**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ  
ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО**

Факультет автоматизированных и информационных систем

Кафедра «Информатика»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2,3,4  
по дисциплине **«Интернет-технологии и распределенная обработка данных»**

на тему: **«Различные способы интеграции корпоративных приложений»**

Выполнил: студент группы ИП-41  
Бредихин К.А.  
Принял: ст. преподаватель  
Шпетный Е.А.

Дата сдачи отчета:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Гомель 2019

**Цель работы**: Изучить различные способы интеграции корпоративных приложений.

## Описание задачи

Необходимо реализовать клиент-серверный многопользовательский чат, который позволяет клиентам с  помощью командной строки обмениваться сообщениями. Каждая лабораторная работа будет представлять разный способ интеграции между клиентами и сервером.

Приложение должно поддерживать :

1. Подключения нескольких клиентов к серверу.
2. Логирование всех действий клиентов и ошибок.

Содержание каждого задания приведено ниже. Само основное назначение приложения и задач, которое оно решает остается неизменным. Меняется лишь способ взаимодействия, хранения и обмена данными.

### Задание 1

Необходимо реализовать интеграцию на основе обмена файлами (JSON, XML или CSV), тип и формат файла студент может выбрать самостоятельно.

### Задание 2

В лабораторной работе 2 реализация интеграции должна осуществляться на основе общей базы данных. База данных может быть как SQL системой, такой как MySQL, SQL Server, PostgreSQL и т.д. Или NoSQL системой MongoDB, Cassandra и т.д.

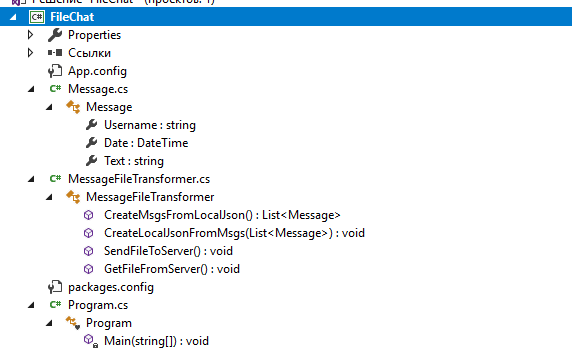
### Задание 3

Лабораторная работа 3 будет посвящена интеграции клиента и сервера через брокер сообщений (Kafka, RabbitMQ или ActiveMQ). Формат сообщений может быть произвольно выбран в зависимости от реализации приложений.

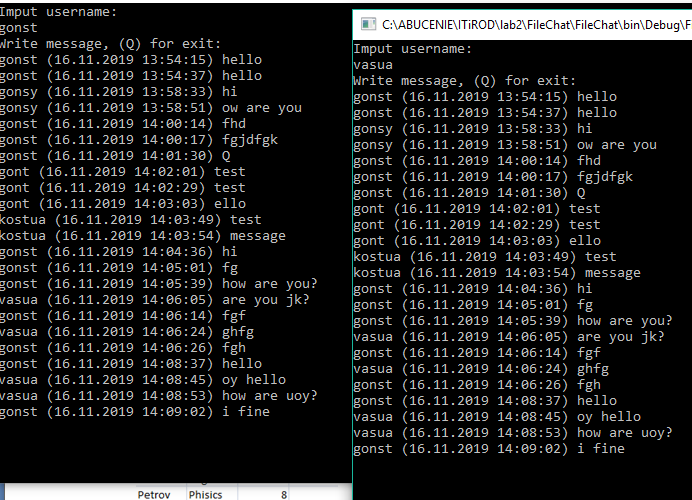
# Описание проделанной работы

1. К заданию 1 было разработано консольное приложение использующее FTP сервер для передачи файлов JSON с сообщениями между пользователями, в него добавлено логирование при помощи Serilog.

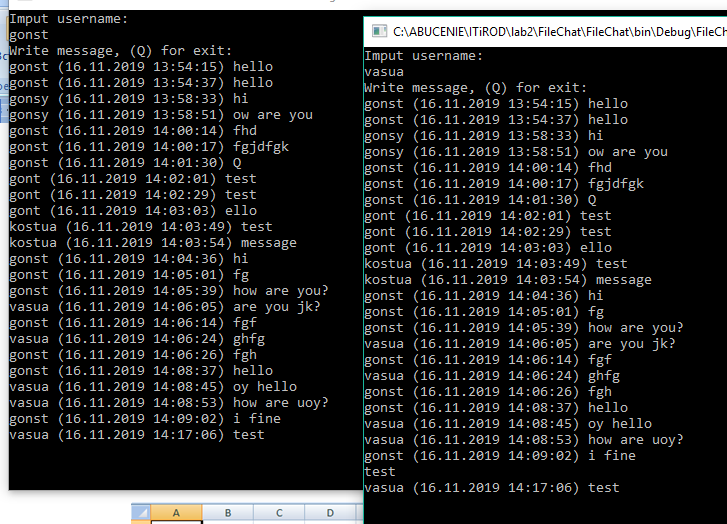
**Структура классов**

****

# Верификация и тестирование



При вводе сообщения отображаются у всех остальных пользователей:



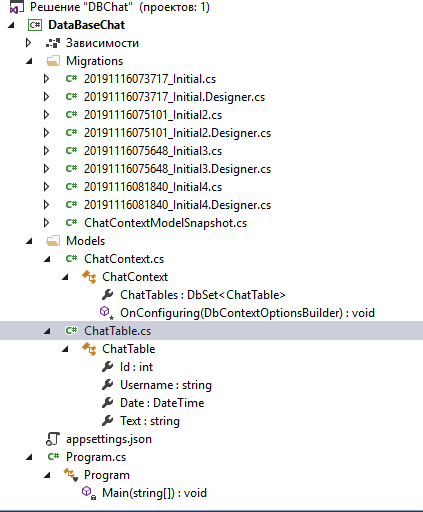
Данные отображаются корректно.

Ссылка на Github:

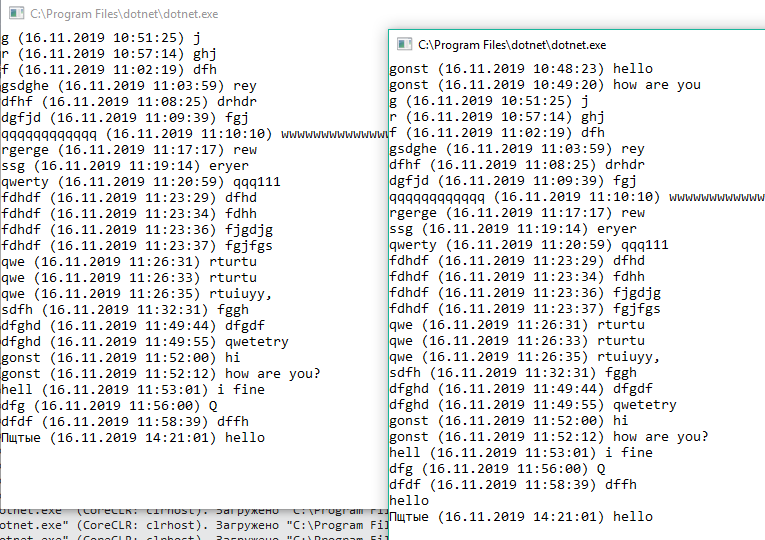
<https://github.com/KonstantinBr/ItiRod>

1. К заданию 2 было разработано консольное приложение использующее MS SQL базу данных (общую) для передачи сообщений между пользователями, в него добавлено логирование при помощи Serilog.

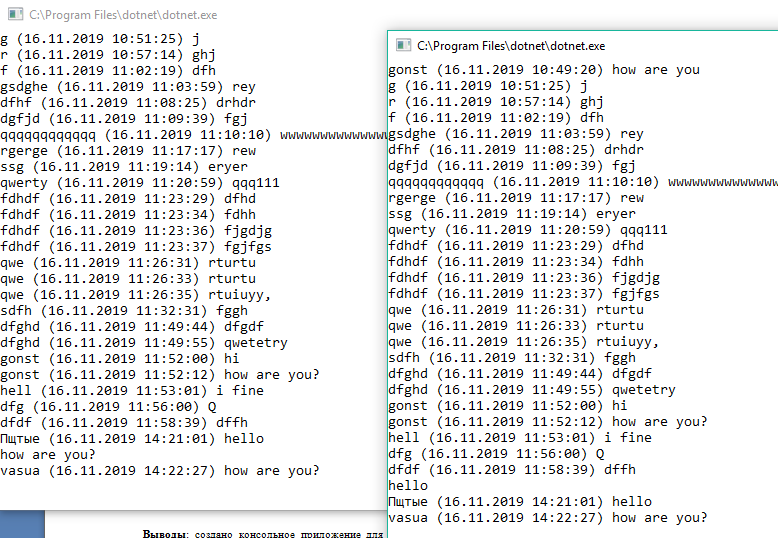
**Структура классов**



# Верификация и тестирование



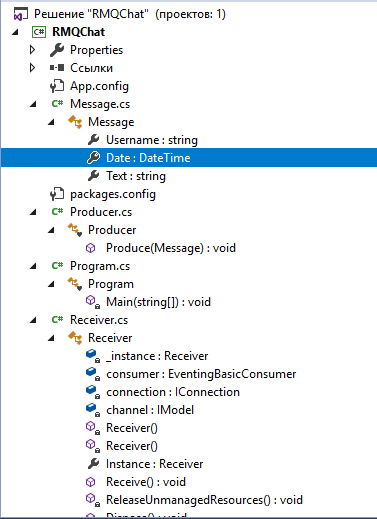
При вводе сообщения отображаются у всех остальных пользователей



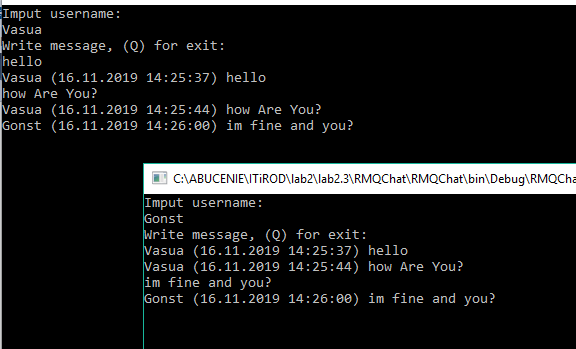
Данные отображаются корректно.

1. К заданию 3 было разработано консольное приложение брокер сообщений RabbitMQ для передачи сообщений между пользователями, в него добавлено логирование при помощи Serilog.

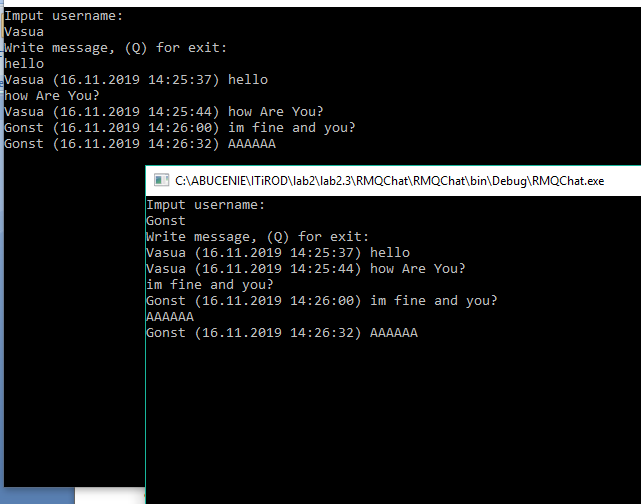
**Структура классов**



# Верификация и тестирование

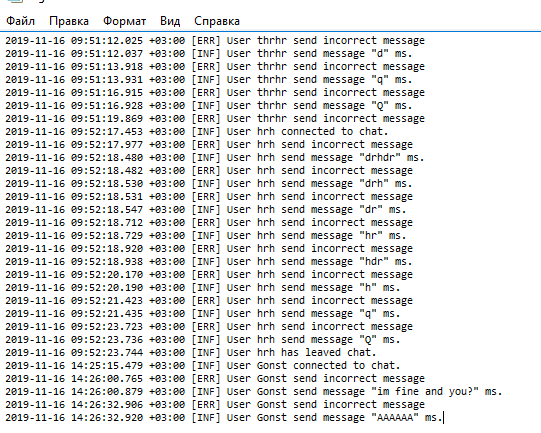


При вводе сообщения отображаются у всех остальных пользователей



Данные отображаются корректно.

Файл логов имеет следующий вид:



Ссылка на Github:

<https://github.com/KonstantinBr/ItiRod>

**Выводы**: изучены различные способы интеграции корпоративных приложений.