МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра математического и компьютерного моделирования

Разработка системы электронного голосования на базе технологии Blockchain

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студентки 4 курса 441 группы

направление 09.03.03 — Прикладная информатика

механико-математического факультета

Анпилоговой Валерии Сергеевны

Научный руководитель
доцент, к.ф.-м.н., доцент _____ Е.Ю. Крылова

Зав. кафедрой
зав. каф., д.ф.-м.н., доцент _____ Ю.А. Блинков

Саратов 2025

Введение. В современном обществе участие граждан в политических и общественных процессах является фундаментальным принципом демократического устройства. Одним из ключевых механизмов реализации этого участия выступает голосование — процесс, позволяющий выразить мнение и выбор по важным вопросам. Традиционные методы голосования, основанные на бумажных бюллетенях, хотя и доказали свою эффективность, сталкиваются с рядом ограничений, таких как длительное время подсчёта голосов, возможность ошибок и необходимость значительных ресурсов.

Данная бакалаврская работа посвящена разработке и исследованию системы электронного голосования, направленной на обеспечение надёжности, удобства и прозрачности избирательного процесса.

Актуальность исследования на тему обусловлена необходимостью повышения безопасности, прозрачности и доверия к процессам голосования в современном обществе. Традиционные методы голосования часто подвержены рискам фальсификаций, манипуляций и технических сбоев, что снижает их надежность и легитимность.

В условиях стремительного развития цифровых технологий и растущих требований к честности и открытости выборов, разработка эффективной системы электронного голосования на базе Blockchain становится актуальной задачей, способствующей укреплению демократических институтов и повышению доверия граждан к избирательным процедурам.

Объект исследования: технология Blockchain.

Предмет исследования: система электронного голосования.

Цель работы: разработать систему электронного голосования на базе технологии Blockchain.

Для достижения цели данной работы необходимо решить следующие задачи:

1. Изучить и проанализировать проблемы, присущие традиционным и современным системам электронного голосования, а также выявить их недостатки и риски.
2. Провести теоретическое исследование технологии Blockchain, ее принципов работы, архитектуры и ключевых характеристик, которые делают ее подходящей для применения в системах голосования.

3. Исследовать существующие Blockchain-платформы и средства разработки децентрализованных приложений, выделив их сильные и слабые стороны в контексте реализации системы голосования.
4. Разработать архитектуру децентрализованного приложения для электронного голосования, учитывая требования безопасности, удобства использования и масштабируемости.
5. Создать прототип децентрализованного приложения, реализующего функции электронного голосования, и провести его тестирование.

В первой главе представлено исследование предметной области, анализ традиционных и современных систем голосования, выявление их проблем и ограничений, а также обзор технологии Blockchain и ее ключевых характеристик.

Вторая глава посвящена исследованию существующих Blockchain-платформ, выбору подходящей среды для разработки приложения и моделирование системы голосования.

В третьей главе описаны процесс реализации прототипа, используемые технологии, а также результаты тестирования приложения.

В заключении сформулированы основные выводы, полученные в результате выполнения работы.

Описание структуры. Работа состоит из введения, трёх глав и заключения.

В первой главе проведен анализ информации: описание предметной области, обзор технологии blockchain, выполнен обзор криптографических протоколов и электронных систем голосования. Проанализировав критику существующих решений систем электронного голосования, можно сформулировать следующие недостатки таких систем и стороны, которые можно улучшить:

- проблемы с безопасностью;
- сложность для некоторых избирателей;
- отсутствие доверия;
- проблемы с анонимностью.

Исходя из общих требований безопасности и отказоустойчивости, всё чаще предлагается протокол на основе технологии Blockchain, который должен положительно сказаться на сформулированных выше недостатках.

Технология Blockchain предлагает решение этой проблемы, предоставляя механизм распределенного доверия. В этой системе записи транзакций хранятся множеством сторон, и каждая из них может проверить, не были ли изменены порядок и отметки времени транзакций. Это создает прозрачность и безопасность, которые значительно улучшают процесс сделок с недвижимостью и другими финансовыми инструментами, минимизируя риск мошенничества и повышая доверие между участниками».

«Децентрализованность системы достигается благодаря технологии одноранговой сети peer-to-peer (P2P), которая отвечает за хранение цепочки блоков транзакций. В этой системе новые блоки периодически добавляются каждым участником, что способствует обновлению и поддержанию актуальности информации о транзакциях. Это создает надежную и безопасную среду для проведения сделок с криптовалютами».

Blockchain является децентрализованной технологией, и P2P-сеть является его основой. В Blockchaine каждый узел (участник сети) хранит копию всей базы данных (реестра транзакций), что позволяет избежать необходимости в центральном сервере для управления данными.

В P2P-сетях узлы могут обмениваться данными напрямую. В контексте Blockchaina это означает, что транзакции могут быть переданы от одного узла к другому без посредников, что ускоряет процесс и снижает затраты.

Также, Blockchain использует P2P-сеть для достижения консенсуса среди узлов. Различные алгоритмы консенсуса (например, Proof of Work или Proof of Stake) помогают обеспечить согласие между участниками сети относительно состояния Blockchaina и предотвращают мошенничество.

Благодаря децентрализованной P2P-структуре Blockchain становится более устойчивым к сбоям и атакам. Если один узел выходит из строя или становится недоступным, остальные узлы продолжают функционировать, обеспечивая целостность сети.

Некоторые Blockchain-протоколы, использующие P2P-технологии, обеспечивают анонимность пользователей, что позволяет им проводить транзакции без раскрытия своей личности.

Простейшая структура сети изображена в соответствии с рисунком 1.1.

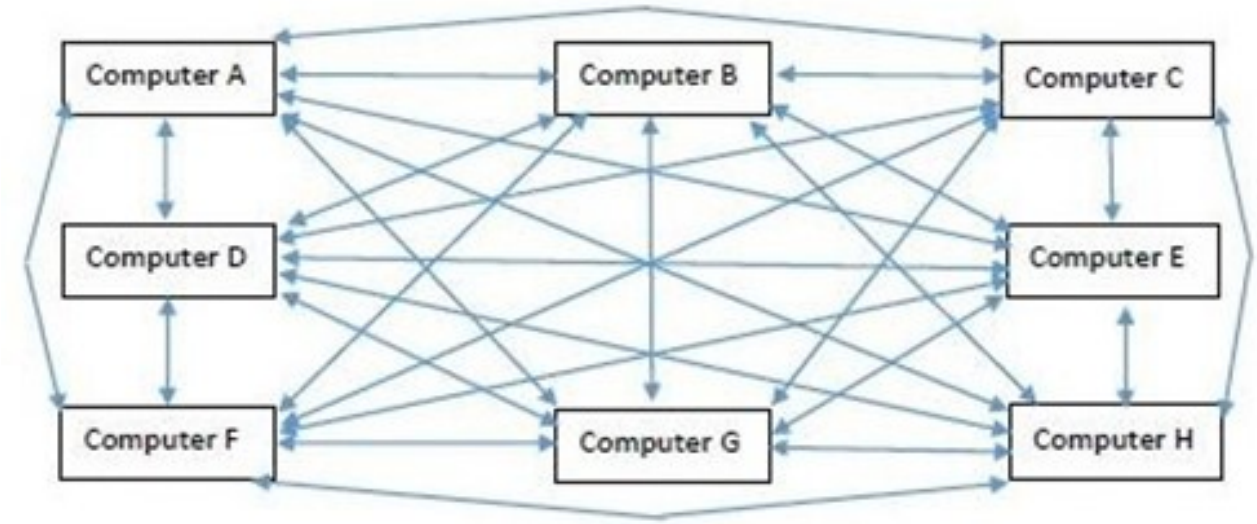

Рисунок 1 — P2P сеть

Во второй главе проведен анализ существующих blockchain-платформ и выполнено проектирование децентрализованного приложения, что позволит реализовать поставленную задачу.

Децентрализованные приложения (или DApps) — это приложения, которые работают на децентрализованных сетях, таких как Blockchain. Они отличаются от традиционных приложений тем, что не зависят от централизованных серверов и обеспечивают пользователям больший контроль над своими данными и взаимодействиями. Давайте

Основные характеристики DApps:

1. Децентрализация. DApps работают на распределённых сетях, что делает их менее уязвимыми к сбоям и атакам.

2. Открытый исходный код. Большинство DApps имеют открытый исходный код, что позволяет сообществу проверять и улучшать их.

3. Использование смарт-контрактов. DApps часто используют смарт-контракты для автоматизации выполнения условий и логики приложения.

4. Токены. Многие DApps используют токены для управления и взаимодействия внутри экосистемы.

Дополнительно, DApps имеют ряд преимуществ по сравнению с традиционными приложениями. Они обеспечивают большую прозрачность, так как все транзакции и изменения записываются в Blockchain и могут быть проверены любым участником сети. Это снижает риск мошенничества и повышает доверие пользователей.

Кроме того, DApps обычно имеют открытую архитектуру, что позволяет разработчикам вносить изменения и улучшения, а также интегрировать новые функции без необходимости централизованного контроля. Это создает экосистему, в которой пользователи могут взаимодействовать друг с другом напрямую, без посредников, что часто приводит к снижению затрат и увеличению скорости операций. В соответствии с рисунком 3 представлен обзор взаимодействия пользователя с децентрализованным приложением.

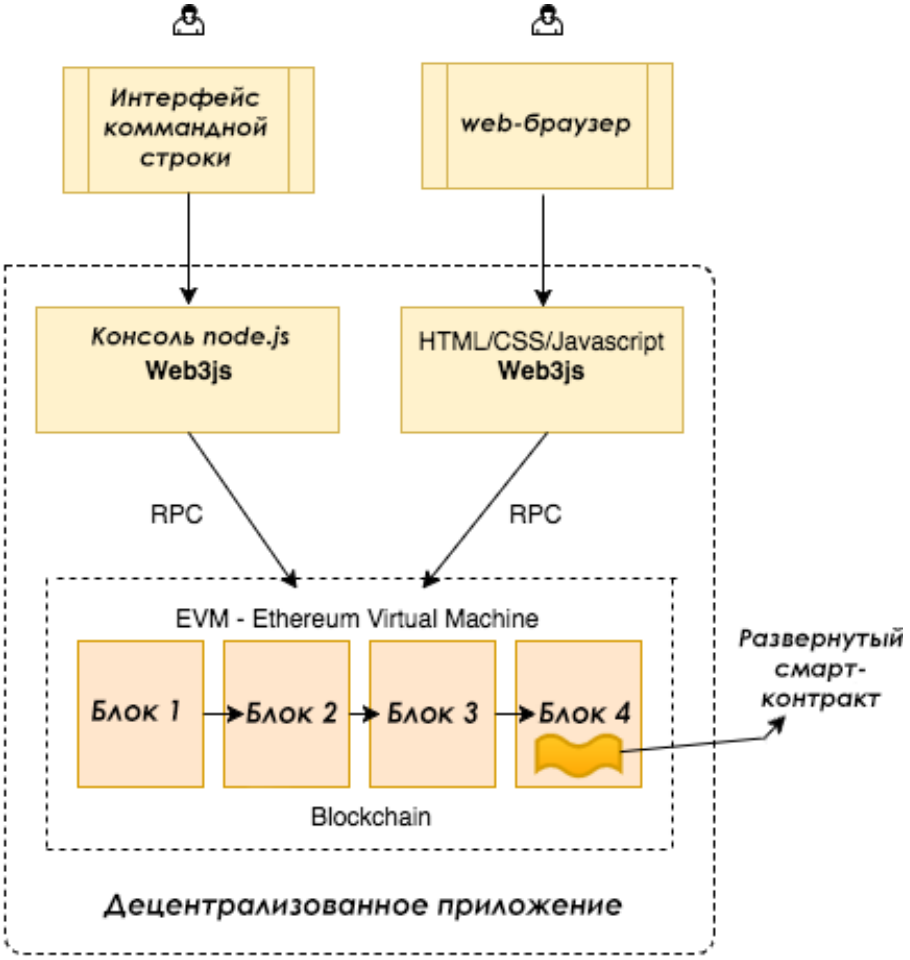

Рисунок 2 — Взаимодействие с Ethereum Dapp

Взаимодействуя с Ethereum Dapp, пользователь имеет возможность обращаться непосредственно к blockchain через специальное программное обеспечение или через интерфейс командной строки на своем устройстве.

Система электронного голосования — это сложный программный продукт, который позволяет избирателям голосовать через электронные устройства. Для успешной разработки такой системы важно провести функциональное и логическое моделирование.

Функциональное моделирование направлено на определение функций и требований системы.

Оно включает в себя следующие этапы:

1. Определение основных функций:

- Регистрация избирателей. Позволяет пользователям зарегистрироваться в системе с использованием личных данных и подтверждения личности;
- Аутентификация. Проверка личности избирателя перед голосованием, чтобы гарантировать, что только зарегистрированные пользователи могут голосовать;
- Голосование. Процесс, в котором избиратели выбирают кандидатов или варианты, представленные на бюллетене;
- Подсчет голосов. Автоматический подсчет голосов и генерация результатов выборов;
- Безопасность. Обеспечение конфиденциальности и целостности голосов, предотвращение мошенничества.

2. Определение пользователей и ролей:

- Избиратели — пользователи, которые регистрируются и голосуют;
- Администраторы (создатели опроса) — лица, ответственные за управление системой, включая регистрацию пользователей и мониторинг голосования.

В соответствии с рисунком 3 представлена модель системы голосования, где показаны основные функции приложения.

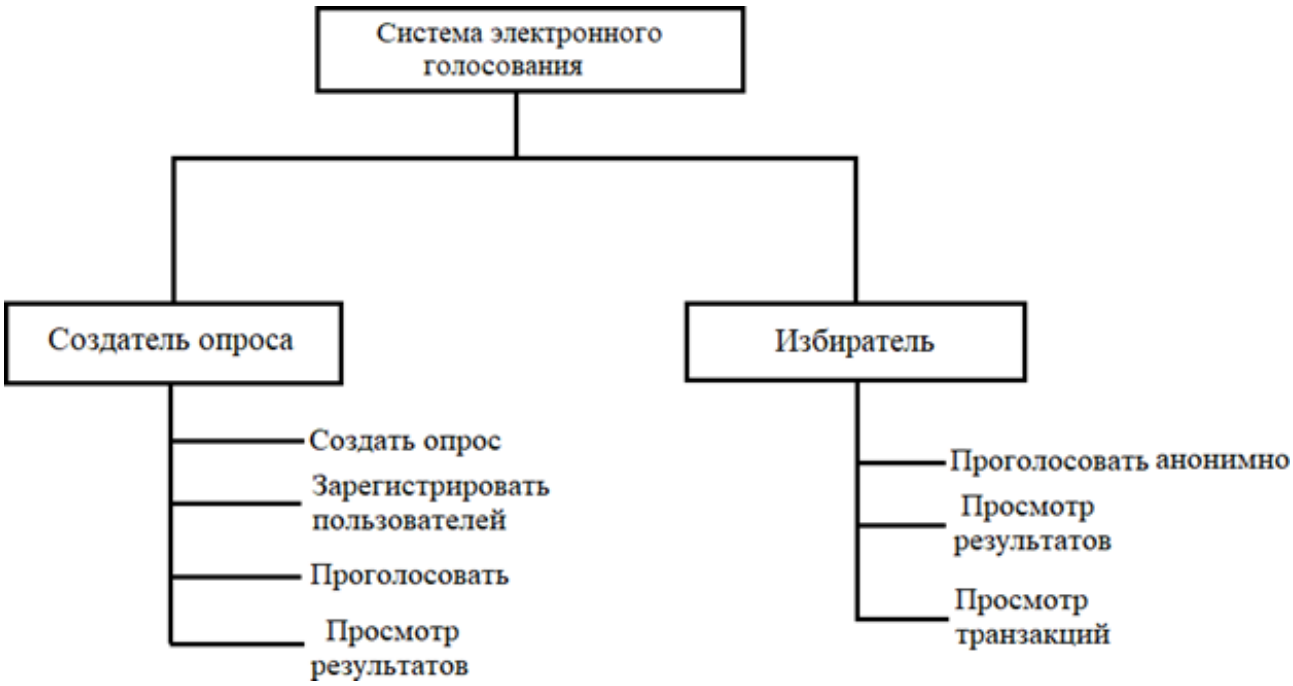

Рисунок 3 — Модель разрабатываемого приложения

При взаимодействии со смарт-контрактом пользователи могут создавать опросы, передавая массив, содержащий название опроса, формулировку вопроса и варианты для голосования. После создания опроса пользователи могут регистрировать новых участников, которые получат право голоса в этом опросе. Кроме того, смарт-контракт предоставляет доступ к данным о результатах голосования.

Логическое моделирование системы электронного голосования — это процесс создания абстрактного представления структуры и поведения системы, отражающего основные компоненты, их взаимодействия и бизнес-правила без привязки к конкретной технологии реализации. Такое моделирование помогает понять, как система должна работать, выявить возможные проблемы и спланировать архитектуру. Для проведения моделирования необходимо создать диаграмму последовательности и диаграмму вариантов использования приложения.

В соответствии с рисунком 4 представлена диаграмма вариантов использования системы голосования пользователем.

Рисунок 4 — Диаграмма вариантов использования

После того, как получено представление об основных функциях системы голосования, можно приступить к её реализации.

Третья глава содержит разработку децентрализованной системы голосования.

Разработка децентрализованного приложения (dApp), особенно в контексте использования смарт-контрактов и Blockchain-технологий, включает несколько ключевых этапов:

- выбор платформы Blockchain;
- проектирование архитектуры;
- разработка смарт-контракта;
- тестирование всего приложения;
- развертывание смарт-контракта.

Смарт-контракт является центральным элементом в разработке децентрализованного приложения. Он представляет собой самовыполняющийся код, который хранится и выполняется на Blockchaine.

После выбора всех необходимых средств для реализации DApps можно смело приступать к его написанию. Этот процесс можно разделить на три этапа:

1. Формирование базовых требований

- точно определить, какие операции контракт должен выполнять (создание, чтение, обновление, удаление данных);
- определить логику управления доступом и права пользователей;

- спланировать обработку транзакций и возможные сценарии взаимодействия.

2. Разработка кода

На основе требований написать код смарт-контракта, используя выбранный язык (например, Solidity).

3. Тестирование

Запускать юнит-тесты для проверки каждой функции. Использовать тестовые сети (testnets) для имитации работы в реальных условиях.

Такой подход помогает создавать надежные и безопасные смарт-контракты, минимизируя риски и повышая качество конечного продукта.

В самом начале в смарт-контракте нужно будет реализовать 5 пунктов:

- конструктор, которому мы будем передавать массив объектов, за которые будет идти голосование;
- присваивание создателю опроса статуса администратора;
- Функция для подачи голосов от зарегистрированных избирателей;
- шифрование выбранного имени объекта голосования для конфиденциальности выбора;
- функция, которая возвращает число голосов у того или иного объекта голосования.

В случае разрабатываемой системы голосования, в начале контракта используется структура, то есть комплексный тип данных, к которым будут ссылаться основные функции.

В данной структуре будут содержаться такие данные, как «Название опроса», «Обсуждаемый вопрос», «Объект голосования 1», «Объект голосования 2», «Объект голосования 3», счетчики голосов за каждого из кандидатов, статус о том, проголосовал тот или иной аккаунт, или нет и создан ли опрос или нет.

Далее идёт конструктор, который вызывается при развертывании смарт-контракта, или в момент, когда необходимо создать новый опрос, в который будет передаваться массив с объектами голосования, названием опроса и обсуждаемым вопросом, то есть все элементы голосования, описанные в структуре.

Далее, создается функция, которая позволяет добавить голос избирателя к счетчику голосов объекта голосования с применением протокола шифрования SHA-3, делается это с помощью функции, предоставляемой языком Solidity «keccak256».

Данная функция позволяет добавить голос к тому или иному объекту голосования, предварительно проверив 2 условия:

1. существует ли опрос, в котором участвует избиратель;
2. была ли задействована эта функция раннее с этого аккаунта.

После проверки этих условий, функция присуждает голосующему аккаунту статус «проголосовавший» и прибавляет один голос к счетчику одного из трех объектов голосования. Описанные в функции голосования условия, определяются функцией, которая проверяет существует ли запрашиваемый опрос и функцией, которая проверяет статус избирателя, была ли произведена передача голоса объекту голосования, или нет.

Также, в смарт-контракте реализуются функции, позволяющие посмотреть варианты голосования в том или ином опросе.

Для того, чтобы осуществить промежуточную проверку работы смарт-контракта применяется инструмент Ganache-CLI, который запускается из bash-консоли и создает локальный тестовый blockchain с десятью аккаунтами. Вызывается он из консоли nodejs, командой ganache-cli, после этого нам показывается 10 тестовых аккаунтов. Тестовая сеть располагается по адресу localhost:8545.

После того, как тестовая blockchain сеть поднята, выполняется компиляция смарт-контракта с расширением «.sol» в интегрированной среде разработки Remix. Далее, необходимо перейти во вкладку «Run» в среде разработки и в поле «Environment» указать адрес созданного с помощью ganache-cli web3-провайдера, то есть http://localhost:8545. После этого, в поле «Account» отобразятся существующие в сети тестовые аккаунты.

После тестирования основных функций следует добавить некоторые ограничения на голосование. Так, на данный момент к такой системе может присоединиться любой участник и проголосовать. Чтобы этого избежать, необходимо продумать как решить данную проблему.

Регистрация предусмотрена белым списком. В данном смарт-контракте реализована возможность создавать неограниченное количество голосований. Это означает, что такой системой могут пользоваться разные организации, разные люди могут создавать свое голосование, свой обсуждаемый опрос и варианты ответов. В разрабатываемой системе голосования было решено назначить каждому опросу администратора, который будет являться создателем опроса. После того, как пользователь создает опрос, ему присваивается статус creator, пользуясь которым он может вызвать функцию создания белого списка и добавить туда необходимые аккаунты в сети blockchain, которым будет позволено голосовать.

При выполнении функция белого листа сначала происходит проверка наличия у пользователя статуса creator. Теперь, для того чтобы участвовать в голосовании, пользователю требуется иметь аккаунт в частной или публичной сети blockchain, в которой развернут данный смарт-контракт и разрешение от создателя, которое дается при внесении пользователя в белый список.

Разработка смарт-контракта закончена.

Итак, разработанный смарт-контракт позволяет совершать следующие манипуляции:

- При создании опроса, пользователь получает статус creator и имеет право дать избирателям доступ к голосованию в своем опросе.
- При создании опроса, в систему передаются данные о названии голосования, предмете спора и варианты ответа.
- Пользователь имеет возможность проголосовать только в том случае, если он добавлен в список избирателей и еще не голосовал.
- Все участники системы имеют право просмотреть результаты голосования.
- Объект голосования шифруется в момент передачи ему голоса по протоколу Secure Hash Algorithm-3 (Keccak), средствами языка solidity, с помощью указания функции «keccak256».

В заключении формулируются выводы, полученные в процессе выполнения работы.

Целью данной работы являлась разработка системы электронного голосования на основе технологии blockchain.

В разработанной системе реализованы функции голосования, регистрация в системе реализована путём создания аккаунта в сети blockchain, в которой развернут смарт-контракт и добавлением пользователя в белый список того или иного созданного опроса.

В ходе выполнения работы был изучен теоретический базис по технологии blockchain, платформы для разработки децентрализованных приложений Ethereum, язык программирования Solidity, на котором происходит разработка смарт-контрактов, регулирующих условия для манипулирования данными в сети blockchain.

Проводился анализ существующих решений систем электронного голосования, выявлены их недостатки в сравнении с аналогичной системой, реализованной с помощью технологии blockchain. Кроме того, были проанализированы различные средства разработки децентрализованного приложения на платформе Ethereum, интегрированная среда разработки Remix, программная платформа node.js, специальные библиотеки node.js, протокол Ethereum go, позволяющий запустить приватную сеть blockchain или подключиться к уже существующей.

Задачи, поставленные в начале работы были выполнены, цель работы была достигнута.