

Министерство образования Республики Беларусь
Учреждение образования «Белорусский государственный университет
информатики и радиоэлектроники»

Факультет инженерно-экономический
Кафедра экономической информатики
Дисциплина «Программирование сетевых приложений»

«К ЗАЩИТЕ ДОПУСТИТЬ»
Руководитель курсовой работы
Ассистент кафедры ЭИ
_____._____.2025 Ю.В.Сильванович

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
к курсовому проекту
на тему:
**«ПРОГРАММНОЕ СРЕДСТВО ФОРМИРОВАНИЯ
ИНВЕСТИЦИОННОГО ПОРТФЕЛЯ КРИПТОВАЛЮТ НА ОСНОВЕ
МОДЕЛИ МАРКОВИЦА»**

БГУИР КР 6-05-0611-01 056 ПЗ

Выполнил студент группы 372303
Давидюк Максим Викторович

(подпись студента)
Курсовой проект представлен на
проверку _____._____.2025

(подпись студента)

Минск 2025

РЕФЕРАТ

БГУИР КР 6-05-0611-0 056 ПЗ

Давидюк, М.В. Программное средство формирования инвестиционного портфеля криптовалют на основе модели Марковица/ М.В. Давидюк. – Минск: БГУИР, 2025. – 51с.

Пояснительная записка 51 с., 42 рис., 10 источников, 2 приложения

ФОРМИРОВАНИЯ ИНВЕСТИЦИОННОГО ПОРТФЕЛЯ КРИПТОВАЛЮТ НА ОСНОВЕ МОДЕЛИ МАРКОВИЦА С МОДУЛЕМ АНАЛИТИКИ, МОДЕЛИ *UML*, *IDEF0*, *BPMN*, СХЕМЫ АЛГОРИТМОВ, ПРОГРАММНОЕ СРЕДСТВО

Цель курсовой работы: автоматизировать процесс формирования оптимального криптовалютного портфеля на основе модели Марковица. Разработанное программное средство повысит эффективность принятия решений за счет использования методов математического моделирования.

Методология проведения работы: в процессе решения поставленных задач использованы принципы системного подхода, аналитические методы, методы компьютерной обработки данных и математического моделирования. Также выполнена визуализация результатов анализа.

Результаты работы: выполнена постановка задачи и определены основные методы ее решения; в ходе объектного моделирования системы построен ряд *UML*-диаграмм; разработаны модели бизнес-процессов предметной области на основе нотаций *IDEF0*; описаны основные алгоритмы работы программного средства; разработана информационная модель системы, представленная в виде схемы базы данных; реализован программный продукт с возможностью анализа рисков и оценки доходности портфеля; проведено тестирование программного средства, подтвердившее его соответствие функциональным требованиям.

Программный продукт разработан на языке *Java* с применением *IntelliJ IDEA Ultimate Edition 2024.3* и СУБД *PostgreSQL*.

Область применения результатов: разработанное программное средство может использоваться инвесторами, аналитиками и финансовыми консультантами для формирования и оптимизации инвестиционного портфеля криптовалют, также помогать принимать решения в области криптовалютных инвестиций.

СОДЕРЖАНИЕ

| | |
|---|----|
| Введение..... | 5 |
| 1 Анализ литературных источников и программных решений | 6 |
| 1.1 Описание предметной области | 6 |
| 1.2 Обзор программных аналогов | 8 |
| 2 Моделирование предметной области и разработка требований к программному средству..... | 12 |
| 2.1 Анализ и формализация бизнес-процессов предметной области | 12 |
| 2.2 Анализ требований к программному средству и разработка их спецификации | 17 |
| 2.3 Образ предлагаемого решения | 19 |
| 3 Проектирование и разработка программного средства | 22 |
| 3.1 Архитектурные решения и технологии реализации программного средства..... | 22 |
| 3.2 Проектирование и разработка пользовательского интерфейса..... | 25 |
| 3.3 Разработка модели данных..... | 27 |
| 3.4 Описание статических и динамических аспектов поведения программных объектов | 31 |
| 3.5 Разработка и описание алгоритмов, реализующих бизнес-логику программного средства | 31 |
| 3.6 Механизмы обеспечения информационной безопасности | 32 |
| 4 Тестирование и проверка работоспособности программного средства..... | 33 |
| 5 Руководство по установке (развертыванию) и использованию программного средства..... | 33 |
| 5.1 Руководство по установке (развертыванию) программного средства .. | 33 |
| 5.2 Руководство пользователя..... | 33 |
| Заключение | 34 |
| Список использованных источников | 35 |
| Приложение А | 36 |
| (обязательное)..... | 36 |
| Отчет о проверке на заимствование в системе «Антиплагиат» | 36 |
| Приложение Б | 36 |
| (обязательное)..... | 36 |
| Листинг кода алгоритмов, реализующих основную бизнес-логику..... | 36 |

ВВЕДЕНИЕ

1 АНАЛИЗ ЛИТЕРАТУРНЫХ ИСТОЧНИКОВ И ПРОГРАММНЫХ РЕШЕНИЙ

1.1 Описание предметной области

В современном мире криптовалюты становятся все более популярным инструментом для инвестиций, привлекая как частных инвесторов, так и крупные финансовые организации. Однако высокая волатильность криптовалютного рынка создает значительные риски, что требует от инвесторов применения научно обоснованных подходов к формированию инвестиционных портфелей. Одним из таких подходов является модель Марковица, также известная как теория портфеля, которая позволяет оптимизировать соотношение доходности и риска в инвестициях.

Модель Марковица была предложена Гарри Марковицем в 1952 году и с тех пор стала фундаментальной основой для управления финансовыми активами. Согласно этой модели, инвесторы стремятся максимизировать ожидаемую доходность при заданном уровне риска или минимизировать риск при фиксированной доходности. Для этого необходимо учитывать не только доходность и риск каждого отдельного актива, но и их корреляцию с другими активами в портфеле. Таким образом, диверсификация становится ключевым элементом успешной стратегии инвестирования [1].

Криптовалютный рынок характеризуется уникальными особенностями, которые делают применение модели Марковица особенно актуальным. Во-первых, криптовалюты демонстрируют высокую волатильность, что увеличивает как потенциальную доходность, так и уровень риска. Во-вторых, корреляция между различными криптовалютами может значительно варьироваться, что открывает возможности для эффективной диверсификации. В-третьих, быстрое развитие технологий и появление новых цифровых активов создают динамичную среду, требующую постоянного пересмотра стратегий инвестирования.

Процесс формирования инвестиционного портфеля криптовалют на основе модели Марковица можно условно разделить на несколько этапов. Первый этап – сбор данных. Необходимо получить исторические данные о ценах выбранных криптовалют за определенный период времени. Эти данные используются для расчета ожидаемой доходности и стандартного отклонения (меры риска) каждого актива, а также для определения корреляции между ними. Источниками данных могут служить криптобиржи, такие как Binance, Bybit или Coinbase, а также специализированные API-сервисы.

Второй этап – расчет параметров портфеля. На этом этапе применяются математические методы, такие как оптимизация с использованием квадратичного программирования, чтобы найти оптимальное распределение средств между активами. Результатом является так называемая "эффективная граница" – набор портфелей, обеспечивающих максимальную доходность при

каждом уровне риска. Инвестор может выбрать портфель, соответствующий его аппетиту к риску.

Третий этап – мониторинг и ребалансировка портфеля. Криптовалютный рынок характеризуется высокой изменчивостью, поэтому регулярное обновление данных и корректировка портфеля являются необходимыми условиями для поддержания его оптимальности. Это может включать продажу переоцененных активов и покупку недооцененных, чтобы вернуть портфель к целевому распределению.

Разработка программного средства для автоматизации этих процессов имеет множество преимуществ. Во-первых, автоматизация расчетов значительно снижает вероятность ошибок, связанных с ручной обработкой данных. Это особенно важно в условиях высокой волатильности криптовалютного рынка, где даже небольшие погрешности могут привести к значительным потерям. Во-вторых, программное средство позволяет быстро анализировать большие объемы данных, что существенно ускоряет процесс принятия решений.

В-третьих, наличие аналитического модуля дает возможность глубже исследовать поведение рынка и выявлять скрытые закономерности. Например, инвестор может проанализировать, как изменение макроэкономических факторов влияет на корреляцию между криптовалютами, или оценить влияние новостей на динамику цен. Это помогает принимать более обоснованные решения и адаптироваться к меняющимся рыночным условиям.

Четвертым преимуществом является возможность гибкой настройки системы под конкретные цели и ограничения инвестора. Программное средство может учитывать такие параметры, как минимальный и максимальный размер инвестиций в каждый актив, ограничения по уровню риска или требования к ликвидности портфеля. Это делает его универсальным инструментом для широкого круга пользователей – от начинающих трейдеров до профессиональных управляющих активами.

Наконец, внедрение такого программного обеспечения способствует повышению конкурентоспособности инвесторов на криптовалютном рынке. Современные технологии позволяют не только автоматизировать рутинные процессы, но и предоставляют доступ к передовым методам анализа данных, что создает дополнительные преимущества перед конкурентами.

Таким образом, разработка программного средства для формирования инвестиционного портфеля криптовалют на основе модели Марковица представляет собой важную и актуальную задачу. Она позволяет инвесторам эффективно управлять рисками, максимизировать доходность и адаптироваться к динамичным условиям криптовалютного рынка. Эти преимущества делают систему незаменимым инструментом для успешного инвестирования в цифровые активы.

1.2 Обзор программных аналогов

В современном мире управления инвестициями ключевым фактором успеха является использование передовых технологий и программных решений для анализа и оптимизации портфелей. Особенно это актуально для криптовалютного рынка, который характеризуется высокой волатильностью и динамичностью. Программные средства, разработанные для управления криптовалютными активами, позволяют не только автоматизировать процессы отслеживания цен и расчетов доходности, но и предоставляют инструменты для минимизации рисков и максимизации прибыли. Такие системы способствуют более эффективному управлению инвестиционными ресурсами, обеспечивают прозрачность данных и помогают инвесторам принимать обоснованные решения на основе актуальной информации. Однако выбор подходящего программного решения зависит от конкретных задач инвестора, таких как уровень контроля над портфелем, возможность интеграции с биржами и поддержка сложных моделей оптимизации, например, модели Марковица.

Рассмотрим такой программный аналог как «Portfolio Performance». «Portfolio Performance» – это открытая платформа для управления финансовыми портфелями, которая поддерживает как традиционные акции и облигации, так и криптовалюты. Она особенно популярна среди инвесторов, предпочитающих гибкость и контроль над своими данными [2].

Рассмотренная программа предоставляет следующие основные функции:

- расчет доходности и рисков портфеля с использованием различных методов;
- импорт данных из CSV-файлов или API-интерфейсов;
- поддержка пользовательских метрик для анализа портфеля;
- создание графиков и диаграмм для визуализации данных;
- интеграция с моделями оптимизации, такими как модель Марковица;
- экспорт отчетов в различные форматы для дальнейшего анализа.

На рисунке 1.1 представлен интерфейс программы «Portfolio Performance».

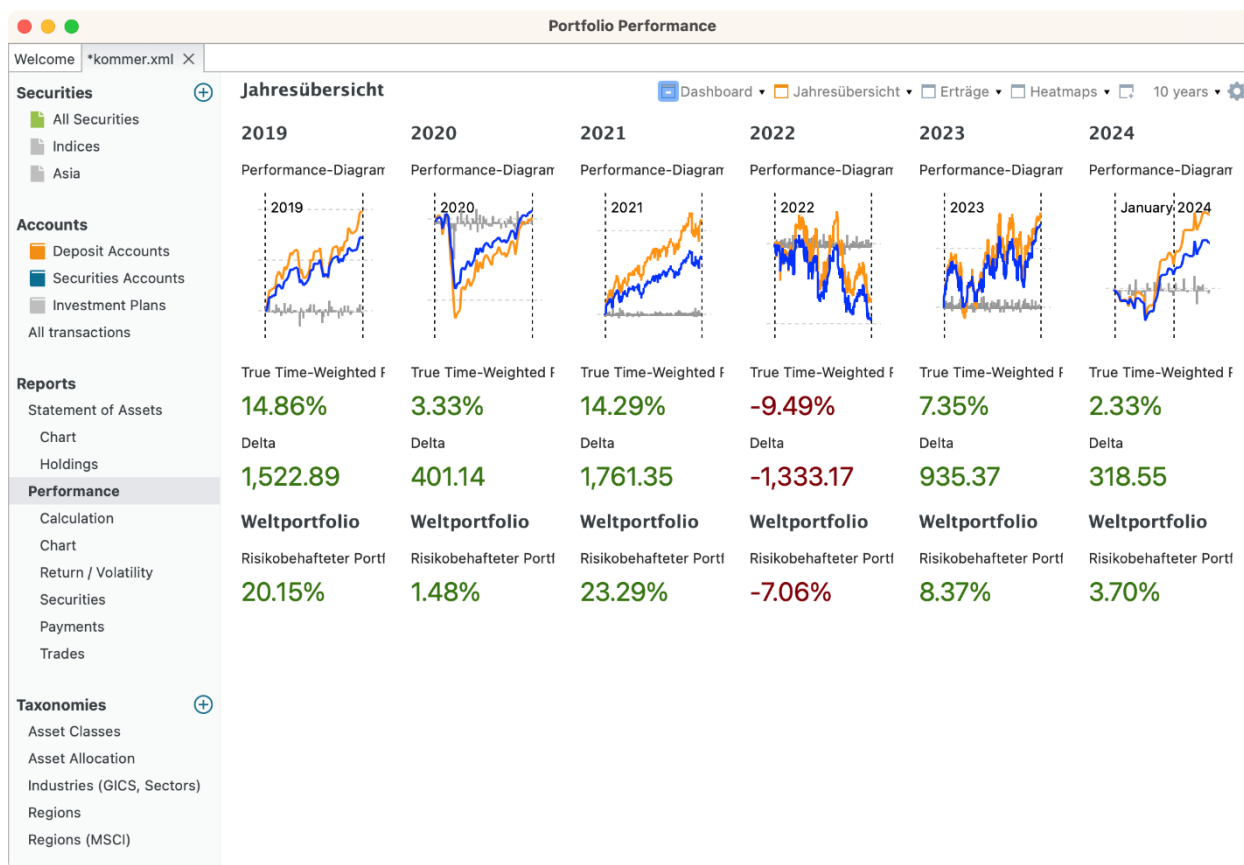


Рисунок 1.1 – Интерфейс «Portfolio Performance»

«Portfolio Performance» является наиболее подходящим аналогом, так как позволяет интегрировать пользовательские модели оптимизации, включая модель Марковица. Это делает её идеальным выбором для инвесторов, стремящихся к научно обоснованному управлению портфелем.

Теперь рассмотрим такую программу как «CoinStats». Это многофункциональное приложение для отслеживания криптовалютных портфелей, которое предлагает инструменты для анализа и управления активами. Программа поддерживает интеграцию с более чем 300 криптобиржами и кошельками, что делает её универсальным решением для инвесторов [3].

Данная программа предоставляет следующие основные функции:

- отслеживание портфеля в реальном времени;
- анализ доходности и рисков по каждому активу;
- интеграция с популярными биржами (Binance, Coinbase, Kraken и др.);
- генерация отчетов о доходности и изменении стоимости активов;
- поддержка мультивалютного учета (криптовалюты, фиатные валюты);
- уведомления о значительных изменениях цен или событиях на рынке;
- аналитика рыночных трендов и новостей.

Интерфейс программы «CoinStats» представлен на рисунке 1.2.

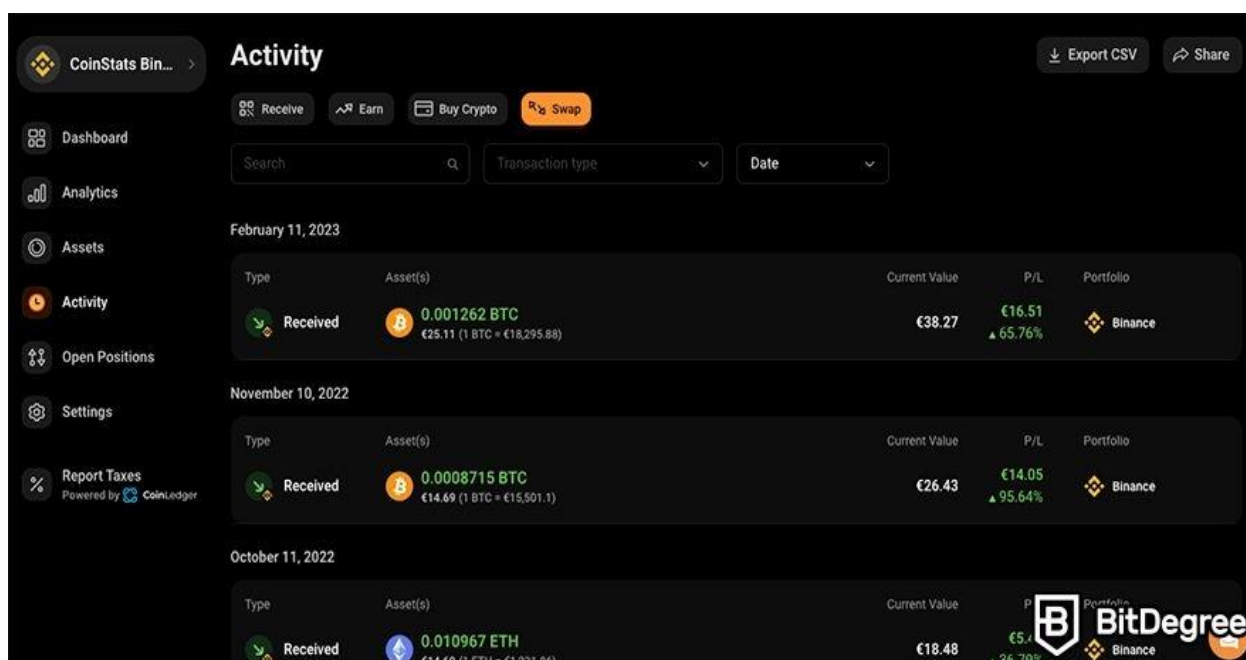


Рисунок 1.2 – Интерфейс «CoinStats»

«CoinStats» подходит для начинающих и опытных инвесторов благодаря своей простоте и широкой функциональности. Хотя она не предоставляет встроенной реализации модели Марковица, её инструменты для анализа доходности и рисков могут быть полезны для предварительной оценки портфеля.

Рассмотрим ещё один аналог «CryptoCompare». Данная платформа представляет аналитические данные о криптовалютах и их взаимосвязях. Хотя она не является полноценным инструментом для управления портфелем, её данные могут быть использованы для создания собственных моделей оптимизации. На рисунке 1.3 изображен интерфейс платформы «CryptoCompare» [4].

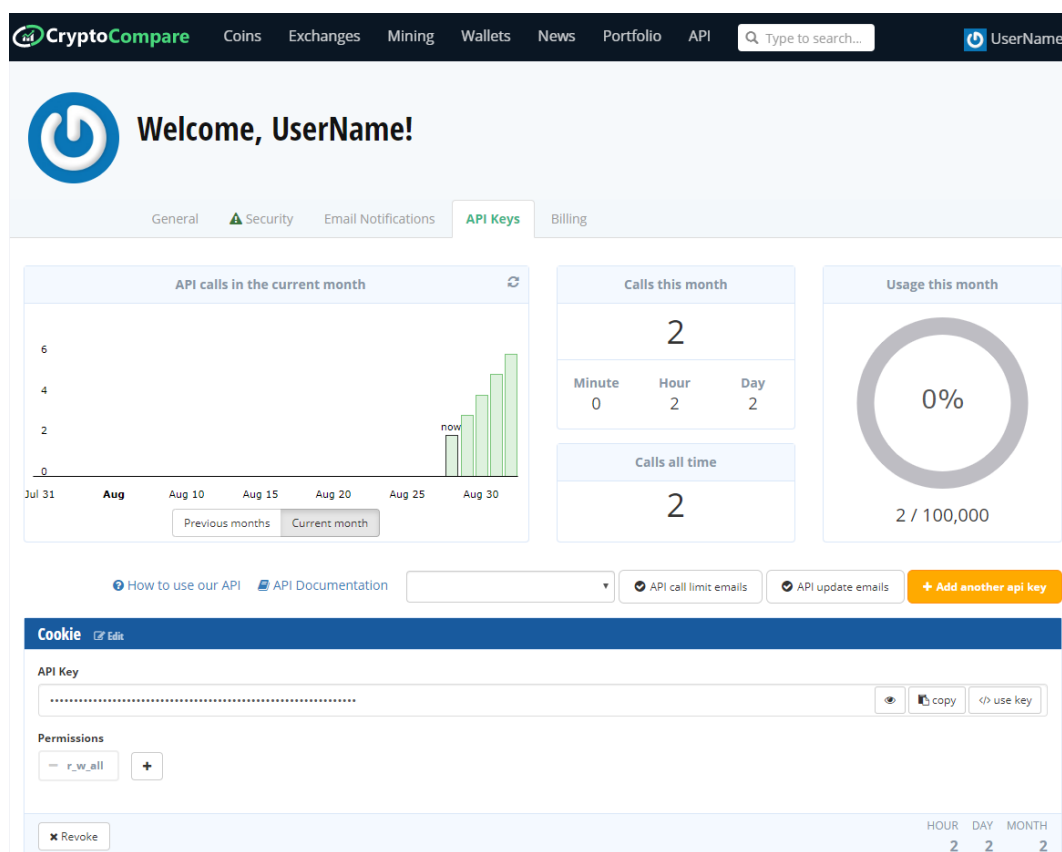


Рисунок 1.3 – Интерфейс «CryptoCompare»

Основные функции «CryptoCompare» включают:

- детальный анализ корреляций между криптовалютами;
- сбор исторических данных о ценах и объемах торгов;
- предоставление API для интеграции с другими системами;
- генерация отчетов о рыночных трендах;
- инструменты для сравнения производительности различных активов.

«CryptoCompare» является отличным выбором для инвесторов, которые хотят глубже понять взаимосвязи между активами и использовать эти данные для построения оптимального портфеля на основе модели Марковица.

Таким образом, рассмотренные программы представляют собой наиболее подходящие решения для управления криптовалютными портфелями. Однако ни одна из них не предоставляет полной реализации модели Марковица как основного механизма оптимизации портфеля. Это создает пространство для разработки нового программного средства, которое объединит преимущества существующих решений с научно обоснованными подходами к управлению рисками и доходностью.

2 МОДЕЛИРОВАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ И РАЗРАБОТКА ТРЕБОВАНИЙ К ПРОГРАММНОМУ СРЕДСТВУ

2.1 Анализ и формализация бизнес-процессов предметной области

Функциональное моделирование по стандарту IDEF0 позволяет анализировать и визуализировать бизнес-процессы, детализируя работу системы и взаимосвязь ее элементов. В контексте автоматизированной системы формирования инвестиционного портфеля криптовалют данный метод необходим для описания ключевых операций, их входных и выходных данных, а также механизмов реализации [5].

Основным процессом курсового проекта является формирование оптимального инвестиционного портфеля. Этот процесс включает последовательные этапы от сбора и анализа исходных данных до расчета ключевых показателей, выявления ограничений и формирования итоговой структуры портфеля, максимально соответствующей целям инвестора.

На рисунке 2.1 представлена контекстная диаграмма процесса «Сформировать оптимальный инвестиционный портфель». Этот процесс начинается с анализа данных о криптовалютах, учета рыночных ограничений и инвестиционных целей, таких как желаемая доходность и допустимый уровень риска. Управление процессом осуществляется через регуляторные нормы для криптоактивов в Республике Беларусь, принятую «Инвестиционную стратегию» и международный стандарт ISO 8000, обеспечивающий качество данных. Для выполнения задач задействуются «Бизнес-аналитик», аналитическое программное обеспечение, системы отчетности и визуализации, а также модель оптимизации. В результате создаются «Оптимальный инвестиционный портфель», графики риска и доходности, «Инвестиционные рекомендации» и «Отчет о составе портфеля», которые служат основой для принятия решений.

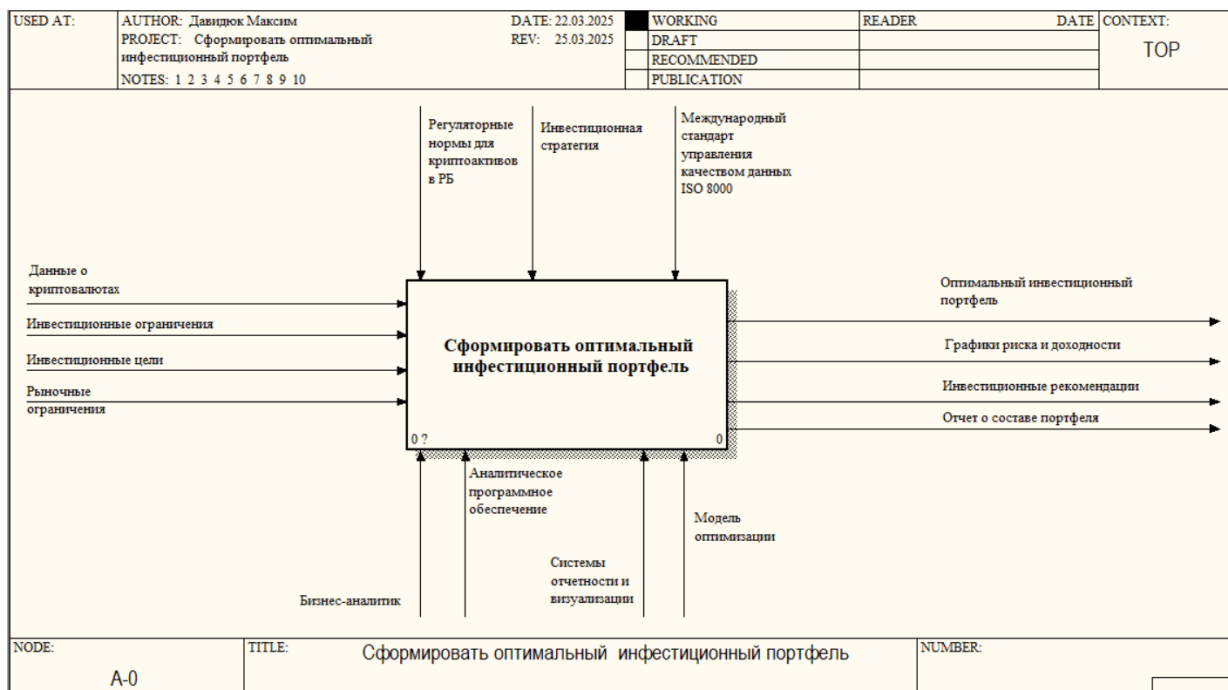


Рисунок 2.1 – Контекстная диаграмма процесса формирования оптимального инвестиционного портфеля

Далее рассмотрим декомпозицию процесса «Сформировать оптимальный инвестиционный портфель», представленную на рисунке 2.2. В данном случае работа делится на четыре этапа: «Сбор данных о криптовалютах», «Определить перечень возможных активов», «Провести анализ рисков и доходностей активов» и «Сформировать итоговый портфель». На каждом этапе используются соответствующие механизмы и управления. Например, при сборе данных ключевую роль играет аналитическое программное обеспечение, а при формировании портфеля — модель оптимизации. После завершения всех этапов формируется окончательный результат, включающий оптимальный портфель и сопутствующие документы.

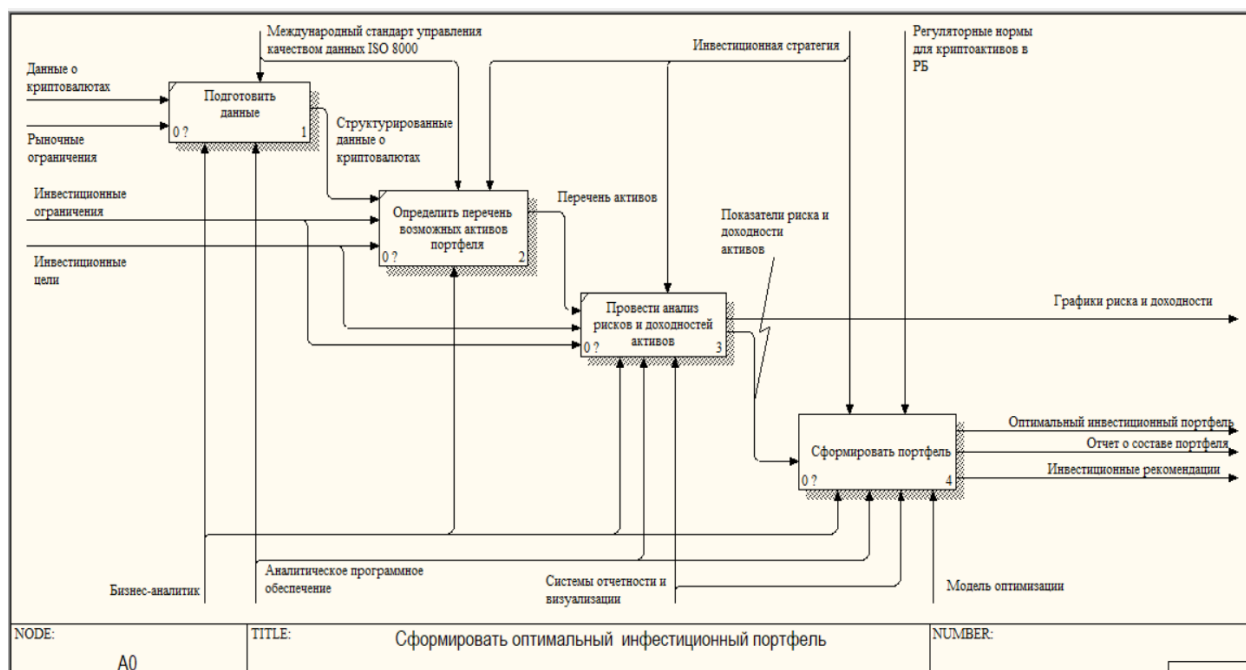


Рисунок 2.2 – Декомпозиция контекстной диаграммы

На рисунке 2.3 представлена декомпозиция процесса «Сформировать оптимальный портфель». Работа разбивается на пять этапов: «Определить целевые параметры портфеля», «Рассчитать границу эффективности», «Оптимизировать структуру портфеля», «Проверить устойчивость к изменениям рынка» и «Выполнить отчет о составе портфеля». На этом уровне особое внимание уделяется взаимодействию между этапами, где выход одного процесса становится входом для следующего. Например, результаты расчета границы эффективности используются для оптимизации распределения активов. По завершении всех этапов формируются итоговые документы, такие как «Инвестиционные рекомендации» и «Отчет о составе портфеля».

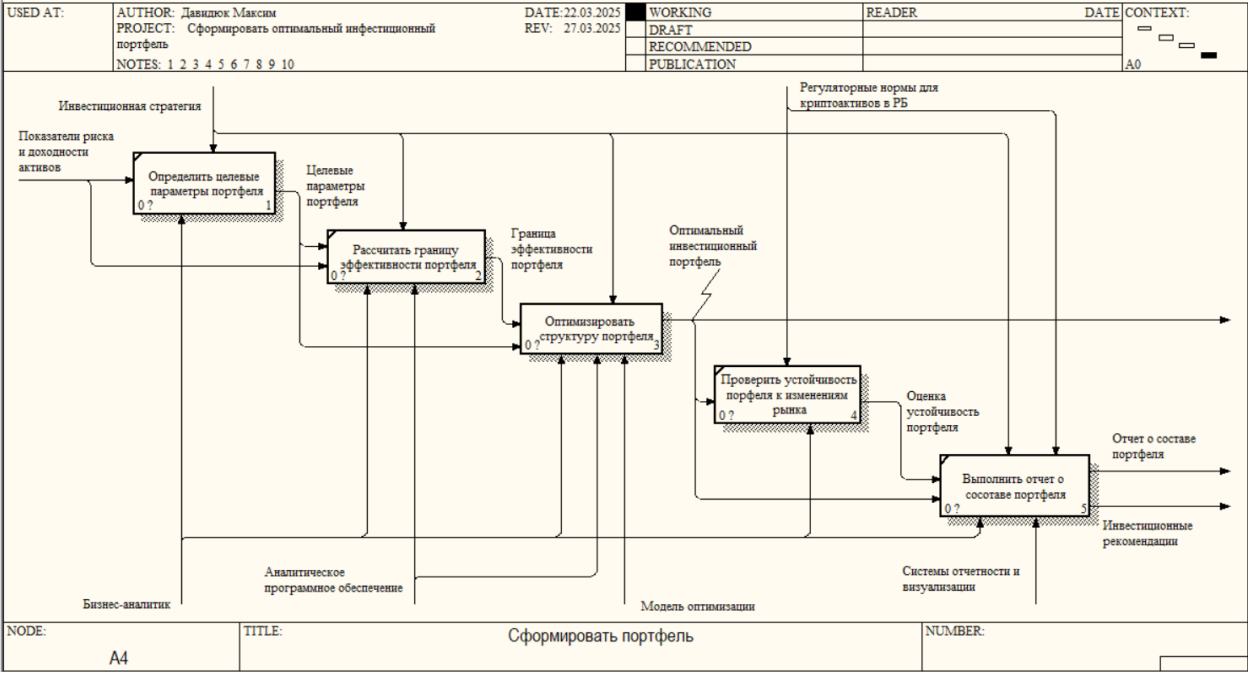


Рисунок 2.3 – Декомпозиции процесса «Сформировать портфель»

На рисунке 2.4 описана оптимизация структуры портфеля. Процесс разбивается на следующие блоки: «Сформулировать цель оптимизации», «Определить ограничения оптимизации» и «Оптимизировать распределение активов». Управление строится на базе инвестиционной стратегии, а механизмы включают аналитиков, модели и программное обеспечение.

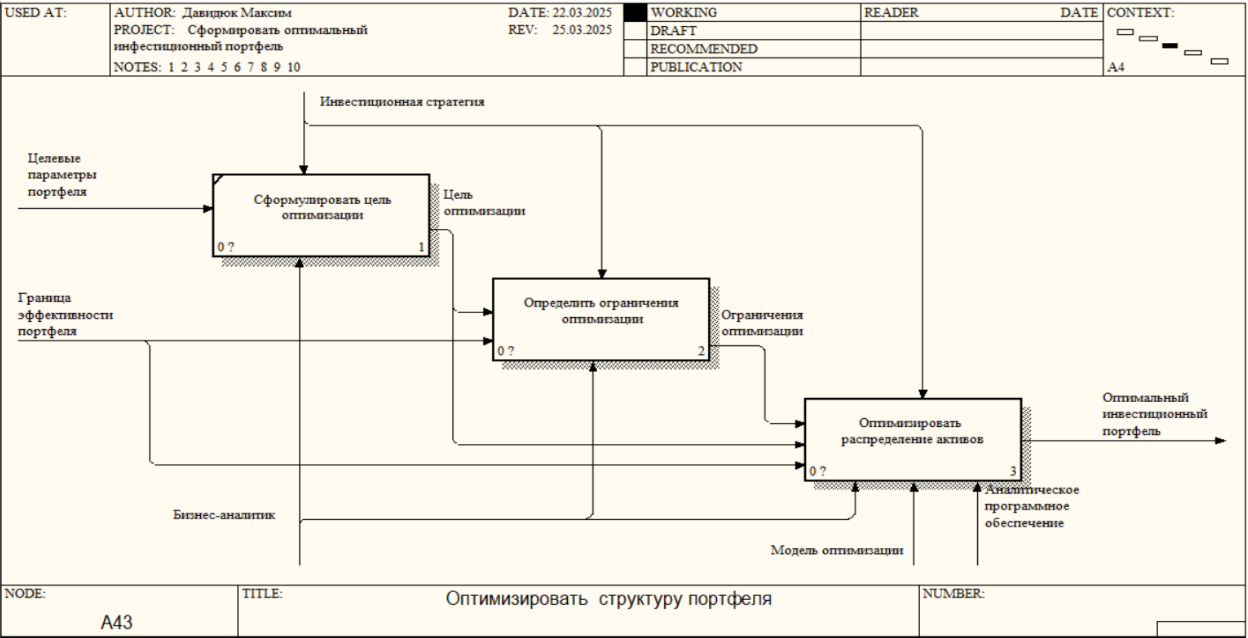


Рисунок 2.4 – Декомпозиция процесса «Оптимизировать структуру портфеля»

Наконец, на рисунке 2.5 показана декомпозиция процесса «Оптимизировать распределение активов». Работа включает три этапа: «Выполнить предварительный расчет», «Проверить соответствие заданным критериям» и «Сформировать окончательный портфель». Каждый этап контролируется «Инвестиционной стратегией» и подкрепляется профессиональной экспертизой бизнес-аналитика. После успешного завершения всех шагов формируется «Оптимальный инвестиционный портфель», готовый к использованию.

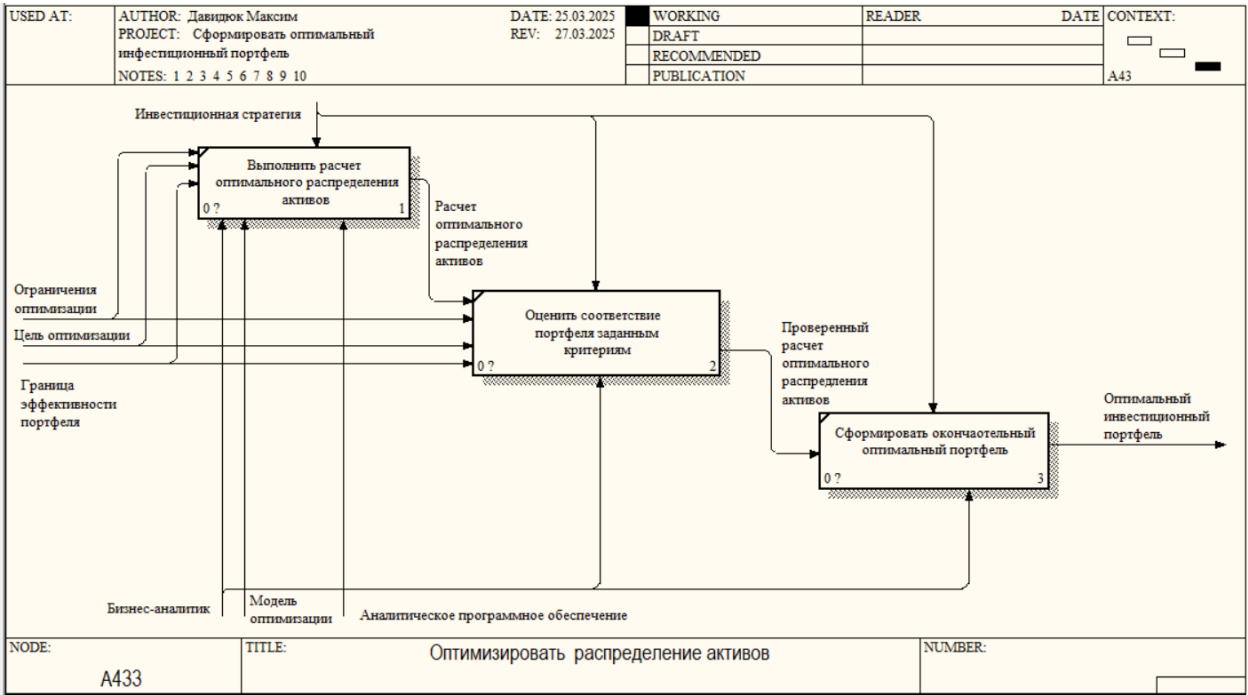


Рисунок 2.5 – Декомпозиции процесса «Оптимизировать распределение активов»

Таким образом, построение функциональной модели процесса «Сформировать оптимальный инвестиционный портфель» позволяет получить четкое представление о последовательности этапов, необходимых для достижения цели. Эта модель помогает не только определить ключевые шаги, но и обеспечить их логическую связь, что способствует повышению эффективности и точности принимаемых решений.

2.2 Анализ требований к программному средству и разработка их спецификации

Анализ требований к программному средству является ключевым этапом разработки системы управления портфелями криптовалют, так как он позволяет определить функциональные и нефункциональные потребности, обеспечивая соответствие системы ожиданиям пользователей. Для структурирования требований и наглядного представления взаимодействия пользователей с системой используется диаграмма вариантов использования (Use Case Diagram), которая является частью методологии UML (Unified Modeling Language). Диаграмма вариантов использования позволяет выделить основные роли пользователей, их цели и действия в системе, а также определить границы функциональности программного средства. Использование таких диаграмм способствует повышению прозрачности требований и упрощению коммуникации между разработчиками и заинтересованными сторонами, что делает их незаменимым инструментом в процессе проектирования [6].

Системой могут пользоваться два типа пользователей: пользователь, который представляет собой инвестора или аналитика, и администратор, отвечающий за техническое сопровождение и управление данными. Для каждой из этих ролей определён свой набор действий, которые они могут выполнять внутри системы, что отражено в диаграмме вариантов использования.

Варианты использования, доступные для пользователя, включают:

- регистрация в системе;
- авторизация в личном кабинете;
- загрузка ранее сохраненного портфеля;
- сохранение портфеля;
- выбор криптовалют для анализа;
- расчет ожидаемой доходности;
- расчет риска;
- построение эффективной границы;
- просмотр графиков цен криптовалют;
- загрузка ранее сохраненного портфеля;
- создание отчета.

Администратор наследует все варианты использования пользователя, а также определяет дополнительный набор функций:

- управление пользователями;
- управление списком криптовалют;
- назначение роли пользователю.

На рисунке 2.6 представлена диаграмма вариантов использования, которая демонстрирует все функциональные возможности системы и показывает взаимодействие между пользователями и приложением. С

помощью этой диаграммы можно понять, какие именно действия могут выполнять участники системы и как они связаны с её компонентами.

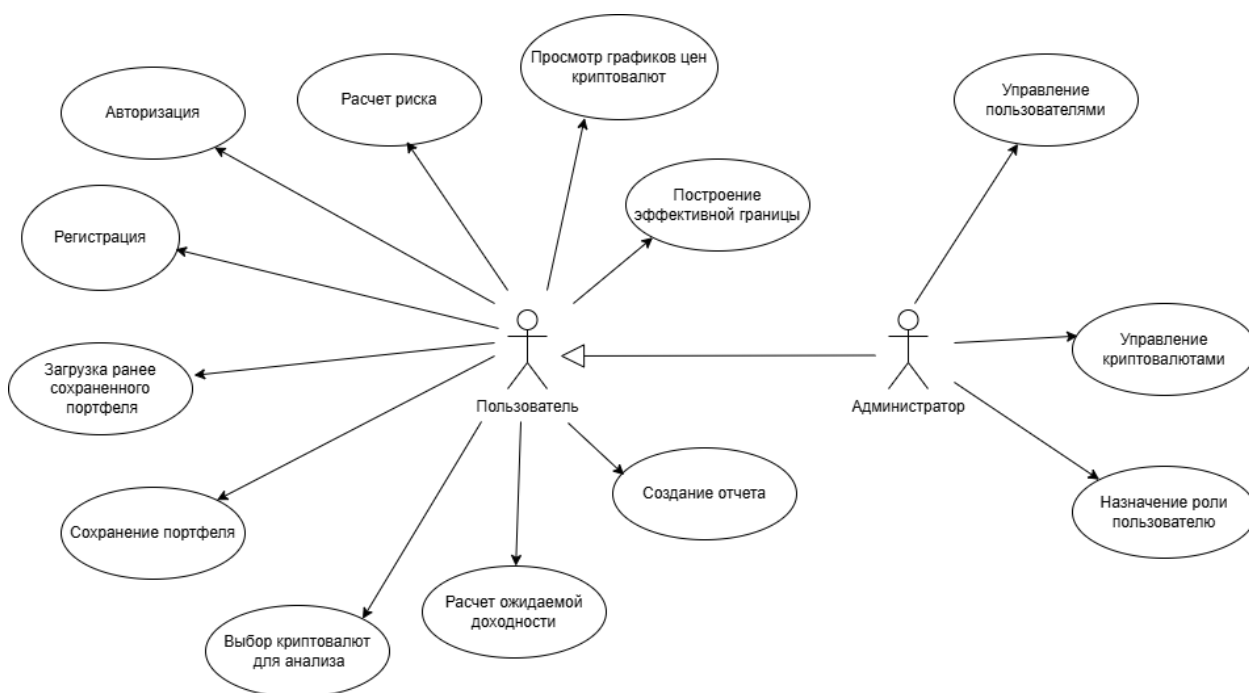


Рисунок 2.6 – Диаграмма вариантов использования

Диаграмма позволяет заранее структурировать функциональные требования, выявить потенциальные проблемы взаимодействия и правильно распределить ответственность между различными ролями. Это особенно важно при разработке сложных финансовых приложений, где каждая операция требует высокой точности и чёткой организации.

Таким образом, применение диаграммы вариантов использования способствует более грамотному проектированию программного средства, снижает вероятность возникновения ошибок на ранних этапах разработки и помогает создать понятное и логически выстроенное приложение. Особенно это актуально в случае создания системы управления инвестиционным портфелем на рынке криптовалют, где требуется учитывать множество факторов и обеспечивать максимальную прозрачность действий пользователя.

2.3 Образ предлагаемого решения

Современные инвестиционные процессы в сфере криптовалют требуют высокой скорости принятия решений, гибкости анализа и автоматизации рутинных операций. Однако в условиях отсутствия единой программной платформы участники инвестиционной деятельности сталкиваются с рядом трудностей — от ограниченного доступа к данным до необходимости ручной обработки информации. Это существенно снижает эффективность управления инвестиционными портфелями и увеличивает нагрузку на административный персонал.

В результате проведенного анализа были выявлены следующие ключевые бизнес-проблемы:

- отсутствие централизованного программного инструмента, обеспечивающего эффективное взаимодействие между инвесторами и аналитиками, что затрудняет координацию процессов управления портфелями;

- ограниченная возможность инвесторов и аналитиков самостоятельно получать доступ к данным о портфелях, аналитических показателях и истории цен криптовалют, что требует постоянного обращения к техническому специалисту;

- отсутствие автоматизированного механизма для подачи запросов на обновление данных или корректировку портфелей, что замедляет принятие инвестиционных решений;

- недостаток инструментов для анализа и фильтрации данных о криптовалютах и портфелях, ограничивая возможности инвесторов и аналитиков в оценке рисков и доходности;

- необходимость ручной обработки данных и подготовки аналитических отчетов техническим специалистом, что увеличивает вероятность ошибок и задержек;

- отсутствие автоматизированной системы распределения ролей и прав доступа, что создает риски несанкционированного доступа к конфиденциальной информации;

- отсутствие средств для автоматического построения графиков и диаграмм, затрудняющих визуальную оценку эффективности портфелей.

Для решения этих проблем предлагается разработка программного средства, предназначенного для автоматизации формирования и управления инвестиционными портфелями криптовалют на основе модели Марковица. Система будет ориентирована на две категории пользователей: инвесторов и аналитиков, выполняющих операции с портфелями, и технических специалистов, обеспечивающих техническую поддержку. В зависимости от роли предоставляется соответствующий доступ к функционалу, что обеспечивает как гибкость, так и безопасность работы с данными. Внедрение данного программного средства обещает повысить эффективность управления инвестициями, минимизировать время на ручные операции и снизить нагрузку

на технических специалистов, подготавливая основу для детального анализа требований в последующих разделах.

Программное средство формирования инвестиционного портфеля криптовалют на основе модели Марковица предназначено для автоматизации процессов создания, анализа и оптимизации портфелей, а также обеспечения безопасного управления доступом к данным. Разработка реализуется в виде клиент-серверного приложения на языке Java, с использованием оконного интерфейса (JavaFX) для клиентской части и многопоточной архитектуры сервера. Система включает серверное приложение, клиентский интерфейс и базу данных под управлением СУБД PostgreSQL, обеспечивая надежное хранение и обработку информации. Взаимодействие между пользователями разных ролей (инвесторы/аналитики, технические специалисты) будет организовано через централизованную систему авторизации и ролевого доступа, что станет предметом более детального рассмотрения в спецификации требований.

Программное средство формирования инвестиционного портфеля криптовалют с модулем аналитики создается с учетом следующих целей:

- автоматизация процессов формирования и оптимизации портфелей криптовалют с использованием модели Марковица для повышения точности расчетов;
- обеспечение удобного доступа инвесторов и аналитиков к данным о портфелях, аналитике и истории цен, что станет основой для их взаимодействия с системой;
- упрощение взаимодействия между инвесторами, аналитиками и техническими специалистами, минимизируя зависимость от ручного управления;
- предоставление инструментов для анализа рисков и доходности портфелей, которые будут интегрированы в функционал системы;
- обеспечение гибкого управления доступом к информации в зависимости от роли пользователя, что будет детализировано в модели прав доступа.

Для достижения этих целей необходимо разработать программное средство, решающее следующие задачи:

- автоматизация расчета рисков и доходности портфелей с учетом корреляции между криптовалютами, что будет основано на математическом моделировании;
- предоставление пользователям доступа к данным портфелей, аналитическим показателям и истории цен через интуитивный интерфейс;
- упрощение подачи запросов на обновление данных или корректировку портфелей, что улучшит оперативность работы;
- фильтрация и сортировка данных о криптовалютах и портфелях по ключевым параметрам, что облегчит анализ;

- формирование аналитических отчетов и визуализаций (графики, диаграммы) для оценки эффективности портфелей, что будет реализовано в аналитическом модуле.

Анализ текущего состояния бизнес-процессов управления портфелями криптовалют выявил следующие недостатки, которые требуют устранения для создания эффективной системы:

- большое количество операций выполняется вручную, что увеличивает затраты времени и ресурсов, подчеркивая необходимость автоматизации;

- отсутствие единой централизованной системы, обеспечивающей взаимодействие между инвесторами, аналитиками и техническими специалистами, что затрудняет координацию;

- невозможность пользователей самостоятельно получать доступ к данным о портфелях и аналитике, что зависит от вмешательства технических специалистов;

- отсутствие автоматизированной системы подачи и обработки запросов на обновление данных, что замедляет процесс управления;

- нехватка инструментов анализа и визуализации данных о рисках и доходности, ограничивая возможности пользователей;

- сложность управления доступом к данным в зависимости от роли пользователя, что создает риски для безопасности.

Таким образом, выявленные бизнес-проблемы и анализ текущего состояния процессов подтвердили необходимость разработки программного средства для автоматизации управления криптовалютными портфелями. Разработка системы, ориентированной на роли пользователей и нацеленной на реализацию аналитических и управленческих функций, позволит устранить существующие недостатки, повысить производительность и упростить взаимодействие всех участников инвестиционного процесса.

3 ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ПРОГРАММНОГО СРЕДСТВА

3.1 Архитектурные решения и технологии реализации программного средства

Выбор архитектурных решений и технологий реализации программного средства является определяющим этапом разработки, поскольку от него напрямую зависят производительность, масштабируемость, безопасность и удобство использования системы. Для системы управления портфелями криптовалют на основе модели Марковица, которая оперирует сложными математическими расчетами, большими объемами финансовых данных и требует интуитивного интерфейса для инвесторов и аналитиков, необходимо обеспечить надежную и эффективную техническую основу. Правильно подобранные технологии позволяют автоматизировать процессы формирования портфелей, минимизировать риски ошибок и обеспечить оперативный доступ к данным, что особенно важно в условиях высокой волатильности криптовалютного рынка. В данном разделе описываются архитектурные решения, используемые технологии, а также обоснование выбора инструментов и паттернов проектирования, которые обеспечивают успешную реализацию проекта.

Для реализации программного средства формирования инвестиционного портфеля криптовалют планируется использование набора современных технологий, обеспечивающих высокую производительность, гибкость архитектуры и удобство пользовательского взаимодействия. В качестве инструмента для разработки графического интерфейса клиентской части приложения будет применяться JavaFX, обладающий широкими возможностями по созданию современных и интерактивных пользовательских интерфейсов. Это обеспечит комфортную работу инвесторов и аналитиков с системой, в том числе позволит реализовать визуализацию аналитических данных, таких как эффективная граница портфеля или динамика изменения цен криптовалют.

Для организации взаимодействия между серверной частью и системой управления базами данных будет использован Hibernate — объектно-реляционный маппер (ORM), упрощающий работу с данными за счёт преобразования объектов Java в соответствующие записи таблиц базы данных. Использование Hibernate позволит централизованно управлять операциями чтения, записи, обновления и удаления данных, а также повысит безопасность приложения за счёт использования параметризованных запросов, минимизирующих риск SQL-инъекций.

В качестве системы управления базами данных выбрана PostgreSQL, обладающая высокой надёжностью, масштабируемостью и поддержкой сложных SQL-запросов, что делает её оптимальным решением для хранения и

обработки данных, связанных с криптовалютными портфелями и пользовательской активностью.

Для построения графиков и диаграмм найдут применение библиотеки JFreeChart, которая обеспечит создание визуализаций, таких как эффективная граница портфеля и динамика цен криптовалют, интегрируясь с JavaFX и предоставляя гибкие инструменты для отображения линий, столбцов и точечных диаграмм, а также встроенные компоненты JavaFX Charts, включая LineChart и ScatterChart, которые будут использованы для отображения аналитических данных, например, графиков цен или зависимости риска и доходности.

Для получения актуальных и исторических данных о ценах криптовалют в режиме реального времени в системе будет использоваться Binance API, предоставляющий доступ к рыночной информации посредством REST- и WebSocket-интерфейсов. С помощью данного API будут получаться данные о ценах, объёмах торгов и других ключевых рыночных метриках по ведущим криптовалютам, таким как *Bitcoin*, *Ethereum* и другим. Полученная информация будет сохраняться в таблице базы данных, что обеспечит необходимую базу для расчёта параметров инвестиционного портфеля в соответствии с моделью Марковица.

Дополнительно в системе планируется использование API CoinGecko, который позволит получить расширенные исторические данные, не всегда доступные через Binance, включая редкие криптовалюты и долгосрочные ценовые ряды. Это обеспечит более полную и достоверную информацию для анализа рыночных трендов и расчета коэффициентов корреляции между активами, что критически важно при формировании сбалансированного портфеля. Интеграция с обоими API обеспечит устойчивость к отказам и повысит точность аналитики за счёт перекрёстной валидации данных из разных источников.

Для выполнения математических расчётов, необходимых при построении оптимального портфеля по модели Марковица, будет использована библиотека Apache Commons Math, предоставляющая инструменты для линейной алгебры, численной оптимизации и статистического анализа. Это позволит реализовать ключевые алгоритмы расчета ожидаемой доходности, дисперсии, ковариации и других параметров, определяющих структуру эффективного портфеля.

Обработка и преобразование данных, полученных в формате JSON от сторонних API, будет осуществляться с использованием библиотеки Gson, которая упростит сериализацию и десериализацию данных, обеспечивая преобразование входных потоков в объекты Java. Это существенно ускорит процесс интеграции с внешними сервисами и обеспечит удобную работу с полученной информацией внутри системы.

Для ведения журналов событий и отладки системы будет использоваться связка SLF4J и Logback, обеспечивающая гибкое и централизованное логирование. Это позволит отслеживать действия пользователей (например,

авторизацию, создание портфелей, выполнение расчетов), а также регистрировать ошибки и исключения, что упростит сопровождение и диагностику приложения.

Обработка временных данных, включая хранение и анализ исторических цен, будет реализована с применением Java Time API, предоставляющего мощные средства для работы с датами, временами и временными интервалами. Это особенно важно при построении временных рядов, расчёте доходности за различные периоды и синхронизации данных из внешних источников.

Система будет реализована в виде клиент-серверного приложения, архитектура которого предполагает чёткое разделение обязанностей между клиентской и серверной частями. Клиент отвечает за отображение интерфейса и взаимодействие с пользователем, а сервер — за выполнение бизнес-логики, обработку запросов, хранение и анализ данных. Такой подход обеспечивает лучшую масштабируемость, модульность и возможность интеграции с внешними источниками данных и сервисами.

Взаимодействие между клиентом и сервером будет осуществляться по сетевому протоколу TCP/IP, что гарантирует надёжную и устойчивую передачу данных в процессе работы приложения. Использование TCP (Transmission Control Protocol) позволит обеспечить целостность и последовательность передаваемых данных, что особенно важно при передаче финансовой информации и параметров инвестиционного анализа. Кроме того, данная архитектура облегчает развертывание системы в распределённых средах, включая облачные решения, а также упрощает обновление и сопровождение отдельных компонентов без необходимости полной переустановки системы у конечных пользователей.

Для обеспечения модульности, расширяемости и поддержки в процессе развития системы в проекте предполагается использование проверенных паттернов проектирования, адаптированных под задачи анализа и управления криптовалютными портфелями.

Таким образом, выбор архитектурных решений и технологий разработки программного средства формирования инвестиционного портфеля криптовалют обоснован спецификой задачи и требованиями к производительности, надёжности и масштабируемости системы. Использование JavaFX для реализации графического интерфейса обеспечит удобную и наглядную работу пользователей с аналитическими инструментами. Внедрение Hibernate и PostgreSQL позволит эффективно управлять данными и обеспечит безопасность взаимодействия с базой данных. Интеграция с внешними API, такими как Binance и CoinGecko, обеспечит систему достоверной и актуальной рыночной информацией. Применение специализированных библиотек — Apache Commons Math, Gson, SLF4J и Java Time API — обеспечит точность математических расчётов, удобство обработки данных и возможность отладки на всех этапах жизненного цикла программы. В совокупности данные решения формируют прочную

техническую основу, способную обеспечить корректную реализацию модели Марковица и адаптироваться к изменяющимся условиям криптовалютного рынка.

3.2 Проектирование и разработка пользовательского интерфейса

Пользовательский интерфейс представляет собой ключевой компонент программного обеспечения, определяющий удобство и эффективность взаимодействия пользователя с системой, что напрямую влияет на её практическую ценность. Он включает совокупность визуальных и функциональных элементов, обеспечивающих доступ к функциональности приложения и отображающих информационную модель предметной области. В условиях высокой сложности вычислений и необходимости визуализации аналитических данных, характерных для системы управления портфелями криптовалют на основе модели Марковица, особенно важно будет спроектировать интерфейс таким образом, чтобы он был не только функциональным, но и интуитивно понятным. Пользователи, включая инвесторов и аналитиков, должны будут легко формировать портфели, отслеживать ключевые метрики (доходность, риск, структура активов) и визуализировать результаты моделирования и анализа. Качество интерфейса станет играть решающую роль в восприятии системы конечными пользователями и обеспечит её практическую применимость в инвестиционной сфере [7].

Разработка интерфейса для программного средства формирования инвестиционного портфеля криптовалют будет вестись с акцентом на ясную навигацию, адаптивность и доступность всех функций как для опытных инвесторов, так и для пользователей без специальной подготовки. Интерфейс станет отражать сложную информационную модель системы, интегрируя данные из базы данных, такие как портфели, аналитические показатели и история цен, и предоставит пользователям инструменты для их эффективного анализа.

Для реализации визуальной части будет использован фреймворк JavaFX, который обеспечит широкий набор компонентов для построения современных графических интерфейсов и поддержит гибкое взаимодействие с данными. Визуальные элементы, такие как таблицы с данными криптовалют, графики зависимости риска и доходности, диаграммы распределения активов и кнопки для управления портфелем, будут распределены по вкладкам с логической структурой. Это создаст целостную и понятную пользовательскую среду, где каждый элемент интерфейса будет соответствовать конкретной задаче пользователя, упрощая доступ к функциям, таким как создание портфеля или просмотр аналитики.

На рисунке 3.1 представлен главный экран программы с таблицей криптовалют и базовыми функциями управления портфелем.

На рисунке 3.2 показано окно создания нового инвестиционного портфеля с возможностью выбора активов и задания ограничений.

На рисунке 3.3 приведена визуализация эффективной границы и области допустимых решений.

На рисунке 3.4 демонстрируется интерфейс отображения исторической динамики цены выбранного актива.

Разработка пользовательского интерфейса для программного средства станет важным шагом в обеспечении его практической ценности, предоставляя интуитивно понятный и функциональный доступ к данным и аналитике. Использование JavaFX в сочетании с логической структурой вкладок и визуальными элементами обеспечит удобство работы для всех категорий пользователей, подготавливая основу для эффективного управления портфелями криптовалют на основе модели Марковица.

3.3 Разработка модели данных

Информационная модель является ключевым элементом проектирования базы данных, обеспечивая структурированное представление данных и их взаимосвязей в рамках системы управления портфелями криптовалют на основе модели Марковица. Она служит инструментом для определения основных компонентов системы, их атрибутов и отношений, что позволяет оптимизировать процессы хранения, обработки и анализа информации. Информационная модель может быть представлена в виде ER-диаграммы (Entity-Relationship Diagram), которая наглядно отображает сущности, их атрибуты и связи между ними, упрощая разработку и поддержку базы данных. Актуальность данной модели обусловлена её применением в инвестиционной сфере, где точность и эффективность управления портфелями криптовалют имеют решающее значение для инвесторов и аналитиков [8].

ER-диаграмма представляет собой графическую модель, которая описывает структуру данных системы и включает в себя 3 основных компонента:

- сущности – объекты реального мира или концепции, которые имеют независимое существование и могут быть однозначно идентифицированы. Каждая сущность представлена в виде прямоугольника на диаграмме;
- атрибуты – характеристики или свойства, описывающие сущности;
- связи – ассоциации между сущностями, показывающие, как они взаимодействуют друг с другом.

Эти компоненты объединены в единую структуру, отражающую логику работы системы и обеспечивающую её целостность. Визуализация информационной модели представлена на рисунке 1.

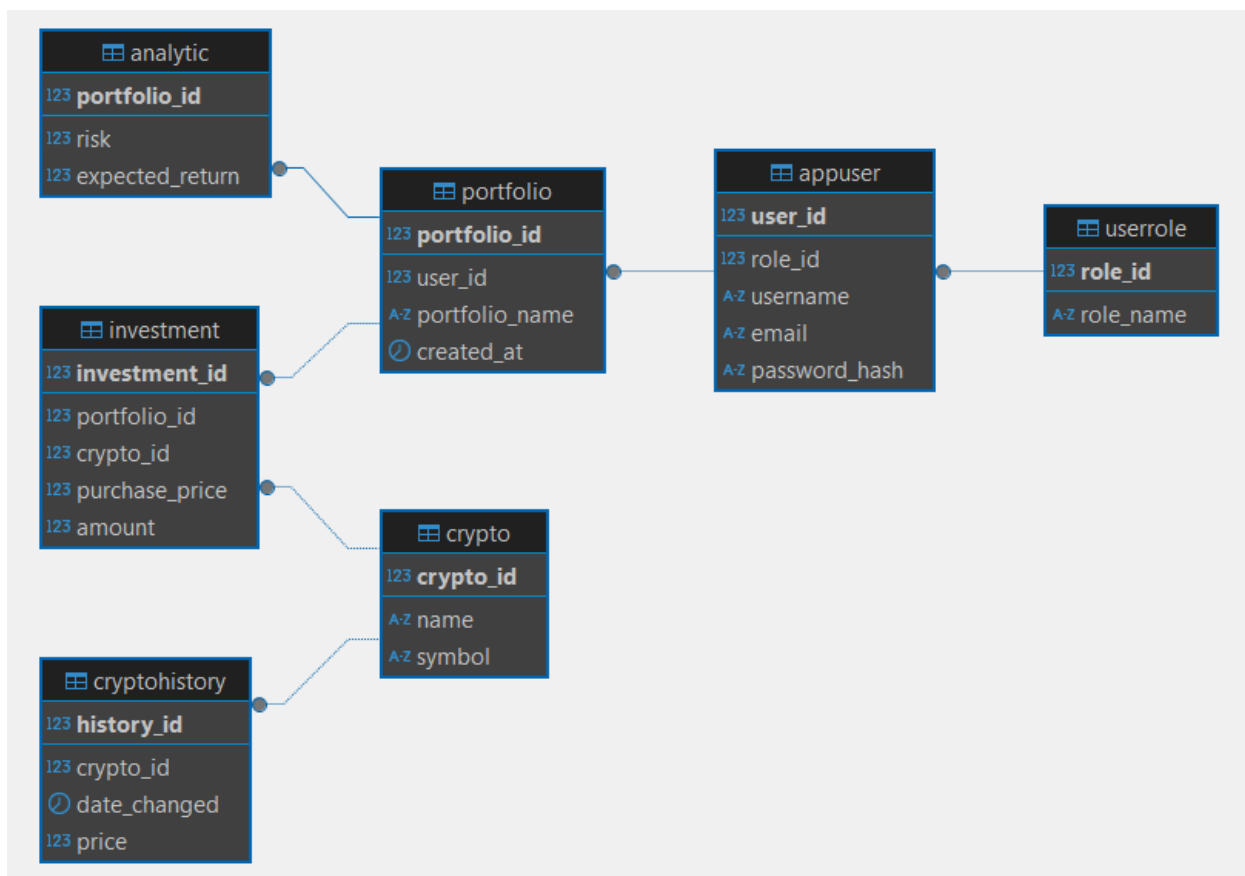


Рисунок 1 – Модель базы данных

Представленная ER-диаграмма соответствует третьей нормальной форме (3NF), что обеспечивает минимизацию избыточности данных и упрощение их обновления. Рассмотрим соответствие модели каждой из нормальных форм:

1 Первая нормальная форма (1NF): все атрибуты атомарны, то есть не содержат повторяющихся групп или множественных значений. Например, поле username в таблице appuser хранит одно значение для каждого пользователя, а инвестиции в портфеле вынесены в отдельную таблицу investment с уникальными записями.

2 Вторая нормальная форма (2NF): все неключевые атрибуты полностью зависят от первичного ключа. Например, в таблице investment атрибуты purchase_price и amount зависят только от investment_id, а связи реализованы через внешние ключи, исключая частичные зависимости.

3 Третья нормальная форма (3NF): отсутствуют транзитивные зависимости. Например, в таблице appuser атрибуты username и email зависят только от user_id, а не от других атрибутов, таких как role_id.

Описание каждой сущности, включая их назначение и роль в системе управления портфелями криптовалют, представлено в таблице 1. Эта таблица предоставляет детализированную информацию о каждом компоненте базы данных, подчёркивая их функциональное предназначение и значимость для эффективного управления инвестициями в криптовалюты.

Таблица 1 – Сущности модели проектируемой базы данных

| Имя | Определение | Определение записи пользователей или программного средства |
|------------------------------------|--|---|
| Role (Роль) | Сущность, представляющая роли пользователей в системе для разграничения доступа. | Определяет права доступа пользователей |
| User (Пользователь) | Сущность, содержащая данные о зарегистрированных пользователях системы. | Хранит учетные данные и роли пользователей для управления доступом. |
| Portfolio (Портфель) | Сущность, представляющая инвестиционный портфель пользователя. | Описывает портфели, созданные пользователями, для управления активами. |
| Analytic (Аналитика) | Сущность, содержащая аналитические данные портфеля (риск и доходность). | Служит для хранения результатов анализа портфелей по модели Марковица. |
| Crypto (Криптовалюта) | Сущность, представляющая доступные для инвестиций криптовалюты. | Обеспечивает справочник криптовалют для выбора активов в портфелях. |
| Investment (Инвестиция) | Сущность, описывающая конкретные инвестиции в криптовалюты в портфеле. | Фиксирует данные о покупках криптовалют пользователями в рамках портфеля. |
| CryptoHistory(История криптовалют) | Сущность, хранящая историю изменений цен криптовалют. | Используется для анализа динамики цен и оптимизации портфелей. |

Таблица 2 содержит описание атрибутов каждой сущности. Для каждого атрибута приведено краткое описание его назначения и роли в рамках соответствующей сущности, что упрощает интерпретацию данных. Эти описания необходимы для точного проектирования базы данных и её дальнейшего использования в системе управления портфелями.

Таблица 2 – Описание сущностей базы данных

| Таблица | Назначение атрибута | Наименование поля | Тип данных | Ключ |
|-----------------------|----------------------------|-------------------|----------------|--|
| Role (Роль) | Идентификатор роли | role_id | serial | PK |
| | Название роли | role_name | nvarchar(50) | |
| User (Пользователь) | Идентификатор пользователя | user_id | serial | PK |
| | Идентификатор роли | role_id | int | FK, связывает с таблицей Role, кардинальность связи – N:1 |
| | Имя пользователя | username | nvarchar(50) | |
| | Электронная почта | email | nvarchar(50) | |
| | Хешированный пароль | password_hash | nvarchar(60) | |
| Analytic (Аналитика) | Идентификатор портфеля | portfolio_id | int | PK, FK, связывает с таблицей Portfolio, кардинальность связи – 1:1 |
| | Риск портфеля | risk | numeric (10,6) | |
| | Ожидаемая доходность | expected_return | numeric (10,6) | |
| Portfolio (Портфель) | Идентификатор портфеля | portfolio_id | int | PK |
| | Код пользователя | user_id | int | FK, связь с таблицей User, кардинальность связи – N:1 |
| | Имя портфеля | portfolio_name | nvarchar(50) | |
| | Дата создания | created_at | timestamp | |
| Crypto (Криптовалюта) | Идентификатор криптовалюты | crypto_id | int | PK |
| | Название криптовалюты | crypto_name | nvarchar(50) | |
| | Символ | symbol | nvarchar(10) | |
| CryptoHistory () | Идентификатор записи | crypto_hystory_id | serial | PK |
| | Идентификатор криптовалюты | crypto_id | int | FK, связь с таблицей Crypto, кардинальность связи – N:1 |
| | Дата изменения | date_changed | date | |
| | Цена | price | numeric (20,8) | |

Продолжение таблицы 2

| Таблица | Назначение атрибута | Наименование поля | Тип данных | Ключ |
|----------------------------|----------------------------|-------------------|------------|--|
| Investment (Инвестиция) | Идентификатор инвестиции | investment_id | int | PK |
| | Идентификатор портфеля | portfolio_id | int | FK, связь с таблицей Portfolio, кардинальность связи – N:1 |
| | Идентификатор криптовалюты | crypto_id | int | FK, связь с таблицей Crypto, кардинальность связи – N:1 |
| | Цена покупки | purchase_price | | |
| | Количество | amount | | |

Таким образом разработанная информационная модель базы данных для системы управления портфелями криптовалют, основанной на модели Марковица, представляет собой тщательно продуманную структуру, воплощённую в ER-диаграмме с чётко определёнными компонентами и связями, опирающимися на авторитетное руководство. Анализ продемонстрировал полное соответствие модели третьей нормальной форме (3NF), что гарантирует отсутствие избыточности данных и упрощает их модификацию, создавая прочную основу для дальнейшего развития системы. Описания компонентов и атрибутов, представленные в таблицах 1 и 2, служат не только ориентиром для реализации программного обеспечения, но и мощным инструментом для обеспечения целостности и эффективности работы всей системы, обещающей стать надёжным решением для инвесторов и аналитиков.

3.4 Описание статических и динамических аспектов поведения программных объектов

3.5 Разработка и описание алгоритмов, реализующих бизнес-логику программного средства

3.6 Механизмы обеспечения информационной безопасности

Информационная безопасность является неотъемлемой частью функционирования системы управления портфелями криптовалют. Для защиты данных пользователей и обеспечения устойчивости к внешним и внутренним угрозам в системе реализован ряд базовых, но эффективных мер, направленных на предотвращение несанкционированного доступа, утечек информации и нарушения целостности данных:

1 Пользователи системы имеют разграниченные полномочия в зависимости от своей роли. Такой подход позволяет ограничить круг доступных функций и данных для каждого типа пользователей, что минимизирует риски злоупотребления полномочиями и доступа к конфиденциальной информации;

2 При регистрации и работе с системой обеспечивается уникальность пользовательских идентификаторов и корректность вводимых данных. Это способствует предотвращению дублирования аккаунтов и повышает надёжность системы аутентификации;

3 Для повышения стойкости учетных записей к взлому применяются базовые требования к сложности паролей. Это снижает вероятность успешного подбора пароля и защищает от несанкционированного доступа;

4 Доступ к внутренним данным системы осуществляется только через серверную сторону программы, что исключает возможность прямого взаимодействия с критически важными компонентами, такими как база данных. Это помогает предотвратить возможные атаки и вмешательства со стороны пользователей;

5 Пароли пользователей обрабатываются с помощью алгоритмов хэширования, что позволяет сохранить их в закрытом виде, даже при несанкционированном доступе к базе данных.

В совокупности эти меры обеспечивают надёжную защиту информации в системе, соответствуя современным стандартам безопасности и требованиям к разработке программных продуктов, обрабатывающих конфиденциальные данные.

4 ТЕСТИРОВАНИЕ И ПРОВЕРКА РАБОТОСПОСОБНОСТИ ПРОГРАММНОГО СРЕДСТВА

5 РУКОВОДСТВО ПО УСТАНОВКЕ (РАЗВЕРТЫВАНИЮ) И ИСПОЛЬЗОВАНИЮ ПРОГРАММНОГО СРЕДСТВА

5.1 Руководство по установке (развертыванию) программного средства

5.2 Руководство пользователя.

ЗАКЛЮЧЕНИЕ

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

[1] Модель Марковица и финансовый портфель – Empirix [Электронный ресурс]. – Режим доступа: <https://empirix.ru/optimizacziya-portfelya-po-markoviczu/>.

[2] Portfolio Performance. (2023). Open Source Portfolio Management Tool [Электронный ресурс]. – Режим доступа: <https://www.portfolio-performance.info>.

[3] Обзор CoinStats: Сервис Агрегации Портфелей [Электронный ресурс]. – Режим доступа: <https://ru.bitdegree.org/crypto/obzor-coinstats>.

4 Обзор Cryptocompare — инструмент для контроля (учёта) своего инвестиционного портфеля [Электронный ресурс]. – Режим доступа: <https://happycoin.club/41066-2/>.

[5] Автоматизация процессов — средства и этапы автоматизации [Электронный ресурс]. – Режим доступа: <https://proresult.by/sredstva-i-etapy-avtomatizatsii/>.

[6] Fowler, M. UML Distilled: A Brief Guide to the Standard Object Modeling Language. 3rd ed. Addison-Wesley, 2003.

[7] Galitz, W. O. The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques. 3rd ed. Wiley, 2007.

[8] Coronel, C., Morris, S. Database Systems: Design, Implementation, and Management. 13th ed. Cengage Learning, 2019.

ПРИЛОЖЕНИЕ А
(обязательное)

Отчет о проверке на заимствование в системе «Антиплагиат»

Рисунок А.1 – Результат проверки на заимствования

ПРИЛОЖЕНИЕ Б
(обязательное)

Листинг кода алгоритмов, реализующих основную бизнес-логику