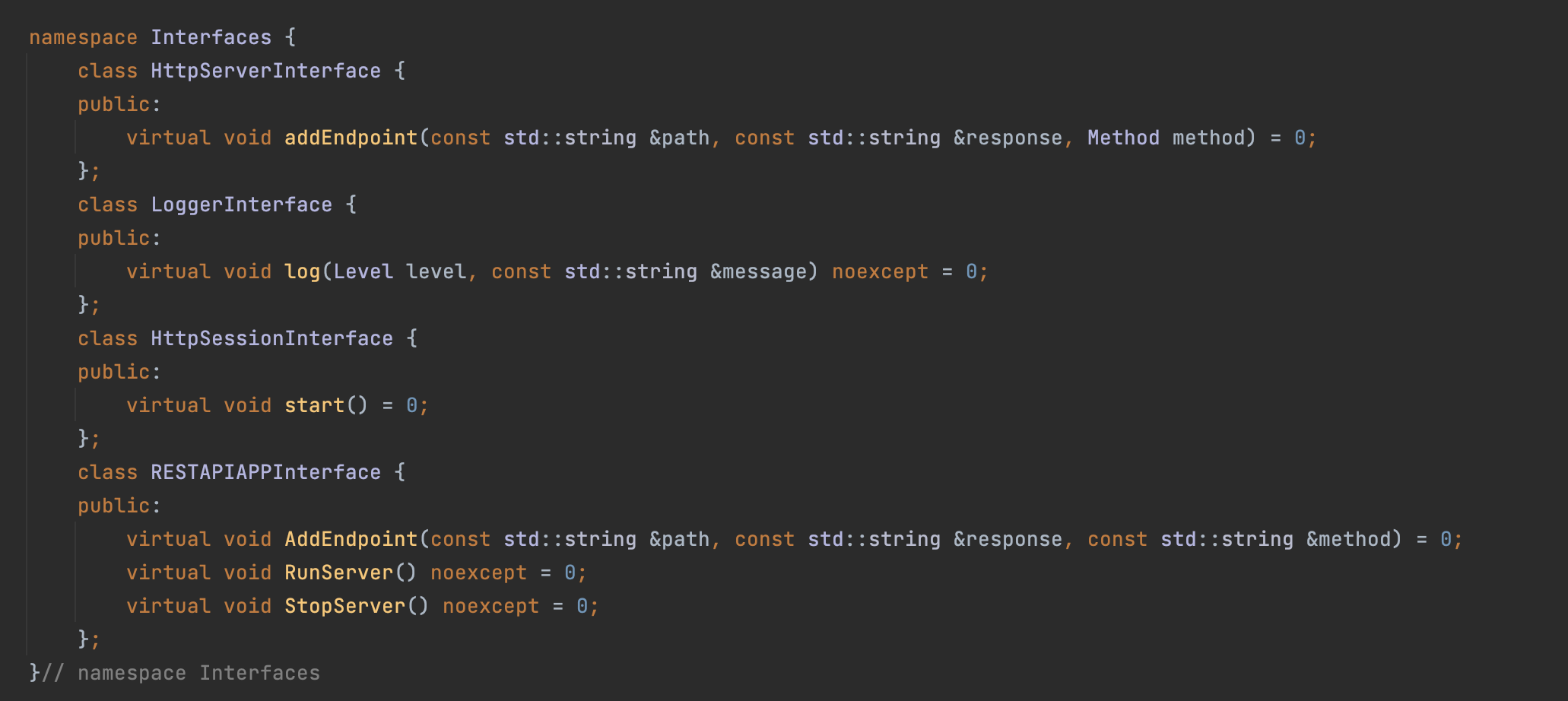


**3.3.4. Остановка сервера (необязательно)**



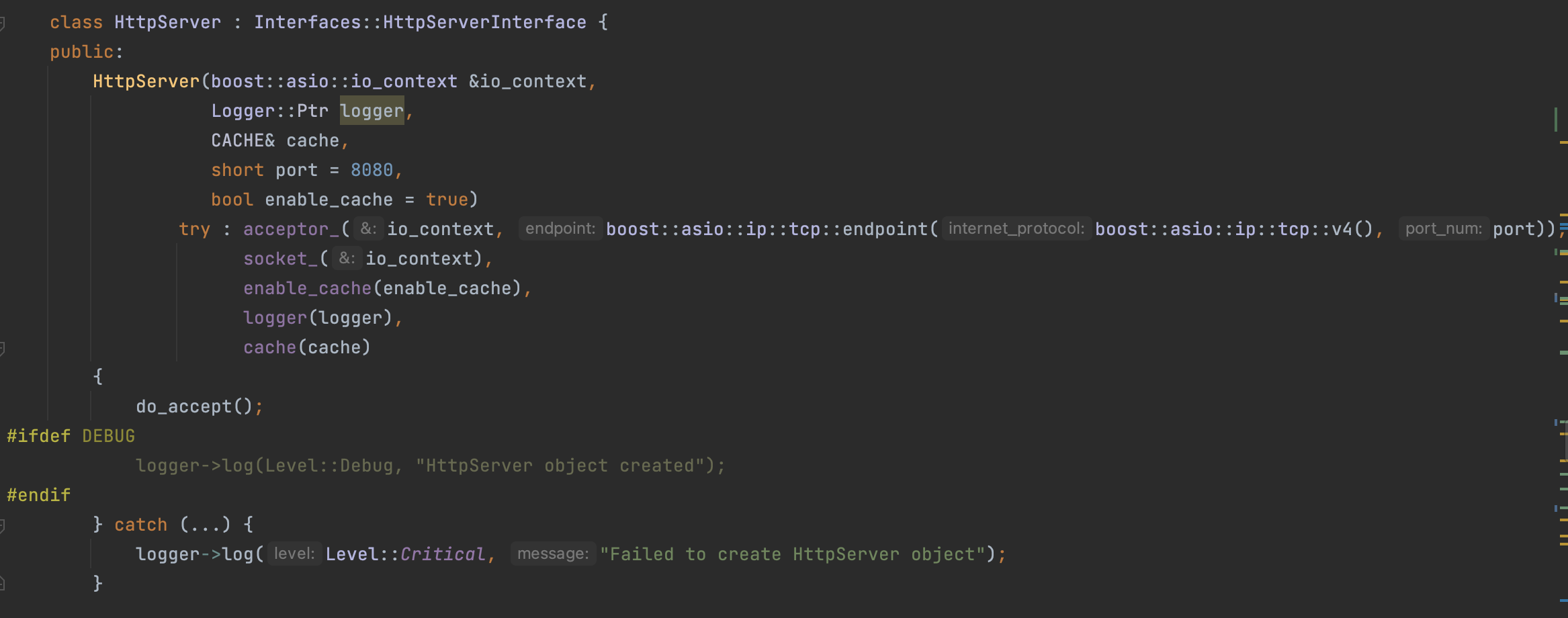
**3.3.5. Наследование отдельных компонентов сервера**

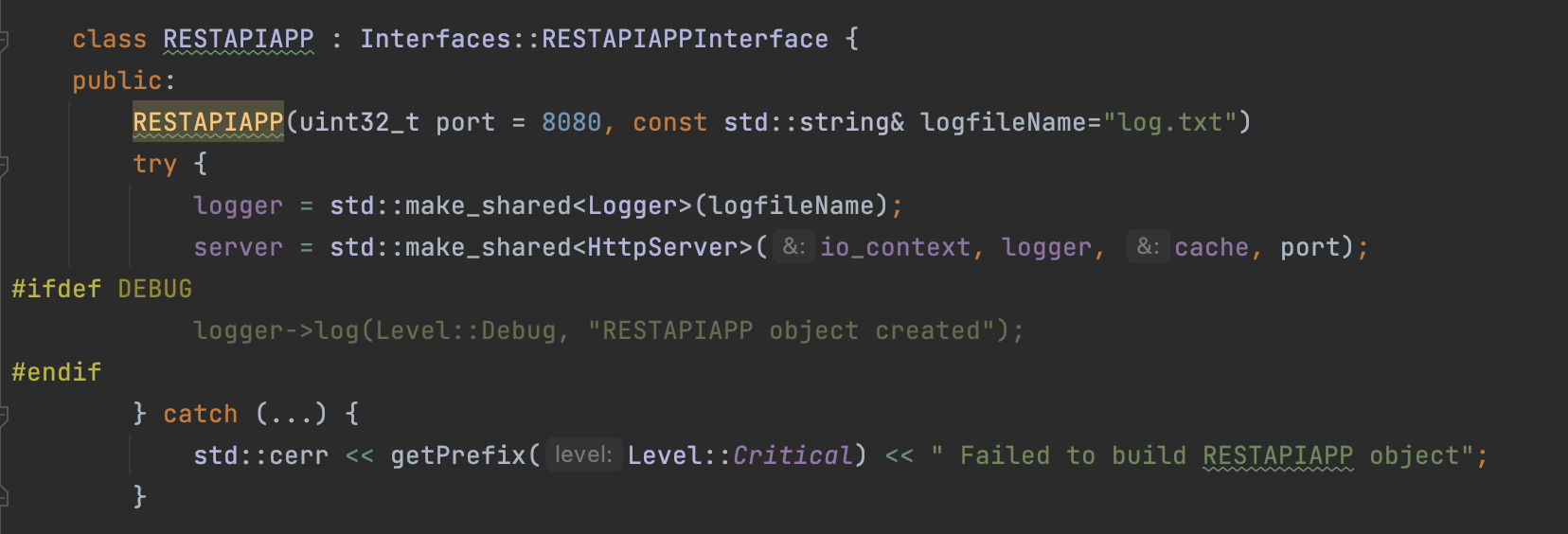
Также, для переиспользования написанного кода, допускается создавать свои собственные классы и наследовать их от сществующих в данном фреймворке. Также можно наследовать интерфейсы и создавать новую функциональность. Так, например, сделано во фреймворке:











# 4. СООБЩЕНИЯ ОПЕРАТОРУ

В программе допускаются исключения и выбросы сигналов, которые могут происходить по причине некорректного пользования программой, а также по любой причине, по которой могут происходить ошибки при работе с используемыми библиотеками (см. документацию используемых библиотек).

Например, ошибка может возникнуть при попытке запустить сервер на занятом порту.

**СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ**

1) Статья про REST API в целом [Электронный ресурс]. Режим доступа: https://habr.com/ru/articles/483202/, свободный. (дата обращения: 9.03.2024)

# ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Изм. | Номера листов (страниц) | | | | Всего листов (страниц) в документе | № документа | Входящий № сопроводительного документа и дата | Подпись | Дата |
| измененных | замененных | новых | аннулированных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |