

Федеральное государственное автономное образовательное учреждение высшего
образования «Национальный исследовательский университет ИТМО»

Лабораторная работа №3
«Удаленное взаимодействие»

Выполнил: Прикота Виталий Александрович
Преподаватель: Кореньков Юрий Дмитриевич

Санкт-Петербург, 2021

1 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:

- 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.
- Составить диаграмму, на которой схематически будут показаны результаты анализа: сущности, их атрибуты и взаимосвязи.
- Составить план постепенного выполнения задания: какие части функциональности, в каком порядке предполагается реализовывать, в каком порядке и как проверять их работоспособность.
- Загрузить все полученные артефакты в отдельную директорию «docs» репозитория, в корневую директорию положить readme.md с номером варианта и кратким описанием.
- Продемонстрировать составленные диаграмму и план преподавателю. Для этого достаточно просто отправить преподавателю ссылку на репозиторий.
- После проверки и получения рекомендаций приступить к реализации программы, создавая на каждый этап выполнения отдельную ветку в репозитории.
- По завершении каждого этапа создать pull-request, включив преподавателя в число reviewer-ов.
- Если pull-request отклоняется преподавателем, выполнить необходимые правки и обновить его, запросив повторное ревью.
- Когда pull-request одобрен, «слить» его с основной веткой кода, после чего создать новую ветку для работы над следующим этапом.

2 Вариант 6: Library catalog

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. Каждая книга описывается карточкой, содержащей сведения о названии, авторе, дате издания, аннотации, и наборе тэгов (таких как жанры и т.п.), количество приписанных к каталогу и количество в наличии. Каждый подключающийся пользователь может помещать книги в каталог, брать на прочтение и возвращать. Предусмотреть поиск сведений о книгах по различным критериям. Если описание книги редактируется или изменяется статус наличия, пользователям должна своевременно показываться актуальная информация по мере её изменения. Визуально любые операции редактирования осуществляются «in-place», то есть по месту визуализации данных поля.

3 Исходный код

<https://github.com/rbetik12/FlexLibrary>

4 Вывод

В ходе выполнения данной лабораторной работы я познакомился с тем, как можно организовать сетевое взаимодействие процессов, используя язык Си. Также разработал каталог, который с помощью текстового интерфейса позволяет просматривать книги, а также редактировать их описание, синхронизируя состояние между клиентами.