**БрГТУ**

**Техническая спецификация для приложения “Requests Scheduler”**

Дата: 26ое Июля 2022

Автор: Иван Иваненко

Версия: 0.1

# Контроль Документа

**Последний раз сохранен:** 27/07/2022 18:29

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Changes | Author | Date |
| 0.1 | Создана первая версия документа | Иван Иваненко | Июль, 26, 2022 |

Содержание

[Контроль Документа 2](#_Toc109746424)

[Введение 4](#_Toc109746425)

[Назначение Приложения 5](#_Toc109746426)

[Основные Задачи Сервиса 5](#_Toc109746427)

[Конечные Пользователи Сервиса 5](#_Toc109746428)

[Функциональные Требования 6](#_Toc109746429)

[Системные Требования 7](#_Toc109746430)

# Введение

Данное приложение является частью внутреннего проекта (одним из его вспомогательных сервисов) Брестского Государственного Технического Университета. Документ предназначен для описания технической спецификации данного сервиса.

# Назначение Приложения

Requests Scheduler предназанчен для симуляции создания заявок на городской общественный транспорт. В ходе разработки основного проекта, его невозможно протестировать в реальных условиях, поэтому было принято решение создать данный сервис.

## Основные Задачи Сервиса

1. Симуляция создания заявок конечными пользователями.
2. Тестирование оснвоного приложения.
3. Использование сервиса для имитации конечного пользователя в ходе демонстрации основного продукта заказчику.

## Конечные Пользователи Сервиса

Сервис предназначен для внутреннего использования в ходе разработки основного продукта.

# Функциональные Требования

Функциональные Требования следующие:

1. Сервис должен работать круглосуточно.
2. Сервис должен генерировать реалистичное количество заявок в зависимости от текущего времени дня.
3. Должна быть возможность конфигурации интенсивности заявок для каждого периода суток. Данная конфигурация должна происходить в отдельном файле конфиуграции.
4. Должна быть возможность конфигурации числа остановок. Количество остановок соответствует количеству потоков сервиса. Должна быть возможность определить количество суточных заявок для каждой остановки.
5. Должна быть возможность быстрой генерации всех заявок для временного периода равного одним суткам. Данная возможность должна быть реализована в виде отдельного приложения.
6. Все сгенерированные заявки должны отправляться на сетевой адрес, указанный в файле конфигурации.
7. Должно быть реализовано приложение, которое считывает все отправленные заявки и заполняет excel файл.
8. Сервис для обработки заявок должен иметь следующие методы:
   1. Метод для обработки одного или нескольких запросов. Полученные запросы должны добавляться в базу данных и в очередь сообщений.
   2. Метод для получения всех запросов.
   3. В дальнейшем список методов может расширяться в зависимости от новых требований.

# Системные Требования

Системные Требования следующие:

1. Сервис должен быть кроссплатформенным. На Unix системах он должен разворачиваться в виде daemon. На системе Windows сервис должен разворачиваться в виде Windows Service.
2. Все вспомогательные приложения для генерации суточных заявок и генерации excel файла также должны быть кроссплатформенными и разворачиваться в виде консольного приложения.
3. Сервис для обработки всех заявок должен разворачиваться в виде REST Web API и иметь единый интерфейс для всех клиентов (Requests Scheduler, консольные приложения). Сервис должен имиеть следующие методы:
   1. Метод для обработки одного или нескольких запросов.
   2. Метод для получения всех запросов.
   3. В дальнейшем список методов может расширяться в зависимости от новых требований.