

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Факультет программной инженерии и компьютерных технологий

Системное программное обеспечение

Лабораторная работа № 3

Выполнила
студентка:

Подсекина Мария Викторовна

Группа: № Р33122

г. Санкт-Петербург

2021

Цель: изучение способов взаимодействия между сетевыми службами низкого уровня в асинхронном режиме.

Описание работы:

Разработать клиент-серверное приложение. Для организации взаимодействия по сети, поддержки множества соединений использовать программные интерфейсы (API) операционной системы.

Сервер и клиенты взаимодействуют по протоколу, реализованному на базе сокетов (использовать TCP, если в варианте задания не указано обратное). Сервер должен поддерживать условно неограниченное количество клиентов. На всех этапах взаимодействия клиента и сервера должна быть предусмотрена обработка данных независимо от их размера.

Порядок выполнения:

1. Выполнить анализ предметной области, которая задается вариантом к лабораторной работе. Результатом анализа должен быть набор сущностей, которые будут в качестве элементов данных (типов, структур, операций) и/или составных частей программной архитектуры (компонентов, модулей) при реализации программы.
2. Составить диаграмму, на которой схематически будут показаны результаты анализа: сущности, их атрибуты и взаимосвязи.
3. Составить план постепенного выполнения задания: какие части функциональности, в каком порядке предполагается реализовывать, в каком порядке и как проверять их работоспособность.
4. Загрузить все полученные артефакты в отдельную директорию «docs» репозитория, в корневую директорию положить readme.md с номером варианта и кратким описанием.
5. Продемонстрировать составленные диаграмму и план преподавателю. Для этого достаточно просто отправить преподавателю ссылку на репозиторий.
6. После проверки и получения рекомендаций приступить к реализации программы, создавая на каждый этап выполнения отдельную ветку в репозитории.
7. По завершении каждого этапа создать pull-request, включив преподавателя в число reviewer-ов.
8. Если pull-request отклоняется преподавателем, выполнить необходимые правки и обновить его, запросив повторное ревью.
9. Когда pull-request одобрен, «слить» его с основной веткой кода, после чего создать новую ветку для работы над следующим этапом.

Описание предметной области по варианту:

Список задач (Task-list)

Программа может выполняться в двух режимах: сервер или клиент. Режим определяется аргументом командной строки. Завершение программы-сервера происходит по нажатию ключевой клавиши (например, Q).

При запуске в режиме клиента через аргументы командной строки задаётся имя пользователя и адрес сервера.

Каждый подключающийся пользователь имеет свой, отдельный от других пользователей, набор списков задач. Каждый список задач описывается названием и состоит из задач, каждая задача представлена датой создания, кратким заголовком, описанием и опциональным планируемым временем завершения. Пользователь может создавать/уничтожать/редактировать листы задач и задачи в них. Повторные подключения того же пользователя должны так же отображать актуальное состояние его списков задач. Переключение между областями ввода по Tab, применение введенного значения по Enter.

Исходный код: <https://github.com/marypodsekina/lab3-spo>

Вывод: в ходе выполнения данной лабораторной работы мною было создано клиент-серверное приложение, на клиентской стороне которого был реализован графический интерфейс для взаимодействия со списками и задачами в них. Мое приложение также было протестировано на асинхронное взаимодействие нескольких подключенных клиентов. Тесты прошли успешно.