

Проектирование теста

Сценарии:

1 Проверка разбора ПКС

1.1 Передача параметра -h (позитивный)

1.2 Передача всех корректных параметров командной строки (позитивный)

1.3 Передача неверного пути к файлу с базой данных с помощью параметра -b (негативный)

1.4 Передача неверного пути для файла с логами с помощью параметра -l (негативный)

1.5 Передача корректного порта с помощью параметра -p (негативный)

1.6 Передача недопустимого параметра командной строки (негативный)

2 Проверка стабильности работы сервера

2.1 Запуск 1000 корректных клиентов

2.2 Запуск 1000 клиентов с другим типом данных

Сценарии тестирования разбора ПКС

Сценарий	Тест	Входные данные	Ожидаемый результат	Полученный результат
1.1	1.1.1	-h	Вывод справки	Вывод справки
1.2	1.2.1	-p=7777 -l=log_test.txt -b = base.txt	Без исключений	Без исключений
1.3	1.3.1	-b = basepsu.txt	Исключение crit_err	Исключение crit_err
1.4	1.4.1	-l = log/log/log.txt	Исключение crit_err	Исключение crit_err
1.5	1.5.1	-p=7777	Без исключений	Без исключений
1.6	1.6.1	-f	Исключение boost::program_o ptions::error	Исключение boost::program_o ptions::error

Сценарии тестирования работы функций программы

Сценарий	Тест	Входные данные	Ожидаемый результат	Полученный результат
2.1	2.1.1	нет	Стабильная работа сервера, запись всех действий сервера в журнал	Стабильная работа сервера, запись всех действий сервера в журнал
2.2	2.2.1	нет	Стабильная работа сервера, запись всех действий сервера в журнал	Стабильная работа сервера, запись всех действий сервера в журнал

Примечания:

Реализация сценария 2.1 и сценария 2.2 находится в приложении А

Документация находится в репозитории:

https://github.com/D4nilkaPolekhin/Kurs_Server

Приложение А

(обязательное)

Реализация теста 2.1:

```
stud@virtdeb:~$ for run in range {0..1000};do ./client int16 t; done
```

Рисунок 1 - Цикл запуска клиента в консоли Linux

Реализация теста 2.2:

```
stud@virtdeb:~$ for run in range {0..1000};do ./client double; done
```

Рисунок 2 - Цикл запуска клиента в консоли Linux