Муниципальное бюджетное общеобразовательное учреждение

города Новосибирска «Лицей № 136 имени Героя Российской Федерации

Сидорова Романа Викторовича»

Образовательная область: технология

Предмет: информатика

Создание сервера и клиента для распределённых вычислений

Исполнитель: Завёрткин Михаил Александрович,

11 класс

Руководитель: Валюхова Светлана Викторовна,

учитель информатики высшей категории

г. Новосибирск, 2022 г.

Оглавление

[Введение 3](#_Toc94957726)

[Актуальность 3](#_Toc94957727)

[Цель 3](#_Toc94957728)

[Задачи 3](#_Toc94957729)

[1. Первая программа на PascalABC.NET 4](#_Toc94957730)

[2. Разрабатываемая версия 4](#_Toc94957731)

[2.1. Вспомогательные инструменты 4](#_Toc94957732)

[2.2. Архитектура 5](#_Toc94957733)

[2.2.1. Сервер 5](#_Toc94957734)

[2.2.2. Клиент 6](#_Toc94957735)

[Заключение 7](#_Toc94957736)

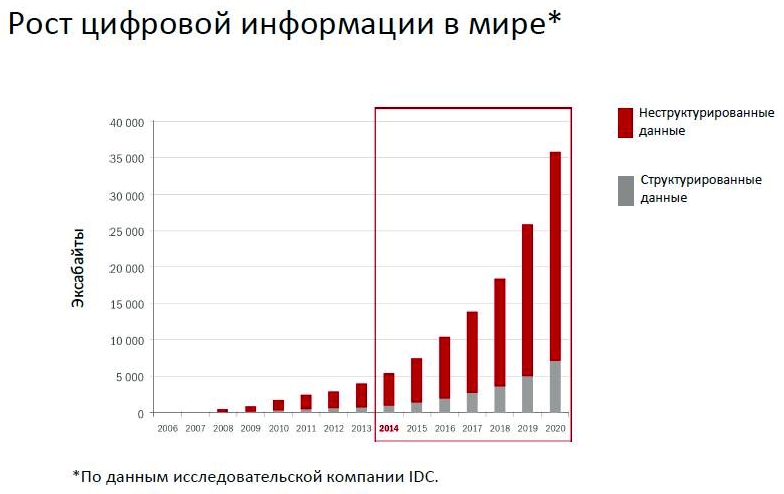
[Список использованной литературы 8](#_Toc94957737)

[Приложения 9](#_Toc94957738)

# Введение

## Актуальность

В наши дни количество получаемой информации всё увеличивается, и эту информацию надо обрабатывать. Делать это всё труднее по мере усложнения алгоритмов обработки и увеличения наборов данных. Один из методов повышения эффективности вычислений – это их распределение. А учитывая тенденцию к росту сбора информации, отрасль её обработки не останется невостребованной.



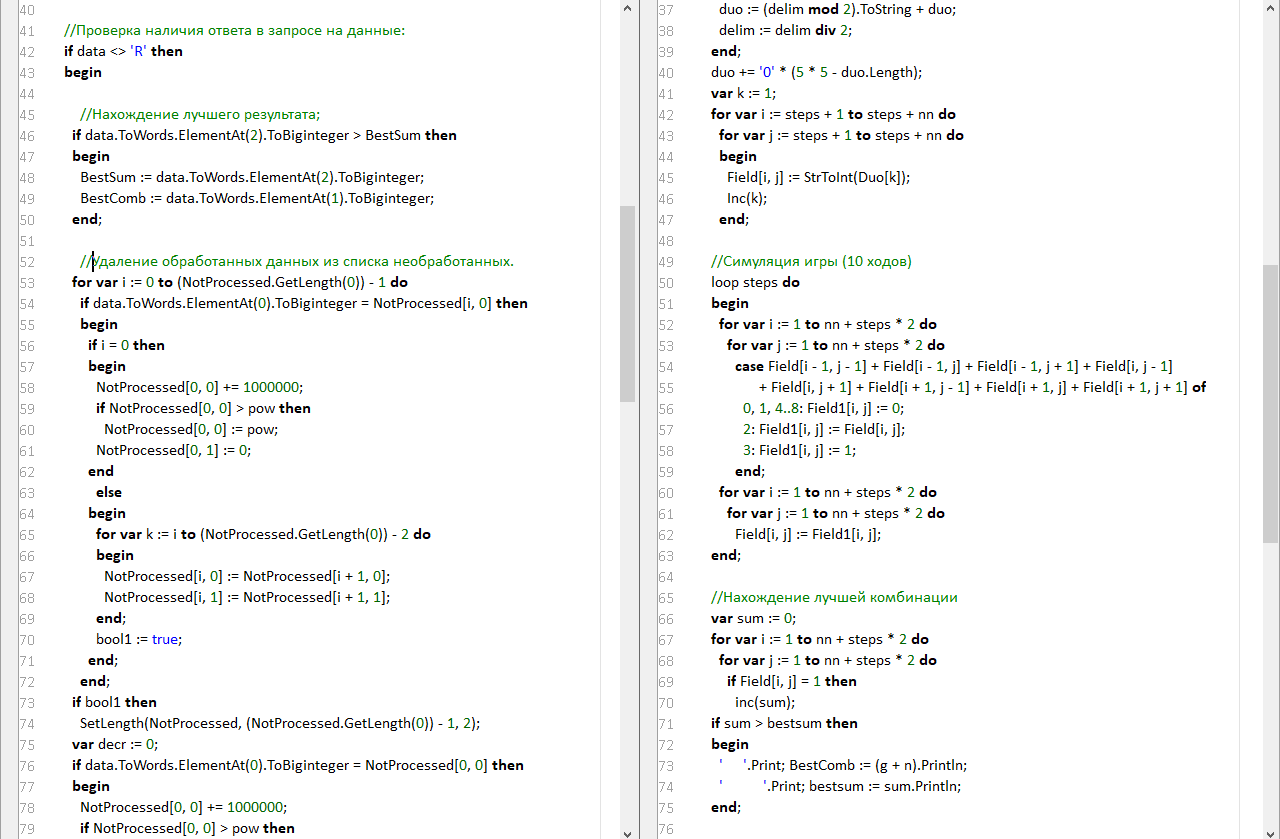
## Цель

Написать приложение для проведения распределённых вычислений с клиент-серверной архитектурой

## Задачи

* Продумать структуру проекта
* Написать реализацию
* Отладить и исправить ошибки

# Первая программа на PascalABC.NET

Несколько лет назад я уже писал аналогичную программу, но на языке программирования PascalABC.NET. Это была программа в функциональном стиле, что сильно снижало её гибкость. 

Исходники можно просмотреть по следующей [ссылке](https://disk.yandex.ru/d/mdxvyt-SWMIwzg).

# Разрабатываемая версия

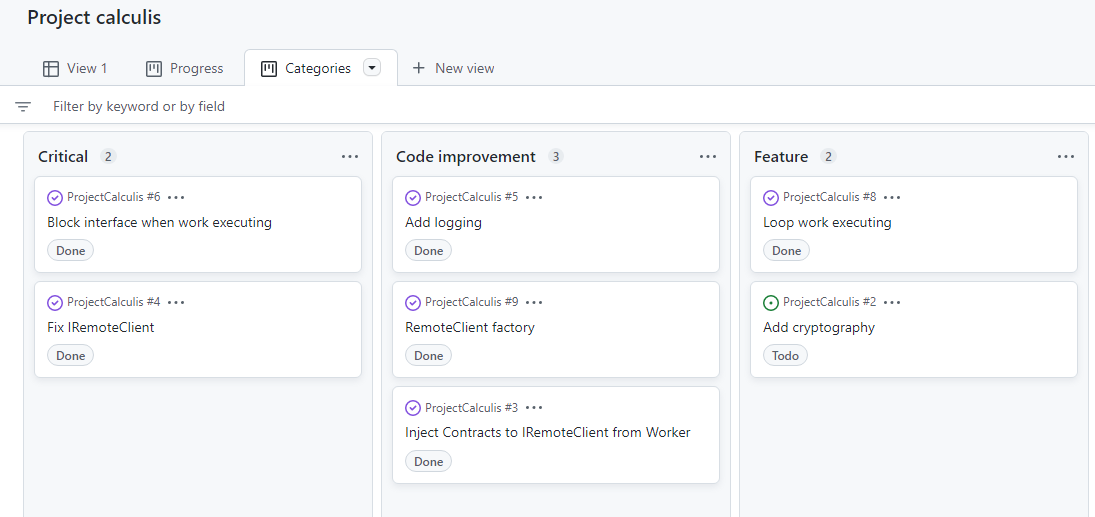
Сейчас мною разрабатывается новое приложение, нацеленное на гибкость кодовой базы. Она достигается путём:

* использования ООП, в частности интерфейсов и абстракций;
* применения инъекции зависимостей с помощью стандартного DI-контейнера;
* использования структурных паттернов, таких как MVVM, Factory.

Внедрение логирования позволяет упростить процесс отладки.

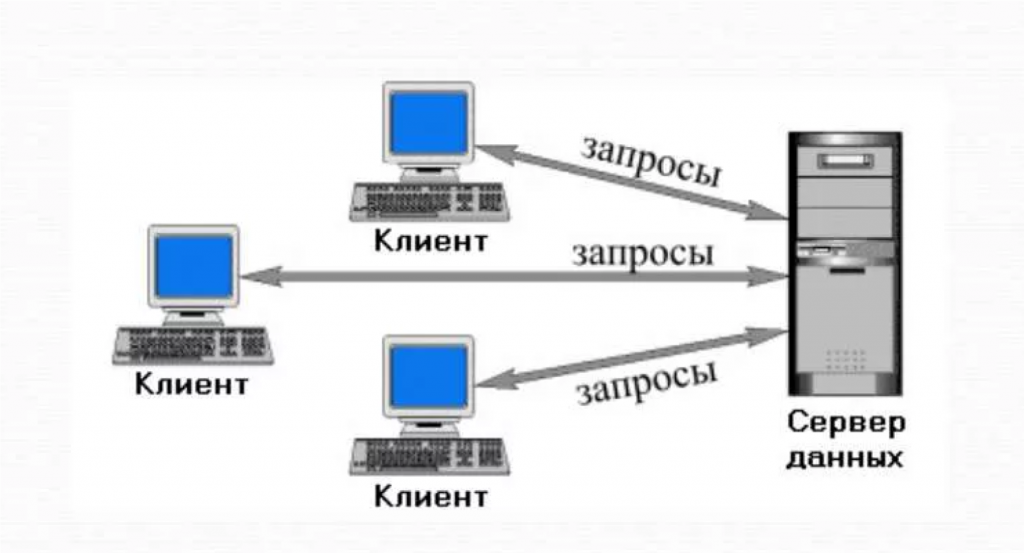
## Вспомогательные инструменты

Для планирования о организации работы широко использовались возможности GitHub, например, проекты.

Проекты позволяют группировать задачи и структурировать их выполнение. 

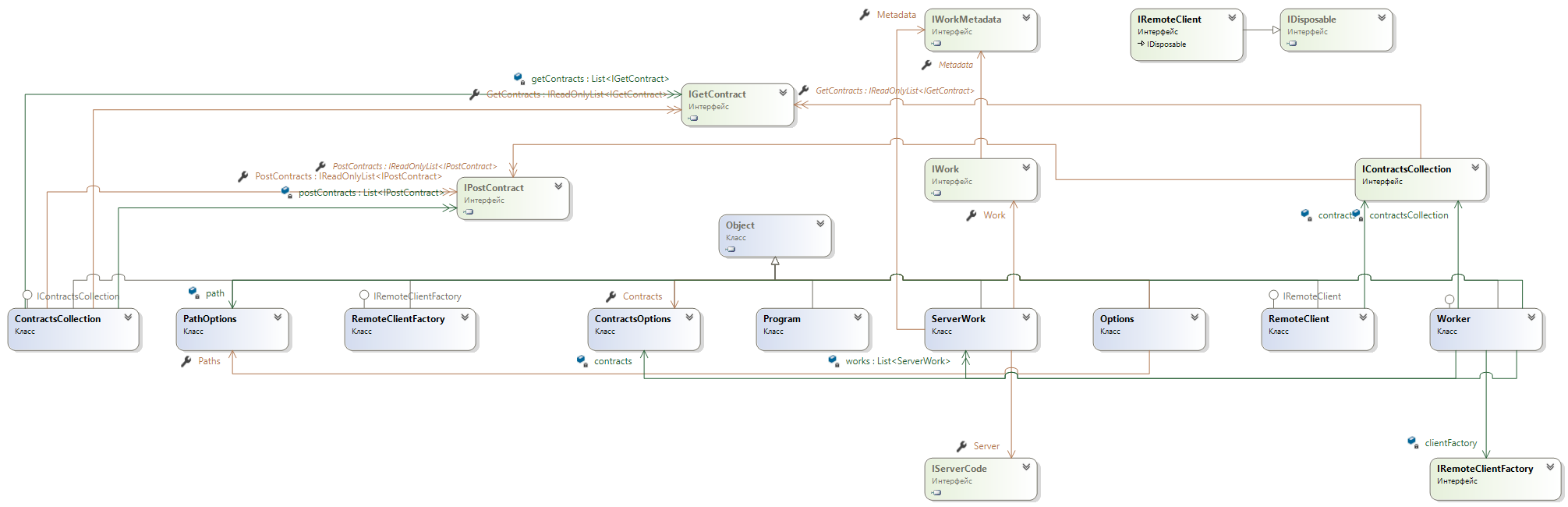
## Архитектура

Проект имеет клиент-серверную архитектуру, что означает наличие двух частей – сервера и клиента. Сервер координирует выполнение задач клиентами, которые используют вычислительные мощности локальных машин



### Сервер

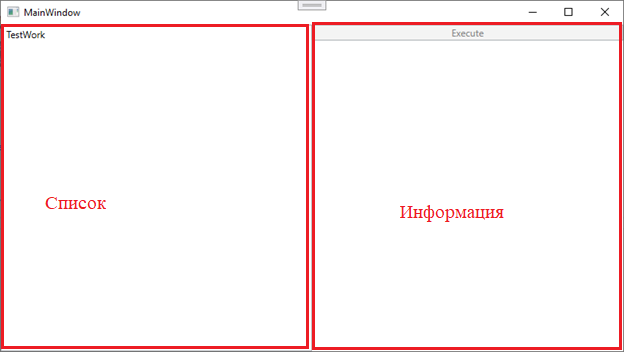
После запуска сервер находится в состоянии прослушивания входящих подключений от клиента. При появлении такого подключения запросы обслуживаются до момента отключения.



### Клиент

Клиент построен с применением паттерна MVVM, что позволяет повысить гибкость приложения.   
После запуска пользователь может выбрать из списка желаемую задачу для выполнения.

Интерфейс разбит на несколько визуальных частей – список задач и подробная информация о выбранной задаче.



# Заключение

На данный момент имеется гибкий и простой для изменения рабочий прототип с большим потенциалом для улучшения. Программу можно применять для проведения распределённых вычислений в клиент-серверной манере.

# Список использованной литературы

[Metanit](https://metanit.com/sharp/)

[Документация по C# и .NET](https://docs.microsoft.com/ru-ru/dotnet/csharp/)

Форумы – [StackOverflow](https://stackoverflow.com/) и [Cyberforum](https://www.cyberforum.ru/net-framework/)

Язык программирования C# 7 и платформы .NET и .NET Core - Джепикс Филипп, Троелсен Эндрю

# Приложения

[Репозиторий проекта](https://github.com/ProMix0/ProjectCalculis)