

**Московский Авиационный институт  
(Национальный исследовательский университет)**



**Институт №8  
«Компьютерные науки и прикладная математика»**

**Кафедра 813**

**«Компьютерная математика»**

**Курсовой проект по дисциплине  
«Интеллектуальные системы»  
по теме: «Использование экспертных систем для составления  
оптимальных портфелей ценных бумаг»**

Студенты: Вартумян В.Д. Таскина А.П.

Группа: М80-310Б-19

Преподаватель: Чернова Т.А.

Оценка:

Дата:

Москва – 2022

## Оглавление

Введение	4
Актуальность	4
Цель работы	5
Задачи	5
Теоретическая часть	6
Экспертные системы	6
Предметная область	8
Основные термины	8
Портфельная теория Марковица	9
Основные положения современной портфельной теории	10
Математическая формулировка	11
Использование портфельной теории в экспертной системе	13
Практическая часть	15
Выбор стека технологий	16
Язык программирования	16
Выбор платформы и фреймворков	17
Выбор системы управления базами данных	18
Выбор API для сбора данных с рынка	19
Алгоритмизация и разработка программного обеспечения	20
Блок-схема алгоритма	21

Архитектура программного обеспечения	22
Тестирование ПО	23
Результат работы программы	24
Общие выводы	27
Список литературы	28
Приложения	29

## **Введение**

### **Актуальность**

Для большинства российских предприятий маркетинговое управление становится одним из условий выживания и успешного функционирования. При этом обеспечение эффективности такого управления требует умения предвидеть вероятное будущее состояние предприятия и среды, в которой оно существует, вовремя предупредить возможные сбои и срывы в работе. Это достигается с помощью прогнозирования как плановой, так и практической работы предприятия по всем направлениям его деятельности, и в частности в области прогнозирования сбыта продукции (товаров, услуг).

Многообразие проблем, возникающих при обеспечении жизнедеятельности предприятия и являющихся предметом прогнозирования, приводит к появлению большого количества разнообразных прогнозов, разрабатываемых на основе определенных методов прогнозирования. Поскольку современная экономическая наука располагает большим количеством разнообразных методов прогнозирования, каждый менеджер и специалист по планированию должен овладеть навыками прикладного прогнозирования, а руководитель, ответственный за принятие стратегических решений, должен к тому же уметь сделать правильный выбор метода прогнозирования.

Анализ методов прогнозирования, изучение этих методов, использование их в разных сферах деятельности является мероприятием рационализаторского характера. Степень достоверности прогнозов можно затем сравнить с действительно реальными показателями, и, сделав выводы, приступить к следующему прогнозу уже с существующими данными, т.е. имеющейся тенденцией. Опираясь на полученные данные, можно во временном аспекте переходить на более высокую ступень.

## Цель работы

Составление экспертной системы выбора оптимального портфеля на основе экспертной системы *PMIDSS*.

## Задачи

- Исследовать предметную область, изучить и исследовать основные представления знаний и способы оптимизации портфеля;
- Сформулировать и реализовать базу знаний с модулем приобретения знаний;
- Реализовать кодовую библиотеку для работы с логистическим вводом-выводом;
- Спроектировать и реализовать экспертную систему как серверное приложение на микросервисной архитектуре;
- Спроектировать интерфейс пользовательского приложения для работы экспертов и обычных пользователей;
- Протестировать работоспособность приложения;

## Теоретическая часть

### Экспертные системы

**Экспертные системы** — это сложные программные комплексы, аккумулирующие знания специалистов в конкретных предметных областях и тиражирующие этот эмпирический опыт для консультаций менее квалифицированных пользователей.

Области применения систем, основанных на знаниях, весьма разнообразны: бизнес, производство, военные приложения, медицина, социология, геология, космос, сельское хозяйство, управление, юриспруденция и др.

Системы, основанные на знаниях (СОЗ) — это системы программного обеспечения, основными структурными элементами которых являются *база знаний* и *механизм логических выводов*. Среди СОЗ можно выделить:

- интеллектуальные информационно-поисковые системы;
- экспертные системы (ЭС).

Интеллектуальные информационно-поисковые системы отличаются от предыдущего поколения информационно-поисковых систем не только гораздо более обширным справочно-информационным фондом, но и важнейшей способностью формировать адекватные ответы на запросы пользователя даже тогда, когда запросы не носят прямого характера.

Наиболее известным практическим примером СОЗ могут служить экспертные системы, способные диагностировать заболевания, оценивать потенциальные месторождения полезных ископаемых, осуществлять обработку естественного языка, распознавание речи и изображений и т.д. Экспертные системы являются первым шагом в практической реализации исследований в области искусственного интеллекта.

Одним из примеров такой экспертной системы является взятая для основы изучения предметной области система ***PMIDSS***.

***PMIDSS*** — система поддержки принятия решений при управлении портфелем инвестиций. В число решаемых системой задач входят: выбор портфеля ценных бумаг; долгосрочное планирование инвестиций. Данная ЭС пользуется

смешанной системой представления знаний и использует разнообразные механизмы ввода.

Структурные элементы, составляющие экспертную систему, выполняют следующие функции.

- *База знаний* реализует функции представления знаний в конкретной предметной области и управление ими.
- *Механизм логических выводов* выполняет логические выводы на основании знаний, имеющихся в базе знаний.
- *Пользовательский интерфейс* необходим для правильной передачи ответов пользователю, иначе пользоваться системой крайне неудобно.
- *Модуль приобретения знаний* необходим для получения знаний от эксперта, поддержки базы знаний и дополнения ее при необходимости.
- *Модуль ответов и объяснений* формирует заключение экспертной системы и представляет различные комментарии, прилагаемые к заключению, а также объясняет мотивы заключения.

Перечисленные структурные элементы являются наиболее характерными, хотя в реальных экспертных системах их функции могут быть соответствующим образом усилены или расширены.

Знания в базе знаний представлены в конкретной форме и организация базы знаний позволяет их легко определять, модифицировать и пополнять. Решение задач с помощью логического вывода на основе знаний хранящихся в базе знаний, реализуется автономным механизмом логического вывода. Хотя оба эти компонента системы с точки зрения ее структуры являются независимыми, они находятся в тесной связи между собой и определение модели представления знаний накладывает ограничения на выбор соответствующего механизма логических выводов.

## Предметная область

### Основные термины

**Брокер** — профессиональный участник рынка ценных бумаг, за деятельностью которого следит государство и Центральный банк. С его помощью вы можете покупать и продавать ценные бумаги, торговать валютой и другими финансовыми инструментами.

**Ценная бумага** — документ, который дает вам право на владение каким-то финансовым инструментом. Самые распространенные ценные бумаги — акции и облигации.

**Акция** — вид долевого ценной бумаги, дающий владельцу право на получение части чистого дохода от деятельности акционерного общества в виде дивидендов, а также на часть имущества компании в случае ее ликвидации.

**Облигация** — это долговая ценная бумага, по которой эмитент - компания или государство - обязуется выплатить инвестору определенную сумму и определенный процент в будущем.

**Дивиденды** — это выплаты инвесторам, которые они получают за владение простыми или привилегированными акциями. На итоговую сумму влияет несколько факторов: вид ценных бумаг; количество купленных акций; итоги работы предприятия в рамках отчетного периода (года, полугодия и т. д.). Решение о возврате инвесторам части прибыли компании в форме дивидендов принимает совет директоров.

**Инвестиции** — это финансовый инструмент, который предполагает вложение средств с целью получения прибыли.

**Инвестор** — тот, кто вкладывает собственные средства с целью получения прибыли.

**Портфель акций** - это совокупность всех акций, находящихся во владении одного лица. В случае, если речь идет об инвестиционном портфеле в более широком смысле, подразумевает, что помимо акций он может включать и другие финансовые инструменты: облигации, валюту и т.д.



## Портфельная теория Марковица

**Портфельная теория** — теория инвестиционного менеджмента, основанная на статистических методах оптимизации формируемого инвестиционного портфеля по избранному критерию соотношения уровня доходности и риска.

Эта теория состоит из следующих основных разделов:

- *Оценка инвестиционных качеств отдельных объектов инвестирования* — описание всех видов активов с точки зрения ожидаемых дохода и риска;
- *Формирование индивидуальных инвестиционных решений* — определение того, каким образом активы должны быть распределены между различными классами инвестиций, такими, как акции или облигации;
- *Оптимизация портфеля* — уравнивание риска и доходности при выборе ценных бумаг, которые будут включены в портфель вложений, например определение того, какой портфель акций предлагает наивысший доход при данном уровне ожидаемого риска;
- *Совокупная оценка инвестиционного портфеля по соотношению уровня доходности и риска* — выделение различных видов результатов, показываемых каждой из акций (риск), и классификация их на относящиеся к рынку (систематические) и относящиеся к данному промышленному сектору или к данному/виду ценных бумаг (остаточные).

Портфельная теория — это комплексный подход к принятию инвестиционных решений, который позволяет инвестору классифицировать, оценивать и контролировать как тип, так и объем ожидаемого риска и дохода; также называется теорией управления портфелем или современной портфельной теорией.

Важными положениями портфельной теории являются численное выражение соотношения риска и дохода и предположение о том, что инвесторы должны получать компенсацию за принимаемый ими риск. Портфельная теория вложений расходится с традиционным анализом ценных бумаг в том, что она смещает основной акцент от анализа характеристик отдельных вложений к определению статистических взаимосвязей конкретных ценных бумаг, составляющих портфель в целом.

*Портфельная теория* — это концепция, разработанная Г. Марковицем в 1952 г., в которой исследуются способы максимизации ожидаемого дохода

на инвестиции, составляющие портфель ценных бумаг, при правильном распределении риска. Марковиц считал, что рациональные инвесторы будут рисковать своими сбережениями только в том случае, если ожидаемый доход будет достаточной компенсацией за риск. Значительная часть инвесторов предпочитает иметь эффективный инвестиционный портфель, т.е. максимальную безопасность вложений относительно ожидаемого дохода или максимально высокий доход относительно данной степени риска. Практические выводы портфельной теории заключаются в том, что инвесторы должны правильно распределять риски, составляя свой портфель из разных инструментов фондового рынка.

Модель Гарри Марковица, также известная как среднedisперсионная модель основана на ожидаемой доходности (средний показатель) и стандартных отклонениях (дисперсии) различных портфелей. Применяя данную модель, можно сделать наиболее эффективный выбор, анализируя различные портфели определенных активов. Метод наглядно указывает инвесторам, как снизить риск, в случае, если они выбрали активы не «двигающиеся» синхронно.

### **Основные положения современной портфельной теории**

1. Современная портфельная теория опирается на следующие ключевые понятия:
2. Для покупки и продажи ценных бумаг нет транзакционных издержек. Нет спада между ценой покупки и продажи. Не уплачивается налог, единственное, что играет роль в определении того, какие ценные бумаги инвестор будет покупать, это риск.
3. Инвестор имеет возможность открыть любую позицию любого размера и для любого актива. Ликвидность рынка бесконечна, и никто не может сдвинуть рынок. Так что ничто не мешает инвестору открыть позицию любого размера для любого актива.
4. При принятии инвестиционных решений, инвестор не принимает во внимание налоги, полученные дивиденды или прирост стоимости капитала.
5. Инвесторы, как правило, рациональны и не склонны рисковать. Они в курсе всех инвестиционных рисков, и занимают позиции, степень риска которых известна и ожидают повышения доходности при повышенной волатильности рынков.
6. Соотношения риска и доходности, рассматриваются в течение одного и того же периода времени. Долгосрочные и краткосрочные спекулянты имеют одинаковые мотивы: ожидаемая прибыль и временной диапазон.

7. Инвесторы имеют одинаковые взгляды на оценку рисков. Всем инвесторам предоставляется информация и их продажа или покупка зависит от идентичной оценки инвестиций, и все имеют одинаковые ожидания от инвестиций. Продавец мотивирован совершить продажу только потому, что другой актив имеет уровень волатильности, которая соответствует его желаемой прибыли. Покупатель совершит покупку, потому что этот актив имеет уровень риска, который соответствует желаемой доходности.
8. Инвесторы стремятся контролировать риск, только путем диверсификации активов.
9. На рынке все активы могут быть куплены и проданы, в том числе человеческий капитал.
10. Политика и психология инвестора не влияют на рынок.
11. Риск портфеля напрямую зависит от нестабильности доходов данного портфеля.
12. Инвестор отдает предпочтение повышению уровня утилизации.
13. Инвестор либо максимизирует свою прибыль с минимальным риском либо максимизирует доходность своего портфеля при заданном уровне риска.
14. Анализ основан на одной периодичной модели инвестирования.

### **Математическая формулировка**

Доходность портфеля измеряется как средневзвешенная сумма доходностей входящих в него бумаг.

$$r_p = \sum_{i=1}^n w_i \cdot r_i ,$$

где  $w_i$  — доля инструмента в портфеле,  $r_i$  — доходность инструмента.

Риск отдельного инструмента оценивается как среднеквадратичное (стандартное) отклонение его доходности. Для расчета общего риска портфеля необходимо отразить совокупное изменение рисков отдельного инструмента и их взаимное влияние (через ковариации и корреляции — меры взаимосвязи).

$$\sigma_p = \sqrt{w_i \cdot w_j \cdot V_{ij}} = \sqrt{\sum_{i=1}^n w_i^2 \cdot \sigma_i^2 + 2 \sum_{i=1}^{n-1} \sum_{j=i+1}^n w_i \cdot w_j \cdot k_{ij} \cdot \sigma_i \cdot \sigma_j}$$

где  $\sigma_i$  — стандартное отклонение доходностей инструмента,  $k_{ij}$  — коэффициент корреляции между  $i, j$ -м инструментом,  $V_{ij}$  — ковариация доходностей  $i$ -го и  $j$ -го финансового инструмента,  $n$  — количество финансовых инструментов в рамках портфеля.

Таким образом, в рамках правильно подобранного портфеля риски снижаются за счет обратной корреляции инструментов. При этом устраняются не только специфические риски инструмента, но и снижается систематический (рыночный) риск.

Для составления портфеля решается оптимизационная задача. При этом в базовом виде использование заемных средств не предполагается, то есть сумма долей активов равняется единице, а доли эти положительны.

*Минимизируем риск при минимально допустимом уровне доходности*

$$\begin{cases} \sqrt{\sum_{i=1}^n w_i^2 \cdot \sigma_i^2 + 2 \sum_{i=1}^{n-1} \sum_{j=i+1}^n w_i \cdot w_j \cdot k_{ij} \cdot \sigma_i \cdot \sigma_j} \rightarrow \min \\ \sum_{i=1}^n w_i \cdot r_i > r_p \\ \sum_{i=1}^n w_i = 1 \\ w_i \geq 0 \end{cases}$$

*Максимизируем доходность при заданном уровне риска*

$$\begin{cases} \sum_{i=1}^n w_i \cdot r_i \rightarrow \max \\ \sqrt{\sum_{i=1}^n w_i^2 \cdot \sigma_i^2 + 2 \sum_{i=1}^{n-1} \sum_{j=i+1}^n w_i \cdot w_j \cdot k_{ij} \cdot \sigma_i \cdot \sigma_j} < \sigma_p \\ \sum_{i=1}^n w_i = 1 \\ w_i \geq 0 \end{cases}$$

## Использование портфельной теории в экспертной системе

Основой экспертной системы является база знаний о предметной области. База знаний содержит знания – совокупность информации об объектах предметной области и связях между ними. В большинстве случаев знания экспертной системы являются эвристиками и носят вероятностный характер, что отражает степень неуверенности в достоверности фактов и точности правил вывода.

В экспертных системах для представления знаний наиболее часто используются правила вывода, семантические сети и фреймы.

Представление знаний, основанное на правилах вывода, обычно построено на использовании выражений вида ЕСЛИ условие, ТО заключение, отражающих ход рассуждений человека-эксперта. Правила обеспечивают естественный способ описания предметной области, процесса принятия решений. В системе, основанной на правилах, знания отделены от программного кода, реализующего механизм вывода заключений. Такой способ организации системы дает возможность легко вносить изменения в базу знаний в процессе ее использования.

Семантические сети и фреймы используют для решения исследовательских задач искусственного интеллекта.

Таким образом, для решения задачи построения исследовательского прототипа экспертной системы наиболее подходящим способом представления знаний являются правила вывода.

Правила – это утверждения вида

$$A \leftarrow B_1, B_2, \dots, B_n,$$

где  $n \geq 0$ ,  $A$  – заголовок правила, последовательность  $B_i$ , – «тело» правила, причем элементы  $B_i$  могут представлять собой как факты, так и правила.

На естественном языке правила в общем виде могут быть представлены так:

Если

$$\text{Объект}_1 = \text{Значение}_1, \text{КД}_1 = \text{К}_1$$

...

$$\text{Объект}_k = \text{Значение}_k, \text{КД}_k = \text{К}_k$$

То Объект<sub>i</sub> = Значение<sub>i</sub>, КД<sub>i</sub> = K<sub>i</sub>,

где Если, То и КД – ключевые слова языка представления правил, используемого интерпретатором правил («машиной» вывода) в конкретной экспертной системе;

Объект<sub>i</sub> и Значение<sub>i</sub> – соответственно объект предметной области и его значение, КД<sub>i</sub> – степень (коэффициент) достоверности, вещественное число в диапазоне от 0 до 100, соответствующее степени уверенности (вероятности), что состояние объекта характеризуется указанным значением.

Кроме правил, база знаний экспертной системы содержит факты – информацию о текущем состоянии объектов предметной области.

Факты – это утверждения вида

Объект<sub>j</sub> = Значение<sub>j</sub>, КД<sub>j</sub> = K<sub>j</sub>

Факты появляются в базе знаний в процессе консультации – взаимодействия пользователя с экспертной системой, как результат ответов пользователя на вопросы экспертной системы, а также продуцируются самой экспертной системой (механизмом вывода) в процессе согласования фактов и правил.

Основой ЭС является база знаний. Помимо базы знаний (БЗ), содержащей правила логического вывода и факты, и механизма вывода (МВ), экспертная система содержит модули интерфейсов пользователя (ИП) и инженера по знаниям (ИИЗ). Интерфейс пользователя обеспечивает взаимодействие пользователя с экспертной системой в процессе консультации. Интерфейс инженера по знаниям имеет доступ к правилам базы знаний, что позволяет корректировать поведение экспертной системы. В исследовательском прототипе интерфейс пользователя может отсутствовать. Архитектура экспертной системы на правилах приведена на рис. 2.

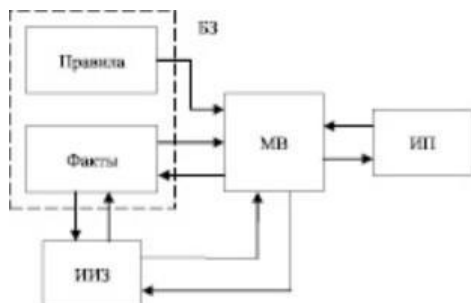


Рис.2 Архитектура ЭС на правилах

## Практическая часть

Исходя из проведенных исследований в теоретической части, можно поставить задачи для практической части:

- Выбрать финансовый API ценных бумаг для сбора данных в виде модуля сбора данных;
- Спроектировать модули советов и объяснений с подключенным к ней механизмом логического вывода на базе алгоритма составления инвестиционного портфеля по Марковицу;
- Спроектировать и развернуть базу знаний;
- Спроектировать и развернуть базу данных для хранения данных рынка;
- Спроектировать пользовательский интерфейс для режима функционирования с пользователем и экспертом;

Из задач практической части можно вынести следующие задачи программирования:

- Реализовать модуль получения данных по рынку по выбранному API;
- Реализовать сервис подключения к базе данных пользователей, регистрацию и авторизацию для сепарации логики приложения для эксперта и пользователя;
- Реализовать подключение и изменение в базе данных рынка;
- Реализовать модуль получения и изменения знаний в базе знаний;
- Реализовать модули советов и объяснений с подключенным к ней механизмом логического вывода на базе алгоритма составления инвестиционного портфеля по Марковицу;

## **Выбор стека технологий**

Поставленные практические задачи можно разделить на следующие области:

- Frontend технологии
  - Язык разметки
  - Язык программирования
  - Платформа, фреймворк или библиотека для пользовательского интерфейса
- Backend технологии
  - Язык описания/фреймворк межсервисного взаимодействия
  - Язык программирования
  - Платформа, фреймворк или библиотека
  - Базы данных
  - API для работы с данными рынка

В случае реализации данной программы были выбраны и использованы следующие технологии.

### **Язык программирования**

Язык C# заслужил большое уважение и популярность среди разработчиков самых разных программных продуктов. Последнюю пару лет C# играл важную роль в производстве устойчивых к сбоям продуктов - от настольных приложений до Web-сервисов, от высокоуровневых решений в автоматизации бизнес-процессов до программ системного уровня и от однопользовательских продуктов до корпоративных решений в сетевых распределенных средах.

Язык C# сертифицирован Международной организацией по стандартизации (ISO). Таким образом, эта разработка Microsoft имеет статус международных стандартов. Также, язык C# признан в качестве стандарта Европейской ассоциацией производителей компьютеров (ЕСМА).

Язык программирования C# имеет ряд преимуществ:

- подлинная объектная ориентированность (всякая языковая сущность претендует на то, чтобы быть объектом);
- компонентно-ориентированное программирование;
- безопасный (по сравнению с языками C и C++) код;
- унифицированная система типизации;



- поддержка событийно-ориентированного программирования;
- «родной» язык для создания приложений в среде .NET;
- объединение лучших идей современных языков программирования: Java, C++, Visual Basic и др.

Исходя с достоинств и с выше перечисленных преимуществ языка C#, в качестве языка для разработки системы был выбран именно этот объектно-ориентированный язык.

### Выбор платформы и фреймворков

В качестве языка платформы для реализации выбор был остановлен на WPF для фронтенда и ASP.NET Core для бэкенда как лучшие представители области для выбранного языка программирования.

**WPF** предоставляет комплексный набор функций разработки приложений, которые включают в себя язык разметки XAML, элементы управления, привязку к данным, макет, двухмерную и трехмерную графику, анимацию, стили, шаблоны, документы, мультимедиа, текст и типографические функции.

**ASP.NET Core** – это кроссплатформенный фреймворк с открытым исходным кодом для создания современных облачных приложений, таких как приложения интернета вещей и мобильные серверные части. ASP.NET Core приложения работают на платформе .NET Core, бесплатной кроссплатформенной среде выполнения приложений с открытым исходным кодом. Его архитектура была разработана таким образом, чтобы обеспечить оптимизированную среду разработки для приложений, которые развертываются в облаке или выполняются локально. Он состоит из модульных компонентов с минимальными накладными расходами, поэтому вы сохраняете гибкость при создании своих решений.

Для работы с базой данных в .Net Core принято использовать еще один фреймворк от Microsoft - Entity Framework Core.

**Entity Framework Core** — это современный модуль сопоставления "объект — база данных" для .NET. Он поддерживает запросы LINQ, отслеживание изменений, обновления и миграции схемы. EF Core работает с многими базами данных, включая базы данных SQL (локальные и в Azure), MSSQL, SQLite, MySQL, PostgreSQL и Azure Cosmos DB.

Для межсервисного взаимодействия использовался высокопроизводительный фреймворк gRPC.

**gRPC** - это современный высокопроизводительный фреймворк удаленного вызова процедур с открытым исходным кодом, который может работать в любой среде. Он может эффективно подключать службы в центрах обработки данных и между ними с помощью подключаемой поддержки для балансировки нагрузки, отслеживания, проверки работоспособности и аутентификации. Он также применим в последней миле распределенных вычислений для подключения устройств, мобильных приложений и браузеров к внутренним службам.

### **Выбор системы управления базами данных**

Выбор системы управления баз данных (СУБД) представляет собой сложную многопараметрическую задачу и является одним из важных этапов при разработке программного обеспечения. На сегодняшний день для построения информационных систем применяются различные СУБД, различающиеся как своими возможностями, так и требованиями к вычислительным ресурсам. При выборе системы управления баз данных необходимо выбрать базу данных, которая в наибольшей степени соответствует предъявляемым к информационной системе требованиям.

При выборе СУБД необходимо обратить внимание на следующие показатели:

- Производительность и масштабируемость;
- Максимальное количество пользователей, одновременно обращающихся к БД;
- Серверная операционная система;
- Надежность;
- Производительность труда разработчиков.

Такая СУБД как Microsoft SQL Server включает профессиональное программное обеспечение для управления базами данных уровня предприятия. Несколько конкурентов, таких как MySQL, разработали подобное программное обеспечение в последние годы, но Microsoft SQL Server проще в использовании и имеет больше функций. Например, полная поддержка триггеров поддерживается в продукте Microsoft. MySQL недавно представил триггеры, но они не полностью поддерживаются. Программное обеспечение, предлагаемое Microsoft, также предлагает тесную интеграцию с платформой .NET, что

является одним из главных аргументов при выборе данной СУБД для реализации приложения.

### **Выбор API для сбора данных с рынка**

**API** (Application Programming Interface или интерфейс программирования приложений) — это совокупность инструментов и функций в виде интерфейса для создания новых приложений, благодаря которому одна программа будет взаимодействовать с другой. Это позволяет разработчикам расширять функциональность своего продукта и связывать его с другими. Большинство крупных компаний разрабатывают API для клиентов или для внутреннего использования.

При выборе **API** для работы с данными рынка рассматривались два варианта:

1. **Alpha Vantage API**
2. **Yahoo Finance API**

В связи с прекращением официальной поддержки последней был выбран **Alpha Vantage API**.

**Alpha Vantage** предоставляет данные о финансовом рынке корпоративного уровня с помощью набора мощных и удобных для разработчиков API. От традиционных классов активов (например, акций и ETF) до экономических показателей, от курсов иностранных валют до криптовалют, от фундаментальных данных до технических индикаторов, Alpha Vantage - это универсальный источник для получения исторических данных о мировых рынках в режиме реального времени, предоставляемых через RESTful stock API, Excel и Google Sheets.

## **Алгоритмизация и разработка программного обеспечения**

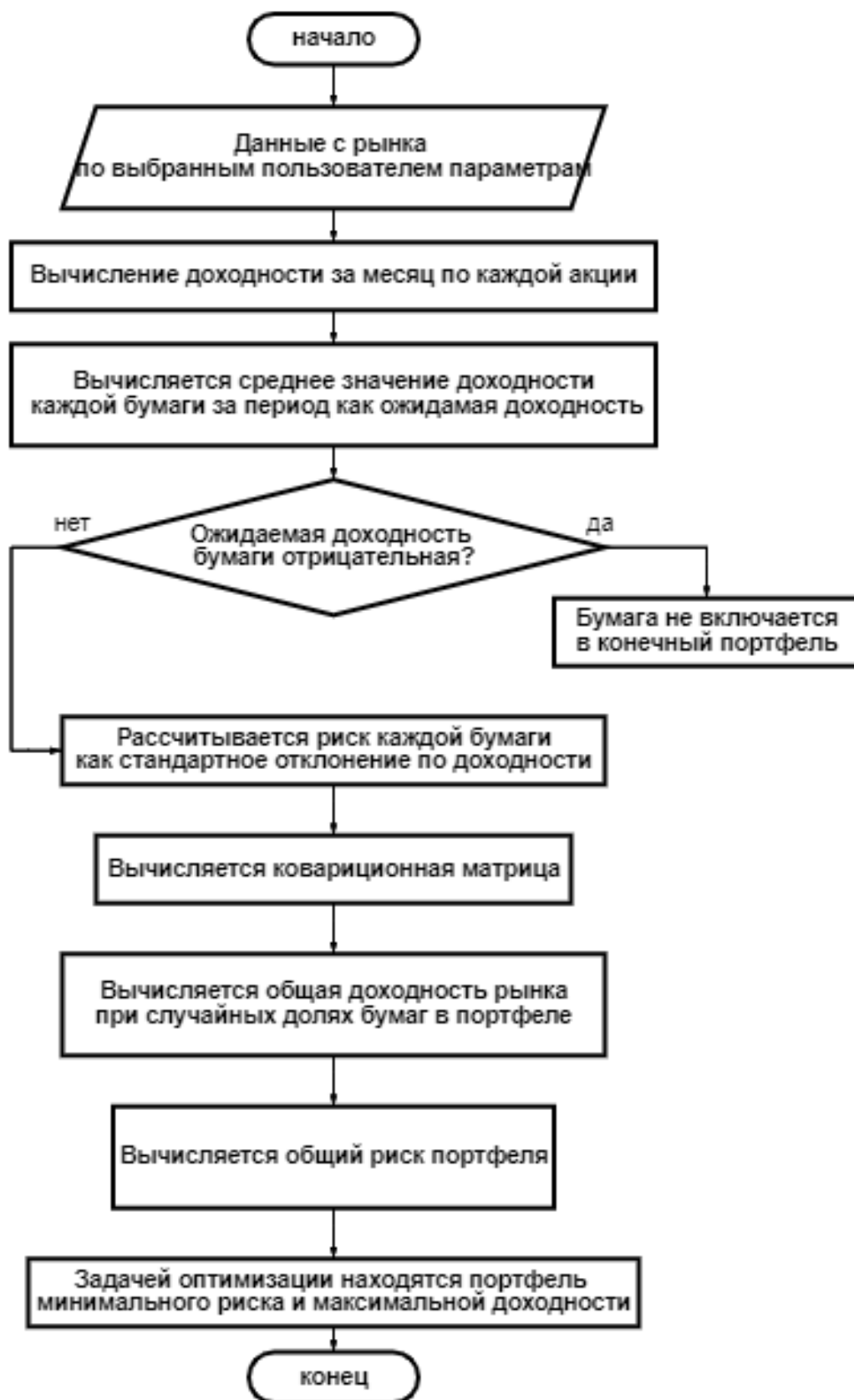
Главной задачей алгоритмизации в данном ПО стоит задача составления портфелей максимальной доходности и минимального риска.

Составление инвестиционного портфеля по Марковицу делят на следующие этапы:

1. Получение данных рынка (берутся данные как минимум за год)
2. Считается доходность за каждый месяц по формуле натурального логарифма.. Для простоты не будем учитываться эффект дивидендов. Если прибыль отрицательная, то бумага выбрасывается из портфеля.
3. Высчитывается риск каждой акции как стандартное отклонение по прибыли.
4. Создается ковариационная матрица для бумаг.
5. Идет расчет общей доходности портфеля. При случайных значениях долей бумаг в портфеле. Высчитывается средневзвешенное значение доходности отдельных бумаг.
6. Расчет общего риска портфеля.
7. Из полученных данных составляется портфели минимального риска и максимальной доходности.

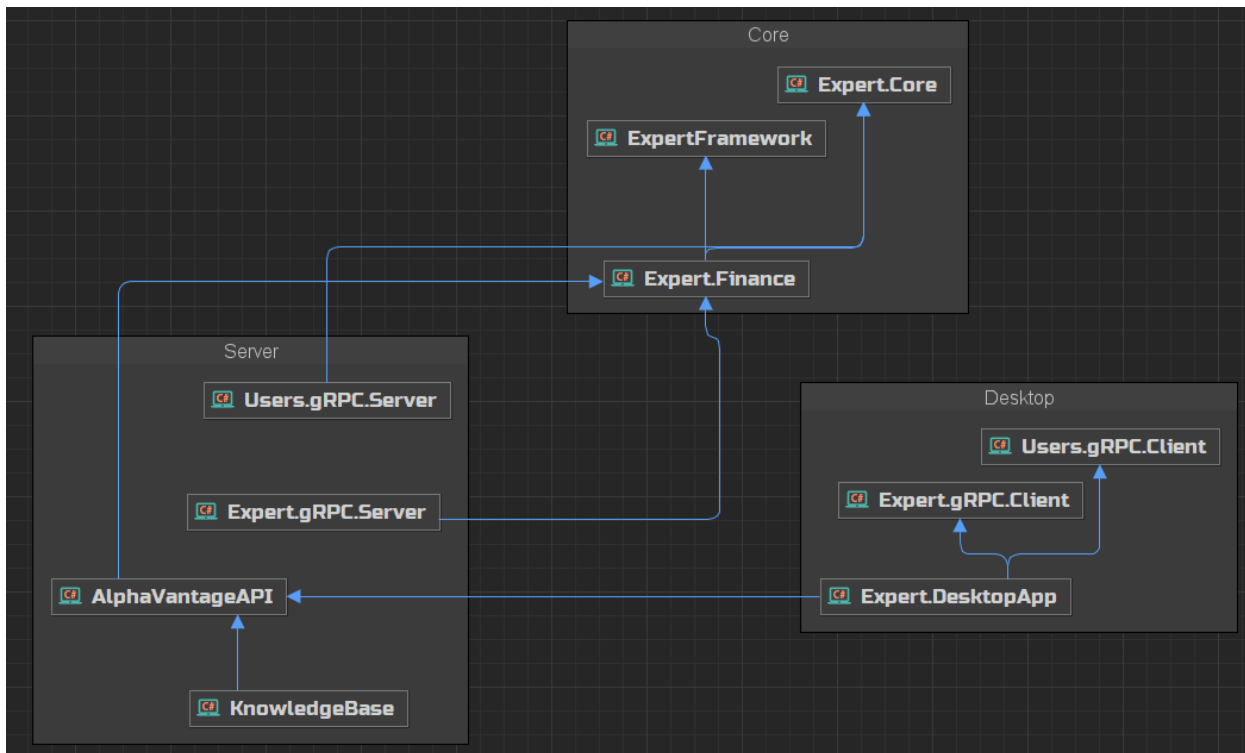
После составления портфелей портфели добавляются в базу знаний для скорейшей оперативной обработки.

## Блок-схема алгоритма



## Архитектура программного обеспечения

Архитектуру ПО можно визуализировать в виде диаграммы зависимости проектов приложения. На диаграмме четко изображена зависимость модулей и сервисов.

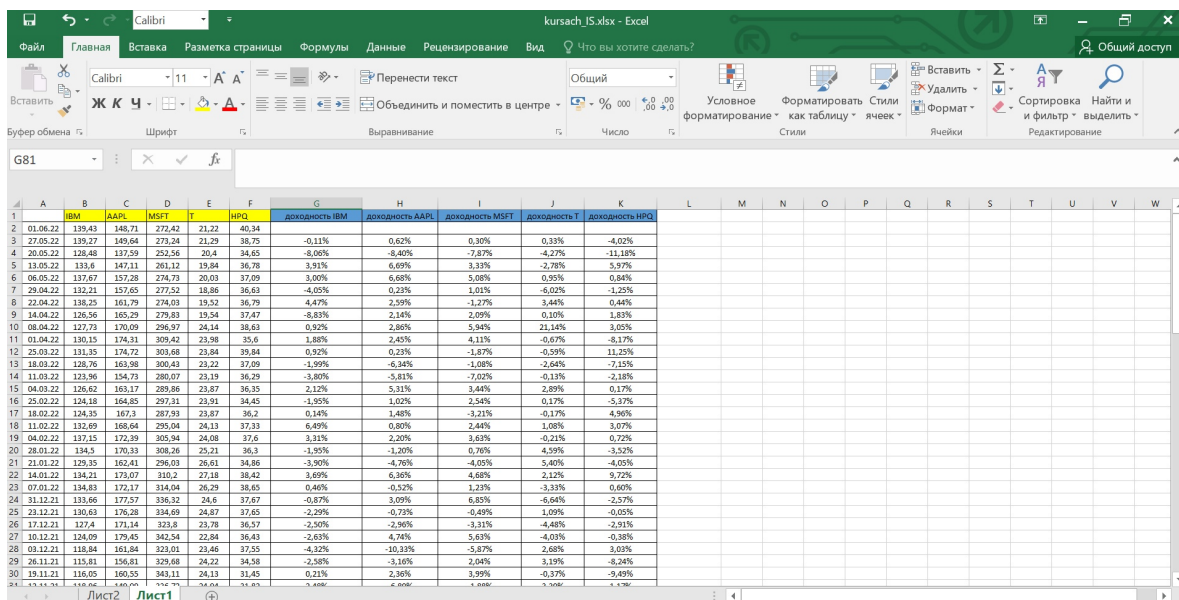


Приложение 1. Архитектура ПО

## Тестирование ПО

Для тестирования приложения был использован готовый шаблон для вычисления портфеля в программе Excel.

Были взяты бумаги T, AAPL, MSFT, IBM, HPQ.



	A	B	C	D	E	F	G	H	I	J	K
1		IBM	AAPL	MSFT	T	HPQ	доходность IBM	доходность AAPL	доходность MSFT	доходность T	доходность HPQ
2	01.06.22	139.43	148.71	272.42	21.22	40.34					
3	27.05.22	139.27	149.64	273.24	21.29	38.75	-0.11%	0.62%	0.30%	0.33%	-4.02%
4	20.05.22	128.48	137.59	252.56	20.4	34.65	-8.06%	-8.40%	-7.87%	-4.27%	-11.18%
5	13.05.22	133.6	147.11	261.12	19.84	36.78	3.91%	6.69%	3.33%	-2.78%	5.97%
6	06.05.22	137.67	157.28	274.73	20.03	37.09	3.00%	6.68%	5.08%	0.95%	0.84%
7	28.04.22	132.21	157.69	277.52	18.86	36.63	-4.05%	0.23%	1.03%	-9.02%	-1.25%
8	22.04.22	138.25	161.79	274.03	19.52	36.79	4.47%	2.59%	-1.27%	3.44%	0.44%
9	14.04.22	126.56	165.29	279.83	19.54	37.47	-8.83%	2.14%	2.09%	0.10%	1.83%
10	08.04.22	127.73	170.09	296.97	24.14	38.63	0.92%	2.86%	5.94%	21.34%	3.05%
11	01.04.22	130.15	174.31	299.42	23.98	39.6	1.80%	2.45%	4.11%	-0.07%	-8.17%
12	25.03.22	131.35	174.72	303.68	23.84	39.84	0.92%	0.23%	-1.87%	-0.59%	11.25%
13	18.03.22	128.76	163.98	300.43	23.22	37.09	-1.99%	-6.34%	-1.08%	-2.64%	-7.15%
14	11.03.22	123.96	154.73	280.07	23.19	36.29	-3.80%	-5.81%	-7.02%	-0.13%	-2.18%
15	04.03.22	126.62	163.17	289.86	23.97	36.35	2.12%	5.31%	3.44%	2.05%	0.17%
16	25.02.22	124.18	164.85	297.31	23.91	34.45	-1.95%	1.02%	2.54%	0.17%	-5.37%
17	18.02.22	124.35	167.3	287.93	23.87	36.2	0.14%	1.48%	-3.21%	-0.17%	4.96%
18	11.02.22	132.69	168.64	295.04	24.13	37.33	6.49%	0.80%	2.44%	1.08%	3.07%
19	04.02.22	137.15	172.39	305.94	24.08	37.6	3.11%	2.20%	3.63%	-0.21%	0.72%
20	28.01.22	134.5	170.31	308.26	25.21	38.3	-1.95%	-1.20%	0.70%	4.99%	-3.52%
21	21.01.22	129.35	162.41	296.03	26.61	34.86	-3.90%	-4.76%	-4.05%	5.40%	-4.05%
22	14.01.22	134.21	173.07	310.2	27.18	38.42	3.69%	6.36%	4.88%	2.12%	9.72%
23	07.01.22	134.63	172.17	314.04	26.29	38.65	0.46%	-0.52%	1.21%	-3.31%	0.60%
24	31.12.21	133.66	177.57	336.32	24.4	37.67	-0.97%	3.09%	6.85%	-6.64%	-2.57%
25	23.12.21	130.63	176.28	334.69	24.87	37.65	-2.29%	-0.73%	-0.49%	1.09%	-0.05%
26	17.12.21	127.4	171.14	323.8	23.78	36.57	-2.50%	-2.96%	-3.31%	-4.48%	-2.91%
27	10.12.21	124.09	179.45	342.54	22.84	36.43	-2.63%	4.74%	5.63%	-0.69%	-0.38%
28	03.12.21	118.94	161.94	323.01	23.46	37.55	-4.32%	-10.33%	-5.87%	2.68%	3.03%
29	26.11.21	115.81	156.81	325.68	24.22	34.58	-2.58%	-3.16%	2.04%	3.19%	-8.24%
30	19.11.21	116.05	160.55	343.11	24.13	31.45	0.21%	2.36%	3.99%	-0.37%	-9.49%

## Приложение 2. Окно тестовой программы

Файл

Главная

Вставка

Разметка страницы

Формулы

Данные

Рецензирование

Вид

Что вы хотите сделать?

Общий доступ

Вставить

Ж К Ч

Шрифт

Выравнивание

Число

Общий

Условное форматирование

Форматировать как таблицу

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

Стили

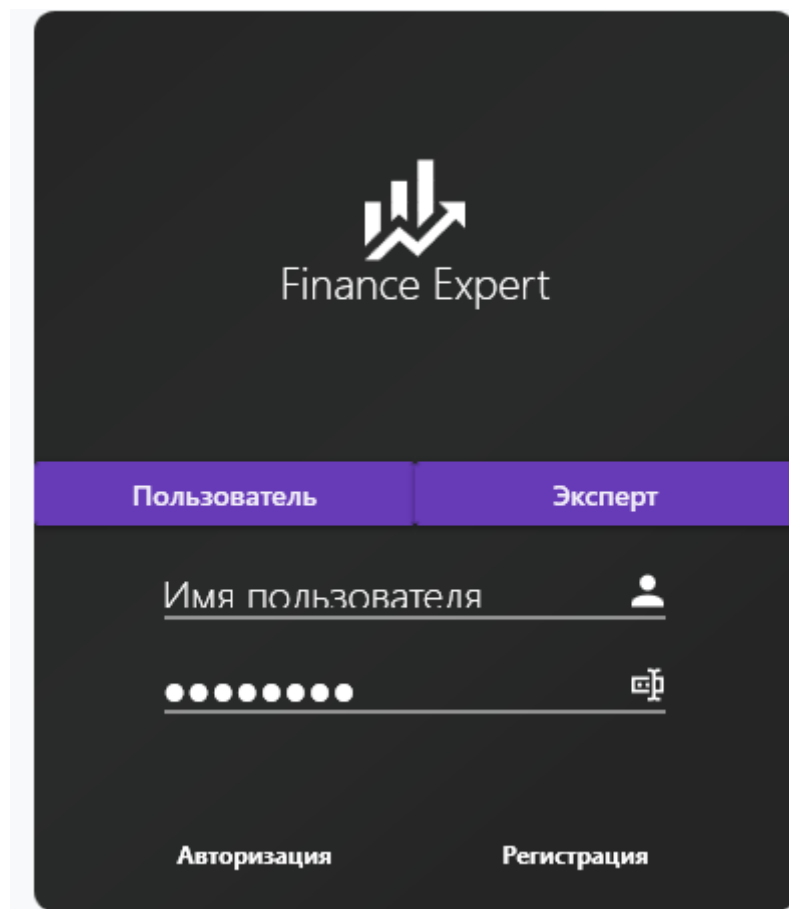
Стили

Стили

С

## Приложение 3. Результат работы тестовой программы

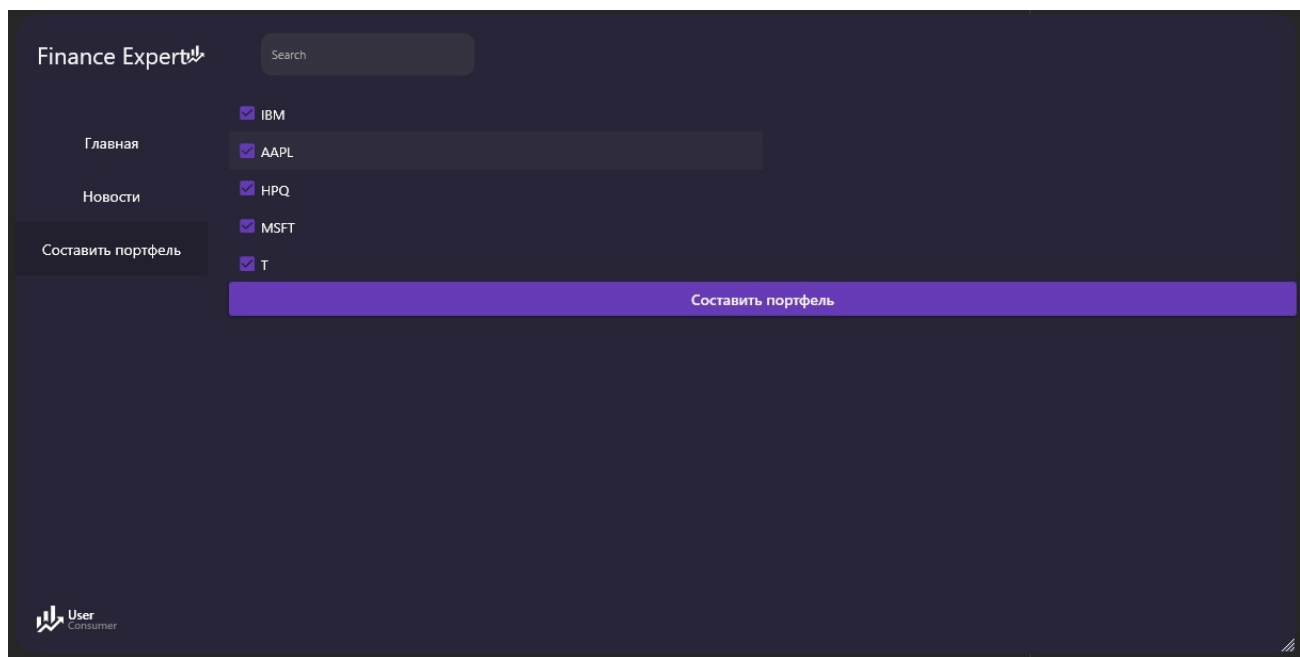
## Результат работы программы



The image shows a login window for a program named "Finance Expert". The window has a dark background. At the top center is a logo consisting of a stylized bar chart with an upward-pointing arrow, followed by the text "Finance Expert". Below the logo is a horizontal bar with two tabs: "Пользователь" (User) and "Эксперт" (Expert). The "Пользователь" tab is selected. Below the tabs are two input fields. The first field is labeled "Имя пользователя" (Username) and has a user icon on the right. The second field is for a password, indicated by a series of dots and a password icon on the right. At the bottom of the window are two buttons: "Авторизация" (Login) and "Регистрация" (Registration).

Приложение 4. Окно авторизации пользователя





Приложение 5. Окно приложения приема параметров



Приложение 6. Окно приложения после получения результатов

## **Общие выводы**

1. Была реализована экспертная система со всеми необходимыми модулями и сервисами для составления оптимального портфеля по заданным параметрам.
2. Анализируя изменения рынка в реальном времени можно прогнозировать поведение рынка ценных бумаг и корректировать операции финансового процесса.
3. Рассмотрен подход к формированию базы данных, содержащей консолидированную информацию, предназначенную для принятия управляющих решений.

## Список литературы

1. Гаврилова Т. А., Хорошевский В. Ф. Базы знаний интеллектуальных систем. Учебник. — СПб.: Питер, 2000.
2. Туккель И.Л. Управление инновационными проектами. СПб.: БХВ-Петербург, 2017. 416 с.
3. Илларионов А.В., Клименко Э.Ю. Портфель проектов: Инструмент стратегического управления предприятием. М.: Альпина Паблишер, 2013. 312 с.
4. Жилиева А.С., Культин Н.Б. Универсальная система критериев оценки инновационных проектов // Инновации. 2017. № 10. С. 100–104.
5. Аблязов В.И., Деттер Г.Ф., Симонцев С.Н., Черняк В.С. Экспертиза инновационных проектов // Научнотехнические ведомости СПбПУ. 2011. № 3. С. 184–188.
6. Яблуновский М.А. Архитектура автоматизированной системы экспертизы инновационных проектов // Научно-технические ведомости СПбПУ. 2011. № 3. С. 134–138.
7. Бланк И.А. Инвестиционный менеджмент. - Киев.: МП "ИТЕМ", 1995. - 448 с.
8. Зимин И.А. Реальные инвестиции: учеб. пособие. - М.: ТАНДЕМ, 2000. - 304 с.
9. Крейнина М.Н. Финансовый менеджмент: учеб. пособие. - М.: "Дело и Сервис", 2001. - 400 с.
10. Шарп У. Инвестиции/ пер. с англ. - М.: ИНФРА-М, 1998. - 1028 с.
11. Курс экономики: учеб. /под ред. Б.А. Райзенберга - М.: ИНФРА-М, 2001. - 716 с.
12. Липсиц И.В. Экономический анализ реальных инвестиций: учеб. пособие. - М.: Экономистъ, 2004. - 347 с.
13. К. Хорсдал. Микросервисы на платформе .NET - Изд: Питер, 2018. - 328 с.

## Приложения

### Код составления портфеля и его оптимизации по полученным данным рынка

```
using AlphaVantageAPI;
using Expert.Finance.Extensions;
using MathOptimization;
using static System.MathF;
namespace Expert.Finance.Markowitz;

public static class MarkowitzPortfolio
{
    #region Utils

    private static double[,] InitSLAUTableMax(List<float>
profit)
    {
        var k = profit.Count;
        var a = new double[2, k + 1];
        for (var i = 0; i < k + 1; i++)
            a[0, i] = 1;
        a[1, 0] = 0;
        for (var i = 0; i < k; i++)
            a[1, i + 1] = profit[i];
        return a;
    }

    private static float GetRiskByAverage(Portfolio
randomPortfolio, double[,] covMatrix)
    {
        var partsMatrix = new double[1,
randomPortfolio.StocksList.Count];
        var partsMatrixT = new
double[randomPortfolio.StocksList.Count, 1];

        for (var i = 0; i < randomPortfolio.StocksList.Count;
i++)
```

```

        partsMatrixT[i, 0] = partsMatrix[0, i] =
randomPortfolio.StocksList[i].Part;

        return
Sqrt((float)partsMatrix.Multiplication(covMatrix).Multiplicat
ion(partsMatrixT)[0, 0]); // 1xN * NxN * Nx1 = 1x1
    }

    private static double[,]
GetClosesMatrix(Dictionary<string, List<decimal>>
dataByStock)
    {
        var iterator = 0;
        var closesData = new double[dataByStock.Keys.Count,
dataByStock.First().Value.Count];
        foreach (var sym in dataByStock.Keys)
        {
            for (var i = 0; i < dataByStock[sym].Count; i++)
                closesData[iterator, i] =
(double)dataByStock[sym][i];
            iterator++;
        }

        return closesData;
    }

    private static Dictionary<string, List<decimal>>
IncomeByStock(Dictionary<string, List<decimal>> dataByStock)
    {
        var incomeByStock = new Dictionary<string,
List<decimal>>();
        foreach (var symbol in dataByStock.Keys)
        {
            incomeByStock.Add(symbol, new List<decimal>());
            for (var i = 1; i <
dataByStock.First().Value.Count; i++)

```

```

        incomeByStock[symbol]

        .Add((decimal)Log((float)(dataByStock[symbol][i] /
dataByStock[symbol][i - 1]))); // loss of accuracy
    }

    return incomeByStock;
}

#endregion

public static IEnumerable<Portfolio>
CreatePortfolios(IEnumerable<StockData> data)//, float
minProfit, float maxRisk
{
    var result = new List<Portfolio>();

    var stockClosesData = data.GetCloseData();

    var incomeByStock = IncomeByStock(stockClosesData);

    var expectedProfitability =
        incomeByStock.Keys
            .ToDictionary(symbol => symbol, symbol =>
(float)incomeByStock[symbol].Average());
    var riskByStock = incomeByStock.Keys
        .ToDictionary(sym => sym, sym =>
incomeByStock[sym].StandardDeviation());

    var closesMatrix = GetClosesMatrix(stockClosesData);

    alglib.covm(closesMatrix, out var covMatrix);

    using var random = RandomExtensions
        .GetRandomValuesWithSum(0, 1,

```

```

expectedProfitability.Keys.Count)
    .GetEnumerator();

    var randomPortfolio = new Portfolio();

    foreach (var stocks in expectedProfitability.Keys)
    {
        randomPortfolio.StocksList.Add(new
Portfolio.PortfolioStock(stocks, random.Current));
        random.MoveNext();
    }

    randomPortfolio.PossibleIncome =
randomPortfolio.StocksList
        .Sum(s => expectedProfitability[s.Stock] *
s.Part);

    randomPortfolio.Risk =
GetRiskByAverage(randomPortfolio, covMatrix);

    var max =
MaxProfitPortfolio(randomPortfolio.StocksList.Select(s =>
s.Part).ToList());

    var maxProfitPortfolio = new Portfolio();

    foreach (var stocks in expectedProfitability.Keys)
    {
        maxProfitPortfolio.StocksList.Add(new
Portfolio.PortfolioStock(stocks, random.Current));
        random.MoveNext();
    }

    for (int i = 0; i < max.Count; i++)
        maxProfitPortfolio.StocksList[i].Part = max[i];

```



```

        maxProfitPortfolio.PossibleIncome =
randomPortfolio.StocksList
            .Sum(s => expectedProfitability[s.Stock] *
s.Part);

        maxProfitPortfolio.Risk =
GetRiskByAverage(randomPortfolio, covMatrix);

        result.Add(randomPortfolio);
        result.Add(maxProfitPortfolio);

        return result;
    }

    private static List<float> MaxProfitPortfolio(List<float>
parts)
    {
        double[,] table = InitSLAUTableMax(parts);

        var result = new double[parts.Count];

        var S = new Simplex(table);
        S.Calculate(result);

        return Enumerable.Cast<float>(result).ToList();
    }
}

```