МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**
Кафедра математической кибернетики и компьютерных наук

РАЗРАБОТКА СМАРТ КОНТРАКТА НА ОСНОВЕ БЛОКЧЕЙНА BINANCE SMART CHAIN

АВТОРЕФЕРАТ МАГИСТЕРСКОЙ РАБОТЫ

Студента 2 курса 273 группы
направления 02.04.03 — Математическое обеспечение и администрирование
информационных систем
факультета КНиИТ
Колесникова Виктора Анатольевича

Научный руководитель
доцент, к. ф.-м. н. _____ С. В. Миронов

Заведующий кафедрой
к. ф.-м. н. _____ С. В. Миронов

Саратов 2022

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Актуальность темы. Внедрение технологии блокчейн несет многочисленные выгоды — вот почему к блокчейну пробудился такой неподдельный интерес, охвативший различные сферы, от академической до промышленной. Во всех этих сферах сейчас неустанно исследуют технологию блокчейн. В результате возникло множество консорциумов, рабочих групп, проектов и профессиональных организаций, занятых разработкой и дальнейшим совершенствованием этой технологии.

Отношение общественности к технологии блокчейн кардинальным образом изменилось, когда курс криптовалюты биткойн вырос с $750 в январе до $20 000 в декабре 2017 г. С этого времени диапазон и амбициозность блокчейн проектов растет колоссальными темпами каждый год. Так, например, власти эмирата Дубай объявили о своих планах стать первым в мире эмиратом, экономика которого будет основана на технологии блокчейн. Также ведутся разработки инициатив в области IT, морского судоходства, регистрации предприятий, медицинской документации, финансов, госпрограмм, СМИ и многих других областях. С развитием цифровых валют, продолжают терять актуальность привычные нам деньги. Конечно, в ближайшие годы не стоит ожидать полного перехода на криптовалюты, но уже сейчас пора внедрять блокчейн технологии в повседневную жизнь. В рамках данной работы будет произведен анализ теоретических и практических аспектов, лежащих в основе технологии блокчейн. На базе основных характеристик блокчейна будут рассмотрены ключевые преимущества и недостатки данной новой технологии, чтобы дать оценку целесообразности ее применения в прикладных задачах. Например с системой выплат стипендий студентам университетами. Сейчас данный процесс занимает большое количество времени из-за регистрации студентов в банковской системе и выдачи им физических пластиковых карт. С переходом на криптовалюту, этот процесс будет занимать считанные секунды, а студенты смогут более свободно пользоваться своими средствами, без каких либо ограничений со стороны банковских систем.

Цель бакалаврской работы — является разработка смарконтрактов и сопутствующих скриптов по разрветыванию, реализующих выпуск собственной криптовалюты для факультета компьютерных наук и информационных технологий, а также создание механизмов контроля чеканки монеты и поддержива-

ния ее ликвидности.

Поставленная цель определила **следующие задачи**:

1. произвести анализ теоретических аспектов блокчейна;
2. написать смартконтракт собственного токена под названием CSIT, на основе криптовалютного стандарта ERC20;
3. написать смартконтракт, реализующий распределение токена CSIT;
4. создать торговую пару для возможности купли продажи токена с другой криптовалютой USDT с помощью децентрализованной биржи Pancake Swap;
5. создать смартконтракт StakingService, реализующий возможность фарминга токенов CSIT на паре CSIT/USDT.

Методологические основы, описывающие технологии, лежащие в основе блокчейна и методов разработки смарт контрактов представлены в работах таких таких авторов как, Имран Башир и Андреас Антонопулос. В книге Дипака Хазанчи подробно рассматривается применение блокчейна для бизнеса, здравоохранения, науки, идентификации, управления, образования, общественных благ и различных аспектов культуры и коммуникации. В книге «Ethereum Smart Contract Development in Solidit» Гэвин Чжэн вводит все основные понятия и термины блокчейна Ethereum, необходимые для понимания общей концепции а так же описывает, как настроить разработку на Solidity, описывает основные особенности данного языка программирования и приводит примеры различных контрактов, а также как децентрализованные приложения могут взаимодействовать со смартконтрактами. В своих книгах Нараян Прусти и Андреас Антонопулос также рассматривают, что такое блокчейн, как он обеспечивает целостность данных, и как создавать прикладные блокчейн-проекты на платформе Ethereum [1–8].

Практическая значимость магистерской работы. Разработанный программный продукт может быть использован как плацдарм для дальнейших работ по внедрению криптовалюты в систему выплат стипендий студентам.

Структура и объем работы. Магистерская работа состоит из введения, трех разделов, заключения, списка использованных источников и трех приложений. Общий объем работы – шестьдесят семь страниц, из них шестьдесят страниц – основное содержание, включая девятнадцать рисунков, список использованных источников информации — двадцать пять наименований.

1 КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Первый раздел «Блокчейн» посвящен истории появления и этапам развития технологии блокчейн, берущей свое начало с Bitcoin, а также теоретическим аспектам, лежащим в основе блокчейна и возможностям их реализации. Подробно рассматривается главный механизм — децентрализация.

В 2008 году вышла статья под названием «Биткойн. Децентрализованная электронная денежная система» на тему пиринговой электронной валюты, написанная под псевдонимом Сатоши Накамото (Satoshi Nakamoto) [9]. Именно в ней появился термин «цепочка блоков». Никому не известно, кто именно скрывается под именем Сатоши Накамото. Термин «цепочка блоков» с годами изменился, и теперь этот феномен называется «блокчейн». У технологии блокчейн масса прикладных вариантов, которые могут быть реализованы в различных сегментах экономики. Так, считается, что достигнутое в финансовом секторе значительное усовершенствование транзакций и расчетов позволило как следует ускорить эти процессы и сократить издержки на них.

Определение **блокчейна** (blockchain):

- *Упрощенное*: блокчейн — это непрерывно растущая, безопасная, разделяемая система учета записей, где у каждого пользователя данных есть такая копия этих данных, обновить которую можно лишь при условии, что на это согласятся все стороны, участвующие в транзакции.
- *Техническое*: блокчейн — это пиринговый криптографически защищенный распределенный, (практически) неизменяемый реестр, поддерживающий только добавление блоков, обновляемый лишь в результате соглашения (договоренности) между всеми участниками.

Блокчейну сопоставляют следующие элементы: адрес, транзакция, блок, пиринговая сеть, скриптовый язык или язык программирования, виртуальная машина, машина состояний, узел, умный контракт.

Децентрализация является основным преимуществом, предоставляемым технологией блокчейн, ее ключевым сервисом. Блокчейн задумывался как идеальный транспортный механизм для создания платформ, которые не нуждаются в промежуточных звеньях и могут функционировать со множеством разных ведущих узлов, выбираемых через механизм консенсуса. Эта модель делает возможной неограниченную конкуренцию за право принимать решения. Такая конкуренция управляется механизмом консенсуса, и наиболее распространен-

ный метод, который при этом используется, известен под названием PoW (англ. Proof of Work – доказательство выполнения работы).

Второй раздел «Ethereum» подробно описывает механизмы блокчейна Etherium, необходимые для реализации поставленных задач, а именно механизм формирование адресов и приватных ключей, механизм формирование транзакций для реализации запросов в блокчейне, а также возможность написание смартконтрактов с помощью Ethereum Virtual Machine.

Стек технологий блокчейна Ethereum состоит из различных компонентов. Во главе стоит сам блокчейн, работающий поверх пиринговой сети. Также существует клиент Ethereum (как правило, это Geth), который работает на узлах и подключается к пиринговой сети, чтобы загрузить блокчейн и сохранить его локально. Он имеет разные функции, такие как майнинг и управление учетными записями. Локальная копия блокчейна регулярно синхронизируется с сетью. Еще одним компонентом является библиотека web3.js, которая позволяет взаимодействовать с клиентом geth через интерфейс удаленного вызова процедур (англ. Remote Procedure Call или RPC).

Формальный список всех высокоуровневых элементов, присутствующих в блокчейну Ethereum:

- ключи и адреса;
- учетные записи;
- транзакции и сообщения;
- криптовалюта Ether (токены);
- EVM;
- смарт-контракты.

Язык программирования Solidity открывает перед разработчиками децентрализованных приложений безграничные возможности. Впервые публичный блокчейн Etherium, предназначенный для создания смарт-контрактов и децентрализованных приложений, был предложен в 2013 году Виталиком Бутериным. Валютные токены в Ethereum называются Ether (эфир).

Третий раздел «Криптовалюта CSIT» посвящен реализации трех смарт-контрактов: токена CSIT, расписания и стейкинг сервиса. Токен CSIT был написан, опираясь на стандарт криптовалют ERC20 (Ethereum Request for Comments). ERC-20 был создан в 2015 году Виталиком Бутериным и Фабианом Фогельстеллером с целью предложить относительно простой формат

для создания токенов на Ethereum. С помощью среды разработки Truffle был создан первичный проект:

— `/contracts` — каталог контрактов, который включает не скомпилированные контракты на языке Solidity;
— `/migrations` — каталог миграций, предназначенных для развертывания контрактов в блокчейн;
— `/test` — каталог автотестов, необходимый для тестирования полей и методов контрактов;
— `/build` — каталог артефактов сборки;
— `truffle-config.js` — корневой файл проекта, отвечающий за настройку соединения с нодами блокчейна и информацию о компиляторе и деплое.

Смартконтаркты будут распространяться по лицензии MIT и компилироваться версиями Solidity >=0.7.0 <0.8.0.

В директории `/contracts` был создан контракт `CSIT.sol`, который содержит реализацию токена. Контракт токена CSIT наследуется от контрактов `ERC20` и `RecoverableBy Owner` и хранит в себе следующие ключевые поля:

— `csitSchedule` — адрес контракта, реализующего распределение токенов. Данный адрес можно будет изменить создателем контракта, если возникает необходимость изменить эмиссию криптовалюты;
— `stakingService` — адрем контракта, реализующего фарминг на паре `CSIT/USDT`;
— `lastSupplyTime` — переменная 40 бит, хранящая последнее время выпуска токена, необходимое для дальнейшей эмиссии.

Для отлавливания изменений, вносимыми методами в контракт, используются ивенты: ScheduleChanged, StakingServiceChanged, которые позволяют пробрасывать логи при совершение транзакций.

Далее следует секция с описанием конструктора, данная функция будет вызвана единоразово при деплое контракта. В конструкторе последовательно происходят следующие действия:

— На вход в качестве параметра принимается адрес контракта, распределяющего токены.
— Вызывается конструктор контракта ERC20, где первым параметром передается название токена `CS&IT Token`, а вторым символ `CSIT`. Благодаря вызову конструктора ERC20 доступны все методы данного стандарта,

а также главный дополнительно метод `_mint`, призванный производить быстрый выпуск токенов.

— Происходит вызов ивента по смене адреса контракта расписания.
— Объявляется внутренняя переменная контракта `csitSchedule`.
— Благодаря функции `_mint` происходит первоначальная эмиссия токенов `CSIT` в размере 1000 единиц на адрес создателя контракта.

Теперь опишем три функции необходимые для корректной работой контракты:

— `changeSchedule` — функция, которая изменяет адрес контракта, устанавливающего распределение токенов.
— `changeStakingService` — функция, которая изменяет адрес контракта стейкинг сервиса, устанавливающего распределение токенов при стейкинге на паре `CSIT/USDT`.
— `supplyCsit` — функция, вызываемая извне стейкинг сервисом для выпуска токена `CSIT`.

Контракт `CsitSchedule.sol` реализует эмиссию токенов и содержит единственное поле `csitPerSecond`, обозначающее количество выпускаемых токенов в секунду. Этот параметр будет инициализирован в конструкторе.

Контракт имеет следующие функции:

— `updateRate` — функция, вызываемая только создателем контракта, изменяющая значение `csitPerSecond`.
— `makeProgress` — функция по расчету количеству токенов, которые необходимо выпустить между интервалами `lastSupplyTime` и `lastSupplyTime + time`.

После того как, контракты написаны, необходимо произвести их развертывание в блокчейне. В качестве целевой сети был выбран блокчейн Binance Smart Chain Tesnet — по своей сути это среда для тестирования децентрализованных приложений на BSC. BSC Testnet предоставляет такие же условия что и в основная сети BSC Mainnet. Здесь разработчики могут развертывать любые смарт-контракты, чтобы опробовать различные функции перед официальным запуском в основной сети, заплатив комиссию с помощью Testnet BNB, а не ценной криптовалютой.

Чтобы получить на счет Testnet BNB, неообходимо воспользоваться сервисом Binance Smart Chain Faucet, который предоставляет возможность полу-

чать основыне токены на сети BSC, а именно: BNB, BTC, BUSD, ETH, DAI и многие другие.

После того, как были получены `BNB` для оплаты комиссии, необходимо развернуть контракты в блокчейне, для этого воспользуемся функционалом, предоставляемым truffle — миграцией. В директорию `/migrations` добавим файлы по развертыванию контрактов. Первым идет деплой расписания, так как данный контракт необходим для деплоя токена.

Для того чтобы другие лица могли получить токен CSIT, необходимо создать контракт, организующий торговлю на интересующей нас паре, а именно `CSIT/USDT`. Данный функционал в DeFi принято организовывать с помощью пула ликвидности. Пул ликвидности — это совокупность криптовалютных токенов, заблокированных в смартконтракте. Пулы ликвидности используются для обеспечения децентрализованной торговли, кредитования и многих других функций.

Для создания пула ликвидности на паре `CSIT/USDT` на сети BSC Testnet воспользуемся функционалом, предоставляемым децентрализованной биржей PancakeSwap. Пройдя на вкладку Liquidity нажмем на кнопку Add liqudity и выберем нужные адреса.

После создания пула ликвидности на адрес, инициализирующий создание пула, должно поступить n-ое количество LP-токенов, в соответствии с алгоритмами PancakeSwap, а также с адреса создателя списаться средства, добавленные в пул. Убедимся в данных операциях по транзакциям, связанным с адресом контракта LP-токена на паре `CSIT/USDT` — `0xA0f0d339b8a860Ee8Ba6121DD462b7A3DCc3D9c1`, далее данный контракт для краткости изложения будет обозначаться `CsitUsdtLp`.

Для привлечения внимания и ажиотажа к своей валюте в области DeFi и увеличения на нее спроса, принято запускать контракты по стейкингу на пулах ликвидности с интересуемой валютой. Суть данного контракта заключается в том, что адреса, являющиеся держателями LP-токенов получают вознаграждения в соответствии с их долей в общем пуле. Для создателя токена это является рычагом для увеличения ликвидности своей валюты, а следовательно и роста ее цены.

Добавим в директорию проекта `/contracts` файл `CsitStakingService.sol`, реализующий контракт фарминга.

Объявим все необходимые поля в контракте:

— `address immutable public csit` — адрес токена `CSIT` в котором полагается выплачивать вознаграждение;
— `address immutable public stakingToken` — адрес `CsitUsdtLp` токена, за стейкинг которого пользователь будет получать вознаграждение;
— `State public state` — состояние в котором находится контракт стейкинга в данный момент;
— `mapping(address => Staker) public stakers` — сопоставление адреса стейкера с его состоянием. Условый словарь, где ключом является `address` стейкера, а значением структура `Staker`.

Далее следует секция с описанием конструктора, данная функция будет вызвана единоразово при деплое контракта и принимает на вход два аргумента которыми инициализирует соответствующие поля:

— `_csit` — адрес токена `CSIT`;
— `_stakingToken` — адрес токена `CsitUsdtLp`.

Для пользователей будут доступны следующие четыре функции:

— `function stake(uint128 amount) external` — функция для стейкинга разрешенного количества токенов `CsitUsdtLp` в размере, передаваемом через параметр `amount`;
— `function unstake(uint128 amount) external` — функция для анстейка токенов `CsitUsdtLp` в размере, передаваемом через параметром `amount`, возвращая их снова на баланс пользователя;
— `function claim(uint128 amount) external` — функция, обновляющая текущий показатель выплаченных вознаграждений `reward` у пользователя и переводящая их на адрес, вызывающий функцию.
— `function totalStaked() external view returns (uint128)` — функция, возвращающая общее количество средсв, застейканные на контракте всеми пользователями на данный момент.

Самым простым способом взаимодействия с контрактами является работа с ними через BscScan. Для этого необходимо залить ABI контракта с помощью API-ключа в BscScan. Следующим этапом был рассмотрен реальный пример по стейкингу на паре с доказательствами корректности работы.

ЗАКЛЮЧЕНИЕ

В ходе данной работы были рассмотрены концепции, лежащие в основе технологии блокчейн, а именно:

— просмотрены теоретические аспекты заложенные в блокчейн;

— рассмотрены все ключевые характеристики блокчейна Etherium.

На основе данных характеристик были выявлены преимущества и недостатки данной технологии и области ее применения.

С практической точки зрения были реализованы и успешно развернуты на бокчейне BSC контракт собственной криптовалюты с именем CSIT, а так же контракты, реализующие эмиссию токена и стейкинг сервиса.

Отдельные части магистерской работы были опубликованы на студенческой конференции КНиИТ.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 *Bashir, I.* Блокчейн: архитектура, криптовалюты, инструменты разработки, смарт-контракты / I. Bashir. — 2 изд. — Москва: ДМК Пресс, 2019.

2 *Bashir, I.* Mastering Blockchain / I. Bashir. — Packt>, 2020.

3 *Zheng, G.* Ethereum Smart Contract Development in Solidity / G. Zheng, L. Gao, L. Huang, J. Guan. — Gateway East, Singapore 189721, Singapore: Springer, 2021.

4 *Prusty, N.* Building Blockchain Projects / N. Prusty. — Livery Place, 35 Livery Street, Birmingham B3 2PB, UK: Packt>, 2017.

5 *Antonopoulos, A. M.* Mastering Bitcoin / A. M. Antonopoulos. — 1005 Gravenstein Highway North, Sebastopol, CA 95472: O'Reilly, 2014.

6 *Khazanch, D.* Blockchain 3.0 for Sustainable Development / D. Khazanch, A. K. Vyas, K. K. Hiran, S. Padmanaban. — Berlin/Boston: Gruyter, 2021.

7 *Crosby, M.* Blockchain technology: Beyond bitcoin / M. Crosby, Nachiappan, P. Pattanayak, S. Verma, V. Kalyanaraman // *Berkeley*. — June 2016. — P. 16.

8 *Antonopoulos, A. M.* Mastering Etereum: Implementing Digital Contracts / A. M. Antonopoulos, G. Wood. — 1005 Gravenstein Highway North, Sebastopol, CA 95472: O'Reilly, 2018.

9 *Nakamoto, S.* Bitcoin: A peer-to-peer electronic cash system / S. Nakamoto. — 2008. — P. 9.