**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

|  |  |
| --- | --- |
| СОГЛАСОВАНО  Доцент департамента  программной инженерии  факультета компьютерных наук  Изображение выглядит как текст  Автоматически созданное описаниеканд. техн. наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н. С. Белова 12 мая 2022г. | УТВЕРЖДАЮ  Академический руководитель  образовательной программы  «Программная инженерия»  Профессор департамента программной инженерии, канд.техн.наук \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В.Шилов  12 мая 2022 г. |

Факультет компьютерных наук  
Образовательная программа «Программная инженерия»

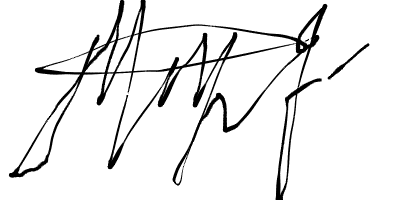
|  |  |
| --- | --- |
| *Подп. и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подп. и дата* |  |
| *Инв. № подл* |  |

**API-МОДУЛЬ ПРОГРАММЫ “БИРЖЕВОЙ БОТ”**

**Техническое задание**

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.****05.04-01 ТЗ 01-1-ЛУ**

**Исполнитель:**  
Студент группы БПИ208  
\_\_\_\_\_\_\_\_\_\_ /Армаш В.В. /  
12 мая 2022 г.

**Москва 2022**

УТВЕРЖДЕН

RU.17701729.05.04 -01 ТЗ 01-1-ЛУ

|  |  |
| --- | --- |
| *Подп. и дата* |  |
| *Инв. № дубл.* |  |
| *Взам. инв. №* |  |
| *Подп. и дата* |  |
| *Инв. № подл* |  |

**API-МОДУЛЬ ПРОГРАММЫ “БИРЖЕВОЙ БОТ”**

**Техническое задание**

RU.17701729.05.04 -01 ТЗ 01-1

**Листов 28**

**Москва 2022**

## АННОТАЦИЯ

Техническое задание – это основной документ, оговаривающий набор требований и порядок создания программного продукта, в соответствии с которым производится разработка программы, ее тестирование и приемка.

Настоящее техническое задание на разработку программы «API-модуль программы “Биржевой бот”», содержит следующие разделы: «Введение», «Основание для разработки», «Назначение разработки», «Требования к программе», «Требования к программным документам», «Технико-экономические показатели», «Стадии и этапы разработки», «Порядок контроля и приемки».

В разделе «Введение» указано наименование и краткая характеристика области применения программы.

В разделе «Основания для разработки» указан документ, на основании которого ведется разработка и наименование темы разработки.

В разделе «Назначение разработки» указано функциональное и эксплуатационное назначение программного продукта.

Раздел «Требования к программе» содержит основные требования к функциональным характеристикам, к надежности, к условиям эксплуатации, к составу и параметрам технических средств, к информационной и программной совместимости, к маркировке и упаковке, к транспортировке и хранению, а также специальные требования.

Раздел «Требования к программным документам» содержит предварительный состав программной документации и специальные требования к ней.

Раздел «Технико-экономические показатели» содержит ориентировочную экономическую эффективность, предполагаемую годовую потребность, экономические преимущества разработки программы.

Раздел «Стадии и этапы разработки» содержит стадии разработки, этапы и содержание работ.

В разделе «Порядок контроля и приемки» указаны общие требования к приемке работы.

Настоящий документ разработан в соответствии с требованиями:

1. ГОСТ 19.101-77 Виды программ и программных документов [1];
2. ГОСТ 19.102-77 Стадии разработки [2];
3. ГОСТ 19.103-77 Обозначения программ и программных документов [3];
4. ГОСТ 19.104-78 Основные надписи [4];
5. ГОСТ 19.105-78 Общие требования к программным документам [5];
6. ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом [6];
7. ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению [7].

Изменения к данному Техническому заданию оформляются согласно   
ГОСТ 19.603-78 [8], ГОСТ 19.604-78 [9].  
Перед прочтением данного документа рекомендуется ознакомиться с терминологией, приведенной в соответствующем разделе настоящего технического задания.

СОДЕРЖАНИЕ

[АННОТАЦИЯ 2](#_Toc103259270)

[ТЕРМИНОЛОГИЯ 6](#_Toc103259271)

[1. ВВЕДЕНИЕ 9](#_Toc103259272)

[1.1. Наименование программы 9](#_Toc103259273)

[1.2. Краткая характеристика и область назначения 9](#_Toc103259274)

[2. ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ 10](#_Toc103259275)

[2.1. Документы, на основании которых ведется разработка 10](#_Toc103259276)

[2.2. Наименование темы разработки 10](#_Toc103259277)

[2.3. Конечное основание для разработки 10](#_Toc103259278)

[3. НАЗНАЧЕНИЕ РАЗРАБОТКИ 11](#_Toc103259279)

[3.1. Функциональное назначение 11](#_Toc103259280)

[3.2. Эксплуатационное назначение 11](#_Toc103259281)

[4. ТРЕБОВАНИЯ К ПРОГРАММЕ 12](#_Toc103259282)

[4.1. Требования к функциональным характеристикам 12](#_Toc103259283)

[4.1.1. Требования к составу выполняемых функций программы 12](#_Toc103259284)

[4.1.2. Организация хранения данных 14](#_Toc103259285)

[4.1.3. Организация входных данных API- модуля для получения информации о ценах акций 14](#_Toc103259286)

[4.1.4. Организация входных данных модуля, эмулирующего работу торгового API 14](#_Toc103259287)

[4.1.5. Организация выходных данных API- модуля для получения информации о ценах акций 14](#_Toc103259288)

[4.1.6. Организация выходных данных модуля, эмулирующего работу торгового API 15](#_Toc103259289)

[4.2. Требования к временным характеристикам 15](#_Toc103259290)

[4.3. Требования к интерфейсу 16](#_Toc103259291)

[4.4. Требования к надежности 16](#_Toc103259292)

[4.4.1. Требования к обеспечению надежного (устойчивого) функционирования программы 16](#_Toc103259293)

[4.4.2. Требования к корректности данных 16](#_Toc103259294)

[4.4.3. Отказы из-за некорректных действий оператора 16](#_Toc103259295)

[4.5. Условия эксплуатации 16](#_Toc103259296)

[4.5.1. Климатические условия эксплуатации 16](#_Toc103259297)

[4.6. Требования к составу и параметрам технических средств 16](#_Toc103259298)

[4.7. Требования к информационной и программной совместимости 16](#_Toc103259299)

[4.8. Требования к транспортировке и хранению 17](#_Toc103259300)

[4.8.1. Требования к транспортировке и хранению программных документов, предоставленных в электронном виде 17](#_Toc103259301)

[4.8.2. Требования к хранению и транспортировке программных документов, предоставляемых в печатном виде 17](#_Toc103259302)

[4.8.3. Требования к транспортировке и хранению программного продукта 17](#_Toc103259303)

[5. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ 18](#_Toc103259304)

[5.1. Предварительный состав программной документации 18](#_Toc103259305)

[5.2. Специальные требования к программной документации 18](#_Toc103259306)

[6. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ 19](#_Toc103259307)

[6.1. Ориентировочная экономическая эффективность 19](#_Toc103259308)

[6.2. Предполагаемая потребность 19](#_Toc103259309)

[7. СТАДИИ И ЭТАПЫ РАЗРАБОТКИ 20](#_Toc103259310)

[8. ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ 22](#_Toc103259311)

[8.2. Общие требования к приёмке работы 22](#_Toc103259312)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 23](#_Toc103259313)

ПРИЛОЖЕНИЕ 1.[СРАВНИТЕЛЬНАЯ ТАБЛИЦА АНАЛОГОВ ПРОЕКТА «БИРЖЕВОЙ БОТ» 25](#_Toc103259314)

ПРИЛОЖЕНИЕ 2.[СРАВНЕНИЕ СУЩЕСТВУЮЩИХ API И ОБОСНОВАНИЕ ВЫБОРА 26](#_Toc103259315)

[ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ 28](#_Toc103259316)

## ТЕРМИНОЛОГИЯ

* 1. API-модуль - программный интерфейс, посредством которого можно обратиться к данным. В данном случае— модуль, отвечающий за взаимодействие с API торговой площадки (биржи), осуществляющий периодические запросы к оному с целью получения актуальной информации о состоянии интересующих позиций.
  2. FIGI, figi, фиджи (англ.: Financial Instrument Global Identifier) — глобальный идентификатор финансового инструмента.
  3. Open-source – проекты с открытым исходным кодом. Разрабатываются преимущественно на некоммерческих началах. Часто проекты публикуются именно так, чтобы появилась возможность улучшить качество продукта посредством его доработки и изучения совместными усилиями.
  4. Агрегация запроса, предварительная агрегация данных — процесс осуществления действий с данными, приводящий их к состоянию, необходимому для работы программы. Пример: перевод времени из человеко-читаемого формата в миллисекунды с момента EPOCH для корректности сравнения с другими временами и датами внутри программы.
  5. Актор (здесь и далее в рамках проекта) - автономный асинхронный модуль, способный принимать сообщения и отвечать на них. Обладает базовым перечнем доступов и функций. Подробнее про работу с акторами можно прочитать в документации Thespian [13] [14].
  6. Акция (share, stock) - вид ценных бумаг.
  7. Виртуальная транзакция — транзакция, не производящая действительных действий на бирже, но меняющая состояние виртуального счёта и баланса акций системы, и т.д.
  8. Виртуальный (здесь и далее в рамках проекта) — имитирующий работу с подлинными сущностями, действующий в рамках тех же правил и ограничений, имеющихся с настоящими объектами. То есть, виртуальный счёт — счёт в широком понимании этого слова, но никак не привязанный к живым деньгам пользователя и имитирующий работу оного.
  9. Восстановление после сбоя — автоматическое возобновление работы программы в случае критической ошибки, предполагает полное соответствие состояния работы программы и значений её данных оным до сбоя, т.е. отсутствие потери данных и т.п.
  10. Измерение эффективности модуля - среднее время обработки одной акции, время от получения запроса до выдачи результата.
  11. Лог — краткий отчёт унифицированного вида о действиях и состоянии системы в данный момент времени.
  12. Логгер — система непрерывного автоматического создания логов.
  13. Лот — единица купли-продажи во время торгов на биржах, аукционах.
  14. Модуль — независимая единица исполнения внутри программы (отдельный поток/процесс), действующий в соответствии с определённым стандартом входных и выходных данных, с возможностью полноценного, в том числе нагрузочного тестирования в отрыве от остальной программы, взаимозаменяемый с любыми другими модулями, действующими в рамках аналогичных стандартов ввода/вывода.
  15. Наблюдаемые позиции (акции) - перечень позиций (акций), добавленных пользователем для наблюдения/отслеживания и совершения программой операций купли/продажи.
  16. Позиция (также встречается под обозначениями „акция”, „метрика”) — интересующая нас единица торговли, в отношении которой проводится работа системы в том или ином виде. В зависимости от контекста может подразумевать как наименование позиции, так и конкретную единицу с таким названием (одну акцию, например), а также временной ряд с информацией о динамике изменения параметров данной позиции (количество лотов, текущая цена, и т.п.)
  17. Полный перечень акций – перечень акций, получаемый с помощью выбранных API.
  18. Свободные денежные средства - те, которые остаются у владельца после уплаты всех текущих обязательств.
  19. Токен (в контексте данного проекта) – набор символов, в котором зашифрована информация о владельце, правах доступ и прочей необходимой для авторизации в TINKOFF INVEST API [12] информации.
  20. Торговый API — предоставляемый биржей API для осуществления операций торговли на ней, а также получения актуальной информации о статусе имеющихся позиций на ней.
  21. Транзакция —процедура разового обмена информацией/деньгами с обратной связью, подтверждающей успешное её проведение. В рамках проекта мы рассматриваем транзакции, удовлетворяющие требованиям ACID.
  22. Трейдинг - совершение торговых операций на биржевом рынке с целью получения прибыли.
  23. Фондовая биржа - площадка для торговли ценными бумагами.

# ВВЕДЕНИЕ

# Наименование программы

Наименование программы – «API-модуль программы “Биржевой бот”».

Краткое наименование программы – «API-модуль».

# Краткая характеристика и область назначения

«Биржевой бот» - автоматизированная распределённая (модульная, многопоточная) система на базе машинного обучения для (виртуальной) торговли на бирже и составления кратковременных прогнозов поведения позиций.

«Биржевой бот» ориентирован на рынок ценных бумаг. Проект потенциально может развиться в «боевого» бота, позволяющего людям автоматизировать торговлю на рынках ценных бумаг, криптовалют и т. п. логируемых позиций.

«API-модуль программы “Биржевой бот”» будет выполнять 2 основные задачи:

1. Эмуляция работы торгового API
2. Использование стороннего APIдля непосредственно получения, агрегации и передачи данных, связанных с акциями.  
   Предметная область данной курсовой работы совпадает с областью всего проекта.. Целевая аудитория - люди, напрямую связанные с инвестициями. Необходимость разработки связана с проблемой получения и обработки актуальных данных, обеспечивающих работу остальных модулей.

# ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ

# Документы, на основании которых ведется разработка

Учебный план подготовки бакалавров по направлению 09.03.04 «Программная инженерия»НИУ ВШЭ и утвержденная академическим руководителем тема выбранного курсового проекта.

# Наименование темы разработки

Наименование программы – «API-модуль программы “Биржевой бот”».

Условное обозначение темы разработки – «API-модуль».

Программа выполнена в рамках темы курсовой работы «API-модуль программы “Биржевой бот”» в соответствии с учебным планом подготовки бакалавров (НИУ ВШЭ, факультет компьютерных наук) по направлению «Программная инженерия».

# 2.3. Конечное основание для разработки

Программа выполняется в рамках темы курсовой работы в соответствии с учебным планом подготовки бакалавров по направлению 09.03.04. «Программная инженерия» Национального исследовательского университета «Высшая школа экономики», факультет компьютерных наук, департамент программной инженерии.

# НАЗНАЧЕНИЕ РАЗРАБОТКИ

# Функциональное назначение

«Биржевой бот» представляет собой программу (бота), который в реальном времени совершает виртуальные операции торговли на бирже и логирует происходящее. Конечный продукт состоит из 2 частей (6 модулей).

Бот реализует набор функций, связанных с виртуальной торговлей: операции с денежными средствами на виртуальном счету, операции с акциями, предоставление информации о текущем размере средств.  
Разработка «API-модуль программы “Биржевой бот”» направлена на получение, обработку и логирование данных по специфическому запросу – т.е., функциональное назначение программы коррелирует с ее модулями:  
API-модуль для получения информации об акциях существует для непосредственно получения, агрегации и передачи данных, связанных с акциями, а основной задачей API-модуля, эмулирующего работу торгового API, является логирование происходящего процесса и сохранение логов в соответствующие файлы.

# Эксплуатационное назначение

«API-модуль программы “Биржевой бот”» предназначен в первую очередь для поддержания корректной работы программы «Биржевой бот».  
  
В рамках данного проекта программа будет предоставлять необходимую для работы остальных модулей информацию и эмулировать работу торгового API.  
Вполне допускается возможность последующей реализации программы в open-source источниках для использования в схожих проектах, однако под конечными пользователями данной программы на текущем этапе подразумеваются разработчики остальных частей проекта, упомянутого выше.

# ТРЕБОВАНИЯ К ПРОГРАММЕ

# Требования к функциональным характеристикам

# Требования к составу выполняемых функций программы

**а) API- модуль для получения информации о ценах акций**В первую очередь необходимо наладить взаимодействие модуля с выбранным API.  
Выбор пал на официальный клиент Tinkoff Invest [12]. Мотивация выбора описана в приложении 2.  
  
Ключевые требования к модулю (влияют на работу других модулей): **1. Обработка и предоставление полученной информации** в установленных форматах (подробнее о форматах в развернутом списке требований, предоставленном ниже и в п. «Организация входных и выходных данных»)  
**2. Эффективность при работе** с большим количеством наблюдаемых акций (агрегация ответа сервера не более секунды)  
**3. Передача информации о наличии неполадок** на стороне сервиса, предоставляющего информацию о ценах (передача сообщений основному модулю о наличии неполадок)  
**4. Проверка входных данных об акциях**, поступающих от основного модуля, на корректность и существование ( т.е. решение ситуаций, когда поступает запрос на несуществующие позиции\ запрос на продажу\покупку невозможного количества акций - считаем невозможными те данные, которые не кратны лоту или <0).

**5. Восстановление после сбоя** (см.п.4.4. «Требования к надежности»)

**Развернутый список требований:**

1. Использование токена для работы с Tinkoff API [12]. Токен генерируется в личном кабинете сайта Тинькофф Инвестиции [11], срок жизни токена – 3 месяца с даты последнего использования. В проекте используется readonly token[10] — он предназначен только для получения различной информации, такой как состояние портфеля, расписание торгов различных торговых площадок, текущие котировки, исторические данные и т.п., а выставление торговых поручений невозможно.
2. Предоставление модулям программы полного перечня акций.  
   Предоставляется: время запроса – общее для всех акций, figi\имя\цена\валюта\статус покупки\статус продажи\лот – для каждой акции.
3. Предоставление перечня акций, доступных к покупке.  
   Предоставляется: время запроса – общее для всех акций, figi\имя\цена\валюта\статус покупки\статус продажи\лот – для каждой акции. Отличие от предыдущего пункта - статус покупки у акций в предоставляемом перечне всегда true.
4. Предоставление перечня акций, доступных к продаже.  
   Предоставляется: время запроса – общее для всех акций, figi\имя\цена\валюта\статус покупки\статус продажи\лот – для каждой акции. Отличие от предыдущего пункта - статус продажи у акций в предоставляемом перечне всегда true.
5. Предоставление цены для конкретной акции за произвольный период в прошлом.  
   Произвольный период и шаг подаются как входные данные, шаг – минута\5минут\15 минут\час\день.  
   Предоставляется список время-цена за запрашиваемый период.
6. Получение информации об акции по ее названию.  
   Требование, существующее для выполнения запросов информации по конкретной акции – либо получение информации (цена\валюта\статус покупки\продажи\лот) на момент запроса, т. е. актуальной, либо получение информации по конкретной акции за произвольный период в прошлом.  
     
   **б) API-модуль, эмулирующий работу торгового API**

В этом модуле логируется запись транзакций.

1. Совершение виртуальных операций купли/продажи на бирже в соответствии с принятым основным модулем решением (в рамках проекта совершение виртуальной операции мы осуществляем через логирование транзакций)
2. Предоставление информации о совершенных операциях в формате [время- позиция - количество единиц - тип операции(купля/продажа) - цена за единицу]
3. Запись истории транзакций в файл [статус транзакции: success\failure, информация об операции – см.п.2 текущего списка].
4. Обработка некорректных\неактуальных данных – передача оповещения основному модулю, продолжение работы программы (за счет игнорирования данных, признанных некорректными).  
   Статус транзакции логируется как failure.

# Организация хранения данных

Модуль-эмулятор, описанный в п. 4.1.1(б), сохраняет логи записей транзакций в отдельный json-файл. Файл именуется датой, когда проводились транзакции. Новый день – новый файл.

# Организация входных данных API- модуля для получения информации о ценах акций

Запросами можно считать весь развернутый список требований к модулю, описанный в п.4.1.1(а).  
Для их выполнения необходимо передавать токен Tinkoff API [10], который уже указан в программе.  
Кроме того, в списке присутствует два типа запросов, для которых формат запроса необходимо уточнить:

1. Запрос текущей информации для наблюдаемых акций в реальном времени, формат запроса [название акции]
2. Запрос цен для конкретной акции за произвольный период в прошлом, формат запроса [название акции - начало периода – конец периода -шаг: шаг устанавливается в рамках дозволенного tinkoff.invest.CandleInterval – не указан\минута\5 минут\15 минут\час\день]

# Организация входных данных модуля, эмулирующего работу торгового API

Программа принимает на вход следующий запрос:  
Решение, касающееся совершения виртуальной операции купли\продажи - сообщение, генерируемое модулем принятия решений в формате [акцию N купить/продать в такое-то время T].

# Организация выходных данных API- модуля для получения информации о ценах акций

Программа предоставляет модулям следующие данные:

1. Передача полного перечня акций (получаемого с помощью API) в формате [время запроса, (figi, имя, цена, валюта, статус покупки, статус продажи, лот)] - словарь.
2. Передача перечня акций, доступных к покупке, в формате  
   [время запроса, (figi, имя, цена, валюта, статус покупки, статус продажи, лот)] – словарь.
3. Передача перечня акций, доступных к продаже, в формате  
   [время запроса, (figi, имя, цена, валюта, статус покупки, статус продажи, лот)] – словарь.
4. Передача информации о конкретной акции за запрошенный период в формате

[время-цена] – лист, размеры определяются запрашиваемым периодом.

1. Передача сообщения о наличии неполадок на стороне сервиса, предоставляющего информацию об акциях. Если возникает одна из установленных ошибок, возвращается сообщение 'Internal error'.
2. Передача сообщения о невозможности обработать запрос (в связи с его отсутствием запрашиваемых позиций\некорректным форматом запроса) - передается сообщение 'No such share name'.

# Организация выходных данных модуля, эмулирующего работу торгового API

Выходными данными модуля считаются json-файлы, содержащие логи транзакций.  
Логируются как успешные, так и неуспешные транзакции. При неуспешных транзакциях пишется причина неудачи.  
*Пример:  
Обработка случая с запросом на покупку\продажу невозможного\* количества позиций.  
Статус транзакции логируется как failure, выводится соответствующее сообщение:*

*"status": "failure",*

*"share": operation['share'],*

*"operation": operation['type'],*

*"message": incorrect quantity to {operation['type']}, lot is {share\_info['lot']}*

*\* невозможным количеством считаем число, которое меньше нуля\не кратно лоту.*

# Требования к временным характеристикам

Скорость работы системы зависит от скорости интернет-подключения и от объема поступающих данных. Эффективной работа системы будет считаться, если агрегация ответа сервера длится не более секунды.

# Требования к интерфейсу

Требования к интерфейсу не предъявляются.

# Требования к надежности

### **4.4.1. Требования к обеспечению надежного (устойчивого) функционирования программы**

При любых действиях пользователя и при любых входных данных, система не должна завершаться из-за ошибки.

### **4.4.2. Требования к корректности данных**

Программа должна обеспечивать проверку корректности входных данных.

Некорректными входными данными считаются те данные, которые не соответствуют описанному формату запроса. Форматы запроса описан в п.4.1.3 и п.4.1.4 «Организация входных данных».

### **4.4.3. Отказы из-за некорректных действий оператора**

Отказ может возникнуть только при условии повреждения исходного кода.

# Условия эксплуатации

## 4.5.1. Климатические условия эксплуатации

Климатические условия эксплуатации должны удовлетворять требованиям, предъявляемым к техническому устройству, используемому пользователем.

Требования к эксплуатации технического устройства предъявляются производителем технического устройства.

# Требования к составу и параметрам технических средств

Для функционирования системы требуются следующие программные совместимости:

1. Предустановленный Python версии 3.9 или выше
2. Доступ к жесткому диску
3. Доступ к интернету
4. Unix-система.

# Требования к информационной и программной совместимости

Для работы «API-модуль программы “Биржевой бот”» разработчику необходимы:

* 1. Предустановленный Python версии 3.9 или выше (основной язык разработки)
  2. Клиент для работы с Tinkoff API - tinkoff.invest[12]
  3. Библиотека для работы с акторами - thespian.actors[14]
  4. Доступ к интернету.

# Требования к транспортировке и хранению

# Требования к транспортировке и хранению программных документов, предоставленных в электронном виде

Программные документы в электронном виде загружаются в систему Smart LMS (Smart Learning Management System). Требования к хранению и транспортировке не предъявляются.

Программа распространяется в виде электронного пакета, содержащего программную документацию, приложение (исполняемые файлы и прочие необходимые для работы файлы).

# Требования к хранению и транспортировке программных документов, предоставляемых в печатном виде

Программные документы, предоставляемые, в печатном виде, должны соответствовать общим правилам учета и хранения программных документов, предусмотренных стандартами ЕСПД (Единой системы программной документации) и соответствовать требованиям ГОСТ 19.602–78[9].

# Требования к транспортировке и хранению программного продукта

Программное изделие может храниться и транспортироваться на USB-носителе или в облачном хранилище.

# ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

# Состав программной документации

1. «API-модуль программы “Биржевой бот”». Техническое задание (ГОСТ 19.201-78[7]);
2. «API-модуль программы “Биржевой бот”». Программа и методика испытаний (ГОСТ 19.301-79);
3. «API-модуль программы “Биржевой бот”». Текст программы (ГОСТ 19.401-78);
4. «API-модуль программы “Биржевой бот”». Пояснительная записка (ГОСТ 19.404-79);
5. «API-модуль программы “Биржевой бот”». Руководство программиста (ГОСТ 19.504-79);

# Специальные требования к программной документации

1. Документы к программе должны быть выполнены в соответствии с ГОСТ 19.106-78 и ГОСТами к каждому виду документа (см. п. 5.1.);
2. Пояснительная записка должна быть загружена в систему Антиплагиат через LMS «НИУ ВШЭ». Лист, подтверждающий загрузку пояснительной записки, сдается в учебный офис вместе со всеми материалами не позже, чем за день до защиты курсовой работы;
3. Вся документация также воспроизводится в печатном виде, она должна быть подписана академическим руководителем образовательной программы 09.03.04 «Программная инженерия», руководителем разработки и исполнителем перед сдачей курсовой работы в учебный офис не позже одного дня до защиты;
4. Документация также сдается в электронном виде в формате .pdf или .docx. проект программы в архиве формата .zip или .rar;
5. Все документы перед защитой курсовой работы должны быть загружены в информационно-образовательную среду НИУ ВШЭ систему Smart LMS (Smart Learning Management System), в личном кабинете дисциплина - «Курсовой проект, 2 курс ПИ», одним архивом.

# ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

# Ориентировочная экономическая эффективность

В рамках данной работы оценка экономической эффективности не предусмотрена.

Экономическая эффективность всего проекта (частью которого является данная работа) описана в общем техническом задании «Биржевой бот».  
Сравнительная таблица аналогов всего проекта предоставлена в приложении 1.

# Предполагаемая потребность

Данная программа будет использована для обеспечения корректной работы проекта «Биржевой бот». Вполне допускается возможность последующей реализации программы в open-source источниках для использования в схожих проектах.

# СТАДИИ И ЭТАПЫ РАЗРАБОТКИ

Стадии и этапы разработки были выявлены с учетом ГОСТ 19.102-77 [2]:

|  |  |  |  |
| --- | --- | --- | --- |
| **Стадии разработки** | **Этапы работ** | **Содержание работ** | **Даты исполнения** |
| 1. Техническое задание | Подготовительные работы | * Постановка задачи. * Сбор исходных теоретических материалов. * Обоснование возможности решения поставленной задачи. * Определение структуры входных и выходных данных. * Предварительный выбор методов решения задач. | 01.12.2021-31.12.2021 |
| Разработка и утверждение технического задания | * Определение требований к программе. * Определение требований к техническим средствам. * Определение стадий. Этапов и сроков разработки программы и документации на неё. * Согласование и утверждение технического задания. | 01.01.2022 - 28.02.2022 |
| 2. Рабочий проект | Разработка программы | Программирование и отладка API модулей, совмещение и настройка с остальными модулями программы | 28.02.2022 - 30.04.2022 |
| Разработка программной документации | Разработка программных документов в соответствии с требованиями ГОСТ 19.101-77 | 25.03.2022 - 30.04.2022 |
| Испытания программы | Разработка, согласование и утверждение порядка и методики испытаний | 30.04.2022 - 08.05.2022 |
| 3. Внедрение | Подготовка и передача программы | Проведение предварительных испытаний | с 9.05.2022 |

Программа и документация к ней разрабатываются к утвержденным срокам сдачи курсового проекта (13.05.2022).

# ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ

* 1. **Виды испытаний**

Производится проверка корректного выполнения программой заложенных в нее функций, то есть осуществляется функциональное тестирование программы. Также осуществляется визуальная проверка интерфейса программы на соответствие с характеристиками интерфейсного модуля и приложением в содержании технического задания. Функциональное тестирование осуществляется в соответствии с документом «API-модуль программы «Биржевой бот». Программа и методика испытаний (ГОСТ 19.301–79), в котором указывают:

1. Перечень функций программы, выделенных в программе для испытаний, и перечень требований, которым должны соответствовать эти функции (со ссылкой на пункт 4.1.1. настоящего технического задания);
2. Перечень необходимой документации и требования к ней (со ссылкой на пункт 5 настоящего технического задания);
3. Методы испытаний и обработки информации;
4. Технические средства и порядок проведения испытаний;

Защита выполненного проекта осуществляется комиссии, состоящей из преподавателей департамента программной инженерии, в утверждённые приказом декана факультета компьютерных наук сроки.

# 8.2. Общие требования к приёмке работы

Прием программного продукта происходит при полной работоспособности программы при различных входных данных, при выполнении указанных в пункте 4.1.1. настоящего документа функций, при выполнении требований указанных в пункте 4.2. настоящего документа и при наличии полной документации к программе, указанной в пункте 5.1., выполненной в соответствии со специальными требования указанными в пункте 5.2. настоящего технического задания.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ 19.101-77 Виды программ и программных документов. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
2. ГОСТ 19.102-77 Стадии разработки. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
3. ГОСТ 19.103-77 Обозначения программ и программных документов. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
4. ГОСТ 19.104-78 Основные надписи. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
5. ГОСТ 19.105-78 Общие требования к программным документам. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
6. ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
7. ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
8. ГОСТ 19.603-78 Общие правила внесения изменений. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
9. ГОСТ 19.604-78 Правила внесения изменений в программные документы, выполненные печатным способом. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
10. Tinkoff Invest Api Token// [Электронный ресурс] - Режим доступа: свободный, URL: <https://tinkoff.github.io/investAPI/token/> (5 мая 2022)
11. Тинькофф инвестиции // [Электронный ресурс] - Режим доступа: свободный, URL: <https://www.tinkoff.ru/invest/> (5 мая 2022)
12. Тинькофф клиент API // [Электронный ресурс] - Режим доступа: свободный, URL: <https://tinkoff.github.io/invest-python/> (5 мая 2022)
13. Thespian Actors User`s Guide// [Электронный ресурс] - Режим доступа: свободный, URL: <https://thespianpy.com/doc/using.html> (5 мая 2022)
14. Thespian Actors Library// [Электронный ресурс] - Режим доступа: свободный, URL: <https://github.com/kquick/Thespian>(5 мая 2022)
15. Finage.co.uk - предоставляет информацию в реальном времени для всех бирж в Соединенных Штатах и для внебиржевых рынков //[Электронный ресурс] - Режим доступа: свободный, URL: <http://finage.co.uk/> (5 мая 2022)
16. **Yahoo Finance API** // [Электронный ресурс] - Режим доступа: свободный, URL: [https://www.yahoofinanceapi.com](https://www.yahoofinanceapi.com/) (5 мая 2022)
17. Alpha Vantage API // [Электронный ресурс] - Режим доступа: свободный, URL:  
    <https://www.alphavantage.co/documentation/> (5 мая 2022)
18. EXANTE API // [Электронный ресурс] - Режим доступа: свободный, URL:  
    <https://exante.eu/ru/technology/> (5 мая 2022)

**ПРИЛОЖЕНИЕ 1**

# СРАВНИТЕЛЬНАЯ ТАБЛИЦА АНАЛОГОВ ПРОЕКТА «БИРЖЕВОЙ БОТ»

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Tickeron | Trade Ideas | Stoic AI | Streak | **Биржевой бот** |
| Наличие открытого исходного кода | **-** | **-** | **-** | **-** | **+** |
| Модульность программы | **-** | **-** | **-** | **-** | **+** |
| Заявленное использование актор-системы как модели параллельных вычислений | **-** | **-** | **-** | **-** | **+** |
| Кроссплатформенность | **+** | **+** | **+** | **+** | **-\*\*** |
| Бесплатная подписка | **-** | **-** | **+** | **-** | **+** |
| Бесплатный пробный период | **-** | **+** | **+** | **+** | **+** |
| Операции с криптовалютой(криптотрейдинг) | **+** | **-** | **+** | **+** | **-\*** |
| Отображение (в виде временного ряла) детального прогноза real-time | **-** | **-** | **-** | **-** | **+** |
| Возможность виртуальной торговли | **-** | **-** | **-** | **+** | **+** |

\*Благодаря модульности программы может быть добавлен соответствующий модуль  
\*\*Может быть собрано пользователем на любой системе на свой страх и риск

**ПРИЛОЖЕНИЕ 2**

# СРАВНЕНИЕ СУЩЕСТВУЮЩИХ API И ОБОСНОВАНИЕ ВЫБОРА

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | [Finage.co.uk](http://finage.co.uk/)[15] | [Alpha Vantage API](https://www.alphavantage.co/documentation/) [17] | [EXANTE](https://exante.eu/ru/technology/)[18] | [Yahoo Finance API](https://www.yahoofinanceapi.com)[16] | Tinkoff API[12] |
| Бесплатная подписка | + | + | + | + | + |
| Верификация данных пользователя (удостоверения личности) для использования сервиса | - | - | + | - | + |
| Лимиты на запрос (обращения к API) | + | \_ | \_ | + | - |
| Подробная документация и примеры использования | + | + | \_ | \_ | + |
| Возможность получения текущих цен акций | + | \_ | + | + | + |
| Возможность получения данных за прошедший период | + | + | + | \_ | + |
| Возможность использования для реальной торговли на бирже (личное пользование) | - | - | + | - | + |

Цветом в табличке обозначена польза для проекта. Зеленый – подходит, красный – не подходит. Насыщенный красный – категорически не подходит. По цветовой схеме видно, что под все предъявляемые требования идеально подошел Tinkoff API.   
  
Следует отметить, что этот выбор не лишен недостатков: для его использования необходим открытый в банке инвестиционный счет, а публичный сервис на основе TINKOFF INVEST API сделать невозможно, т. к. это противоречит установленным условиям использования (TINKOFF INVEST API предоставляется только клиентам Тинькофф без права на ретрансляцию данных).

# ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | | |
| Номера листов (страниц) | | | | | Всего листов (страниц в докум.) | № документа | Входящий № сопроводительного докум. и дата | Подп. | Дата |
| Изм. | Измененных | Замененных | Новых | Аннулированных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |