НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

ФАКУЛЬТЕТ «БИЗНЕС-ИНФОРМАТИКА»

***Командный проект   
по дисциплине   
«Программирование»***

***«Currency Rate Monitor»***

Выполнили: студенты 153 (2) группы

Фофина Е. С.

Пайсон А. Д.

Адрес репозиторiя git:

https://github.com/artempayson/Currency-rate-monitor.git

Москва,

2016 год.

Оглавление

[***Командный проект по дисциплине «Программирование»*** 1](#_Toc469876248)

[*1. Тема проекта и название программы* 3](#_Toc469876254)

[*2. Краткое описание программы и её функционала* 3](#_Toc469876255)

[*3. Адрес репозитория GIT:* 4](#_Toc469876256)

[*4. Члены команды и их роли:* 4](#_Toc469876257)

[*5. Список классов:* 4](#_Toc469876258)

[*6. Интерфейс программы:* 6](#_Toc469876259)

## *1. Тема проекта и название программы*

**Тема проекта:** мониторинг курса валютных пар на бирже биткоин.

**Название программы**: Currency Rate Monitor

## *2. Краткое описание программы и её функционала*

Наша программа обращается к публичному API самой большой онлайн биржи биткоин [BTC-E](https://btc-e.nz/)

Оттуда в режиме реального времени (с частотой обновления 2 Гц) загружаются последние данные о валютной паре

1. Данные, имеющие реализацию в модели и используемые в ней
   1. **high**: макcимальная цена.
   2. **low**: минимальная цена.
   3. **buy**: цена покупки.
   4. **sell**: цена продажи.
   5. **hidden**: скрыта ли пара, значение 0 или 1.
2. Данные, имеющие реализацию в модели, но не нашедшие отражения в функционале
   1. **avg**: средняя цена.
   2. **vol**: объем торгов.
   3. **vol\_cur**: объем торгов в валюте.
   4. **last**: цена последней сделки.
3. Данные, не имеющие явной реализации в модели, но имеющие подготовленную платформу для оперативного внедрения в программу
   1. **decimal\_places**: количество знаков после запятой разрешенные при торгах.
   2. **min\_price**: минимальная цена разрешенная при торгах.
   3. **max\_price**: максимальная цена разрешенная при торгах.
   4. **min\_amount**: минимальное количество разрешенное для покупки/продажи.
   5. **fee**: комиссия пары.

Затем, по полученным данным в реальном времени строится два графика – цены покупки[[1]](#footnote-1) и цены продажи. Функционал программы также позволяет сохранять наблюдаемые колебания курса в json файл. Соответственно также присутствует и функция загрузки данных из сохранений, сделанных этой программой.

## *3. Адрес репозитория GIT:*

<https://github.com/artempayson/Currency-rate-monitor.git>

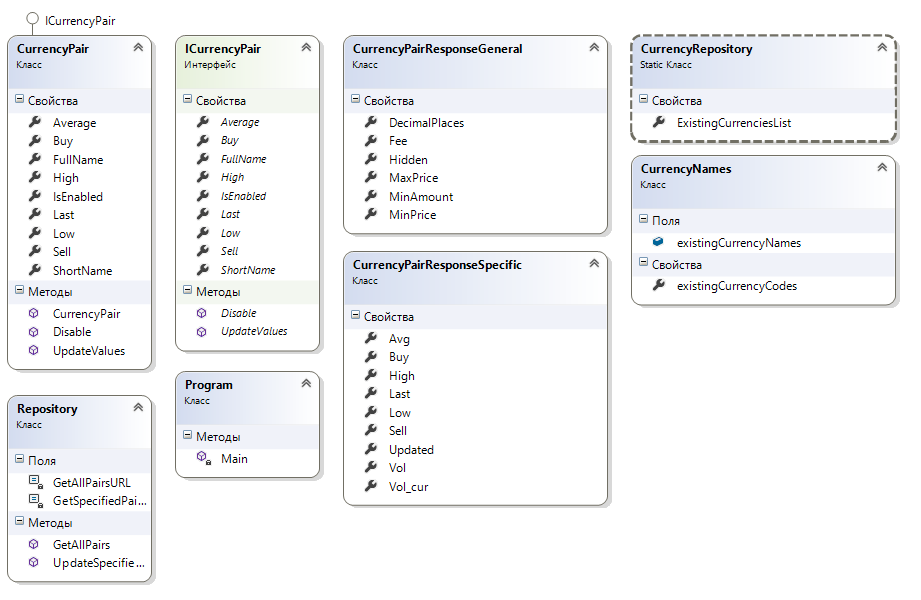
## *4. Члены команды и их роли:*

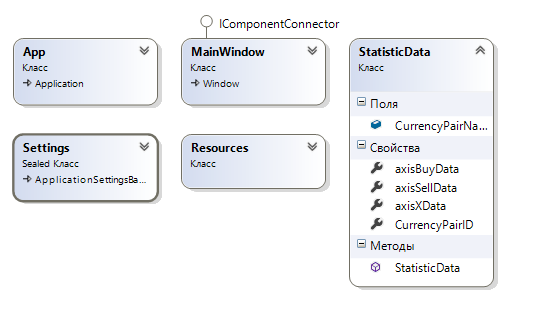
Екатерина Фофина 153(2) отвечает за взаимодействие программы с пользователем

Артем Пайсон 153(2) отвечает за основную логику, работу с API btc-e и функционал сохранения-загрузки колебаний курса, как и за возникающие в связи с этим исключения

## *5. Список классов:*

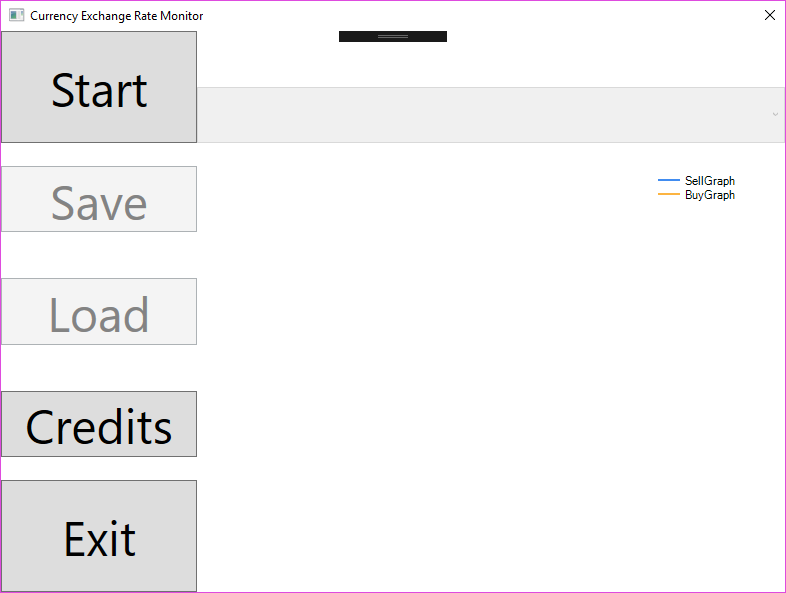
1. Logic часть проекта:
   1. Model
      1. Existing Currencies
         1. CurrencyNames  
            Класс, отвечающий за хранение списка имен и кодов[[2]](#footnote-2) валютных пар
         2. CurrencyRepository  
            Статический класс, отвечающий за создание программного списка валютных пар из имеющихся имен
         3. CurrencyPair  
            Основной класс модели валютной пары btc-e, воплощающий интерфейс ICurrencyPair, предназначенный для взаимодействия с остальными частями проекта
   2. DTO
      1. CurrencyPairResponseGeneral  
         Класс, предназначенный для обработки ответа с сервера, отвечающего за общую информацию по торгам о паре. В основном, эти данные не нашли отражения в модели, но, возможно, в будущем будут использованы
      2. CurrencyPairResponseSpecific   
         Класс, предназначенный для обработки ответа с сервера, отвечающего за конкретную информацию по торгам о паре.
   3. Repository  
      Класс репозитория, предназначенный для взаимодействия логической части с пользовательским интерфейсом
   4. Program  
      В пояснениях не нуждается
2. GUI часть проекта:
   1. MainWindow  
      Основное окно программы
   2. StatisticData  
      Класс, отвечающий за формат сохранения и загрузки данных в json



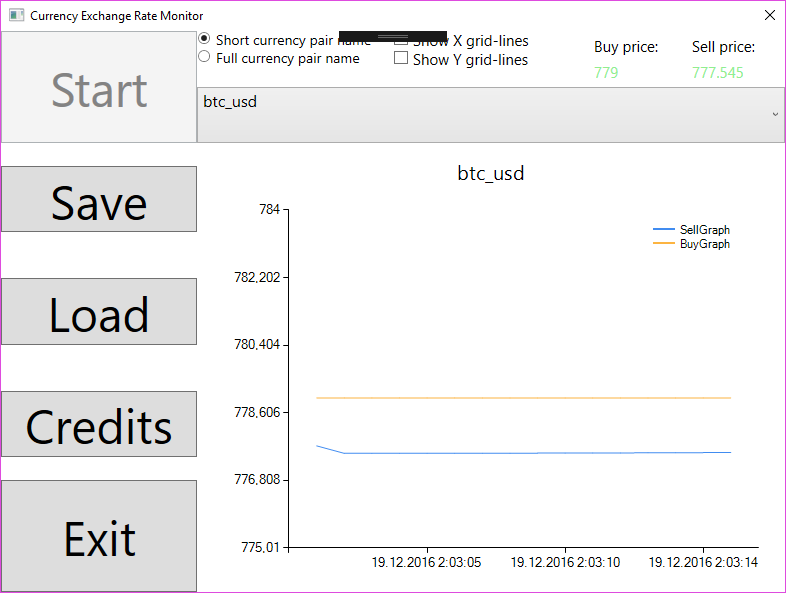


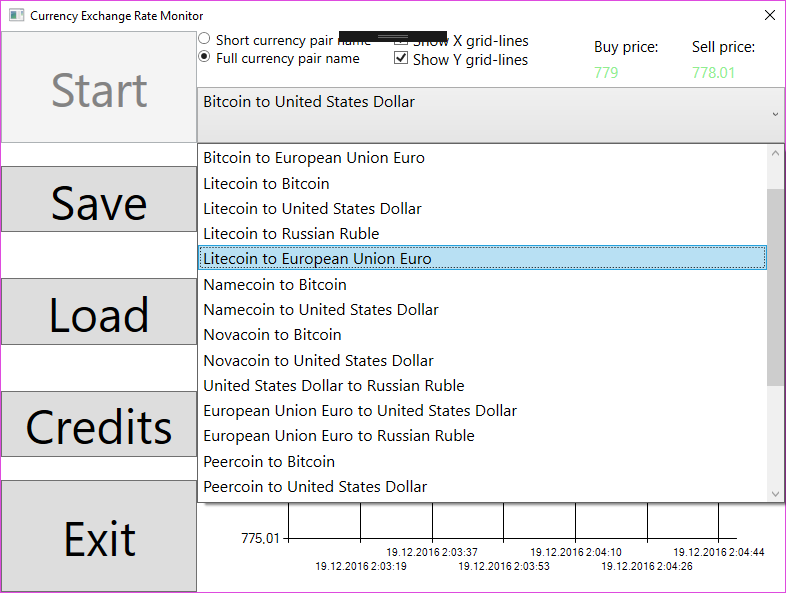
## *6. Интерфейс программы:*

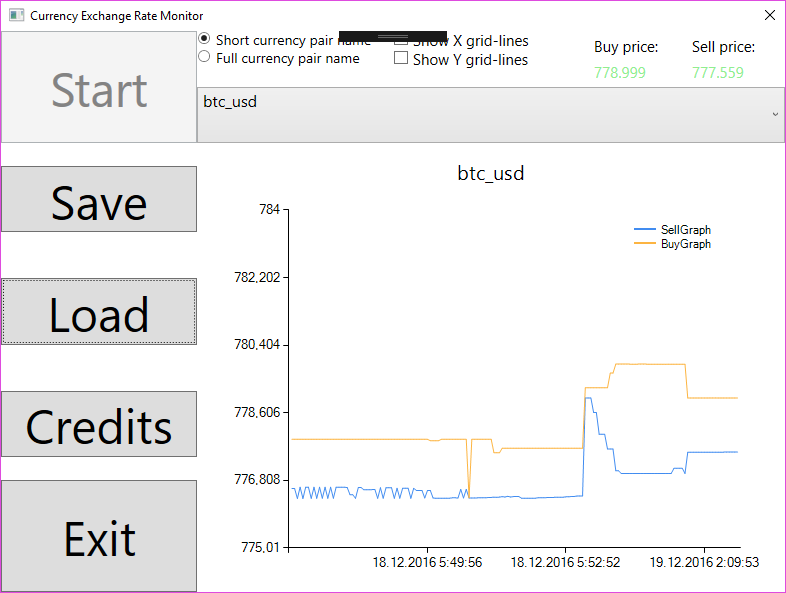
1) Окно при запуске

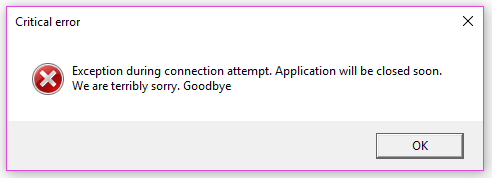


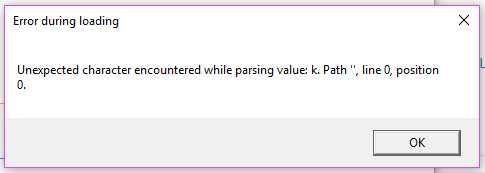
2)Окно после нажатия кнопки start



3) 

4) График после загрузки из файл присоединяется к текущему слева

5)Возможные обрабатываемые исключения:  




1. Здесь и далее под «покупкой» подразумевается обмен первой валюты в паре на вторую, а под «продажей» соответственно второй. Т.е., например, если указывается цена покупки Биткоин-Рубль, то это та сумма в рублях, за которую можно купить один биткоин. [↑](#footnote-ref-1)
2. Под кодом пары понимается сокращенное название пары, например btc\_rur [↑](#footnote-ref-2)