Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича

Кафедра сетей связи и передачи данных

Разработка публичного децентрализованного реестра с использованием технологии blockchain

Выполнил: студент гр. ИКВТ-62 Гарифуллин В.Ф.

Научный руководитель: к.т.н., доцент Владимиров С.С.

Санкт-Петербург 2020

СПб ГУТ)))

Цель работы

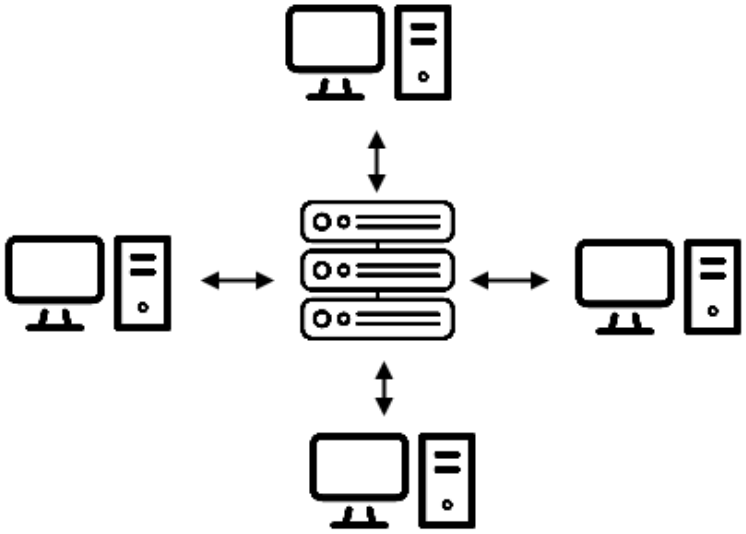
- Проектирование и разработка кроссплатформенного программного обеспечения для организации публичного децентрализованного реестра с использованием технологии blockchain.

Задачи работы

1. Анализ преимуществ и недостатков существующих электронных реестров.
2. Анализ существующих решений для организации публичных децентрализованных реестров.
3. Рассмотрение принципов организации распределённых реестров на основе технологии blockchain.
4. Сравнительный анализ и выбор технологий для использования в реестре.
5. Разработка программного обеспечения для организации реестра и графического интерфейса для удобной работы с ним.

Электронные реестры

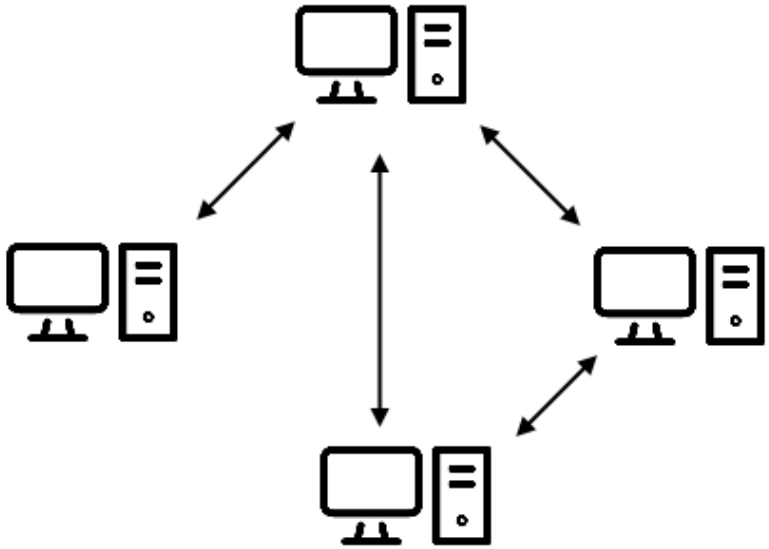
- Преимущества:
  - Экономия времени
  - Экономия средств
- Недостатки:
  - Ненадёжность
  - Непрозрачность



Клиент-серверная архитектура

Распределённые системы

- Существующие решения для распределённого обмена данными:
  - Gnutella
  - BitTorrent
  - IPFS

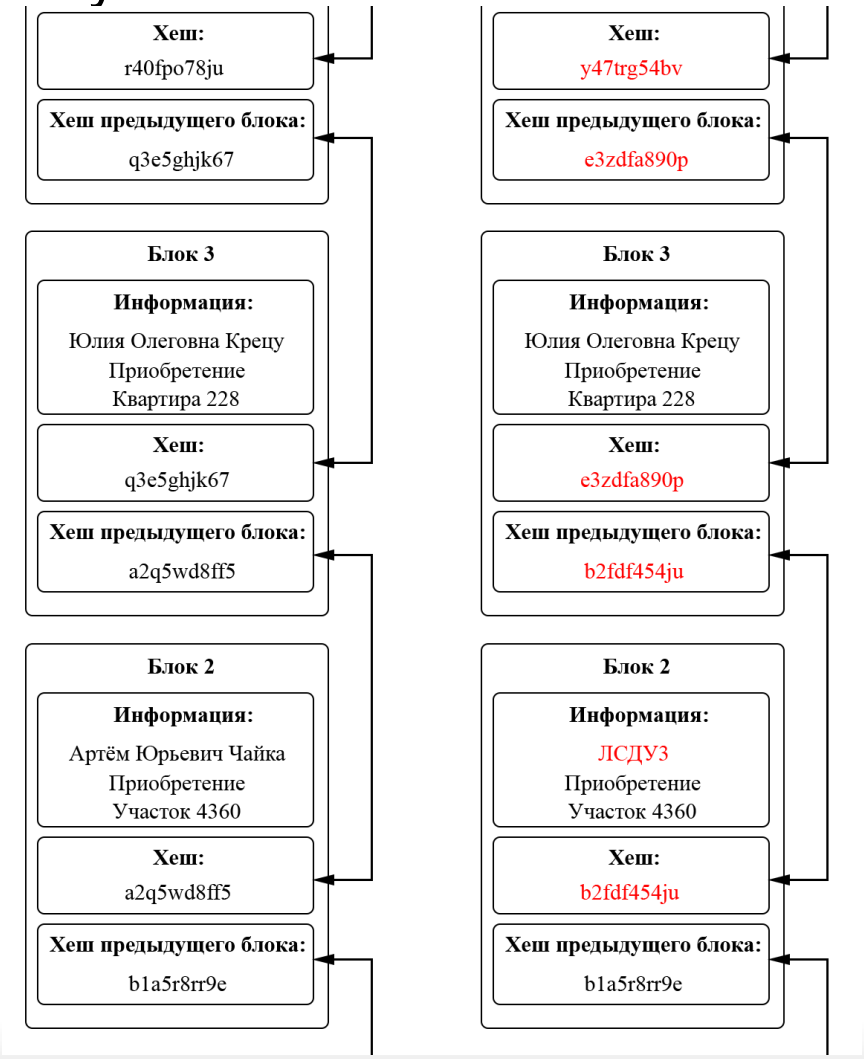


Децентрализованная сеть

- Отсутствующий функционал:
  - Ограничение на добавление информации только уполномоченными лицами
  - Обеспечение достоверной последовательности записей

Технология Blockchain

- Blockchain – цепочка блоков, которые связаны между собой с помощью хеш-сумм.



Выбор хеш-функции

Параметр	SHA-256	SHA-384	SHA-512
Устойчивость к коллизиям	Да	Да	Да
Время хеширования миллиона строк одинаковой длины, с	9,88	36,14	35,98
Размер хеш-суммы, бит	64	96	128

Цифровая подпись (1)

- Криптографические алгоритмы с открытым ключом:
  - RSA
  - DSA
  - Алгоритмы, основанные на эллиптических кривых

Длина открытого ключа в битах при сопоставимой криптостойкости

Эллиптическая криптография	RSA
163	1024
233	2240
283	3072
409	7680
571	15360

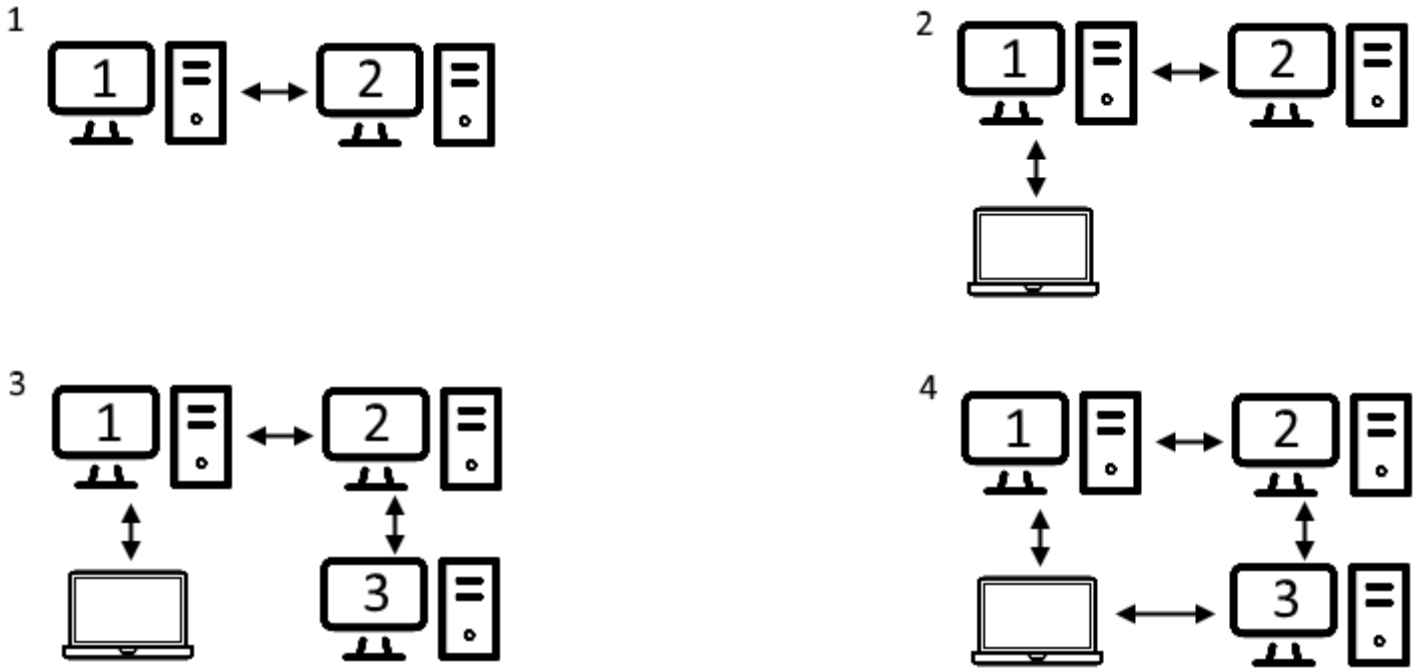
Цифровая подпись (2)

- Ed25519 – схема подписи EdDSA, основанная на эллиптической кривой Curve25519 и использующая SHA-512.

- Преимущества:
  - Эллиптическая кривая Curve25519 считается полностью безопасной.
  - Не используются ветвления и потенциально опасные операции с памятью.
  - В процессе подписи не используется генератор случайных чисел.
  - Используемая длина открытого ключа в 256 бит обеспечивает криптостойкость, сопоставимую с RSA с длиной ключа в 3000 бит (разница в 12 раз).

Структура сети

- Два типа узлов: обычные узлы хранения и мастер-узлы.
- Обычные узлы обращаются к мастер-узлам для получения новых блоков, а также для получения информации о новых мастер-узлах.
- Процесс добавления нового мастер-узла:



Транзакции

- Транзакция – данные, добавляемые пользователем в сеть, и служебная информация.
- Транзакции нужны для предотвращения коллизий.
- Блоки состоят из транзакций.
- Ещё не занесённая в блок транзакция называется неподтверждённой.
- Данные, хранящиеся в транзакции:
  - Временная метка
  - Тип транзакции
  - Передаваемая информация
  - Публичный ключ отправителя
  - Хеш транзакции
  - Цифровая подпись отправителя

Алгоритм консенсуса (1)

- Алгоритм консенсуса нужен для предотвращения коллизий.
- Алгоритм консенсуса позволяет удостовериться, что участник сети, подписавший новый блок, сделал это правомерно.
- Алгоритм доказательства полномочий (Proof of Authority). подразумевает наличие ограниченного числа валидаторов.
- Валидаторы – мастер-узлы, которые имеют право подписывать блоки.

Алгоритм консенсуса (2)

- Используется следующий алгоритм выбора валидатора для подписи блоков:
  1. Фиксируем текущую временную метку
  2. Делим на время валидации одного валидатора
  3. Округляем до ближайшего целого
  4. Берём результат пункта 3 по модулю числа валидаторов
  5. Получившийся результат – номер валидатора, который может подписывать блоки в данный момент

Временная метка, мс	Время GMT+3	Валидатор
1590266468000	20:41:08	1
1590266469000	20:41:09	2
1590266470000	20:41:10	2
1590266471000	20:41:11	0
1590266472000	20:41:12	0
1590266473000	20:41:13	1

Генезис-блок

- Генезис-блок – самый первый блок в цепочке.
- Данные, хранящиеся в генезис-блоке:
  - Время
  - Хеш генезис-блока
  - Генезис-транзакция:
    - Тип транзакции (genesis)
    - Открытые ключи валидаторов
    - Открытые ключи отправителей (людей, имеющих право добавлять записи в реестр)

Структура блоков

- Блоки хранят следующие данные:
  - Время подписи блока
  - Транзакции (неограниченное количество)
  - Публичный ключ валидатора
  - Хеш предыдущего блока
  - Хеш данного блока
  - Цифровая подпись валидатора

Добавление новой информации

1. Узел добавляет в транзакцию данные, подписывает её и отправляет мастер-узлам.
2. Мастер-узлы проверяют транзакцию и заносят в пул неподтверждённых транзакций.
3. Валидатор, который согласно алгоритму консенсуса в данный момент может подписывать блоки, проверяет транзакции, которые находятся в пуле неподтверждённых транзакций, формирует из них блок, подписывает его и заносит в цепочку.
4. Все остальные узлы во время синхронизации получат этот блок, проверят его и занесут его в цепочку.

Разрабатываемые приложения

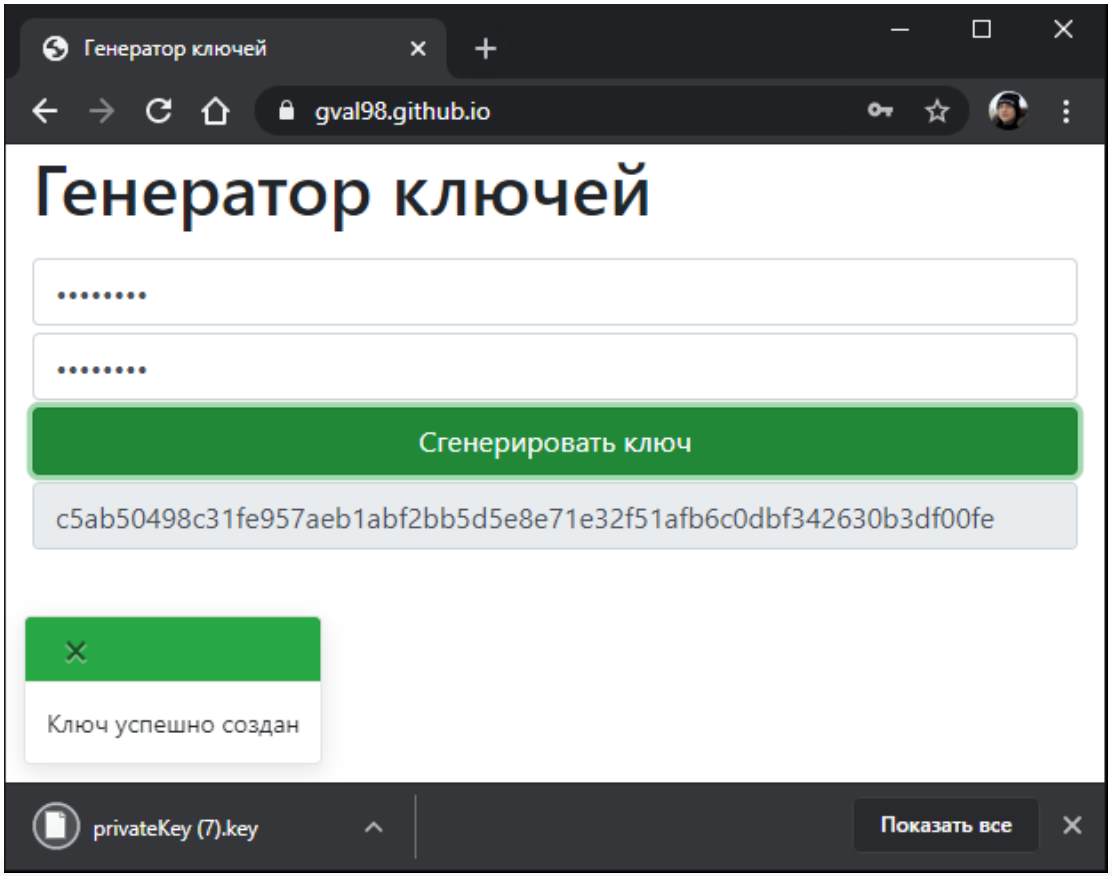
Интерфейс	Консольный	Графический	Веб
Поддерживаемые операционные системы	Windows, Linux	Windows, Linux	Все ОС с наличием браузера
Хранение всей цепочки блоков	Да	Да	Нет
Проверка блоков и транзакций	Да	Да	Нет
Взаимодействие с сетью	Со всеми мастер-узлами	Со всеми мастер-узлами	С одним мастер-узлом
Запуск обычного узла	Нет	Да	Нет
Запуск мастер-узла	Да	Да	Нет
Запуск узла валидации	Да	Нет	Нет
Запуск веб-интерфейса	Да	Нет	Нет
Отправка транзакций	Нет	Да	Да
Удобный просмотр транзакций	Нет	Да	Да

Средства разработки

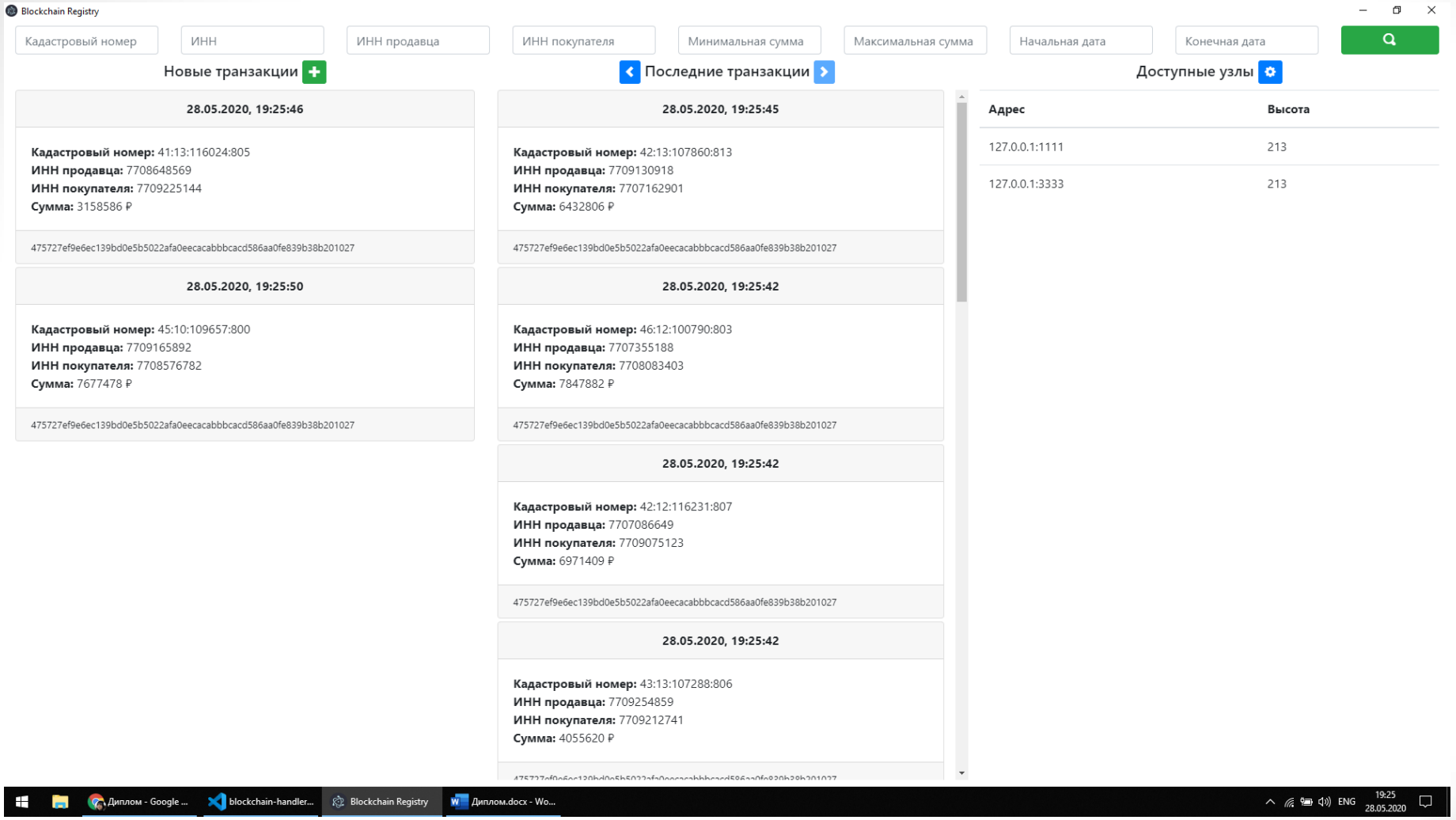
- Критерии выбора средств разработки:
  - Поддержка асинхронности
  - Кроссплатформенный графический интерфейс
- Для разработки было решено использовать следующие средства:
  - Язык программирования JavaScript
  - Среда исполнения Node.js
  - Фреймворк Electron для создания графического приложения
  - Фреймворк Bootstrap для создания интерфейса графического приложения и веб-версии

Хранение и создание ключей

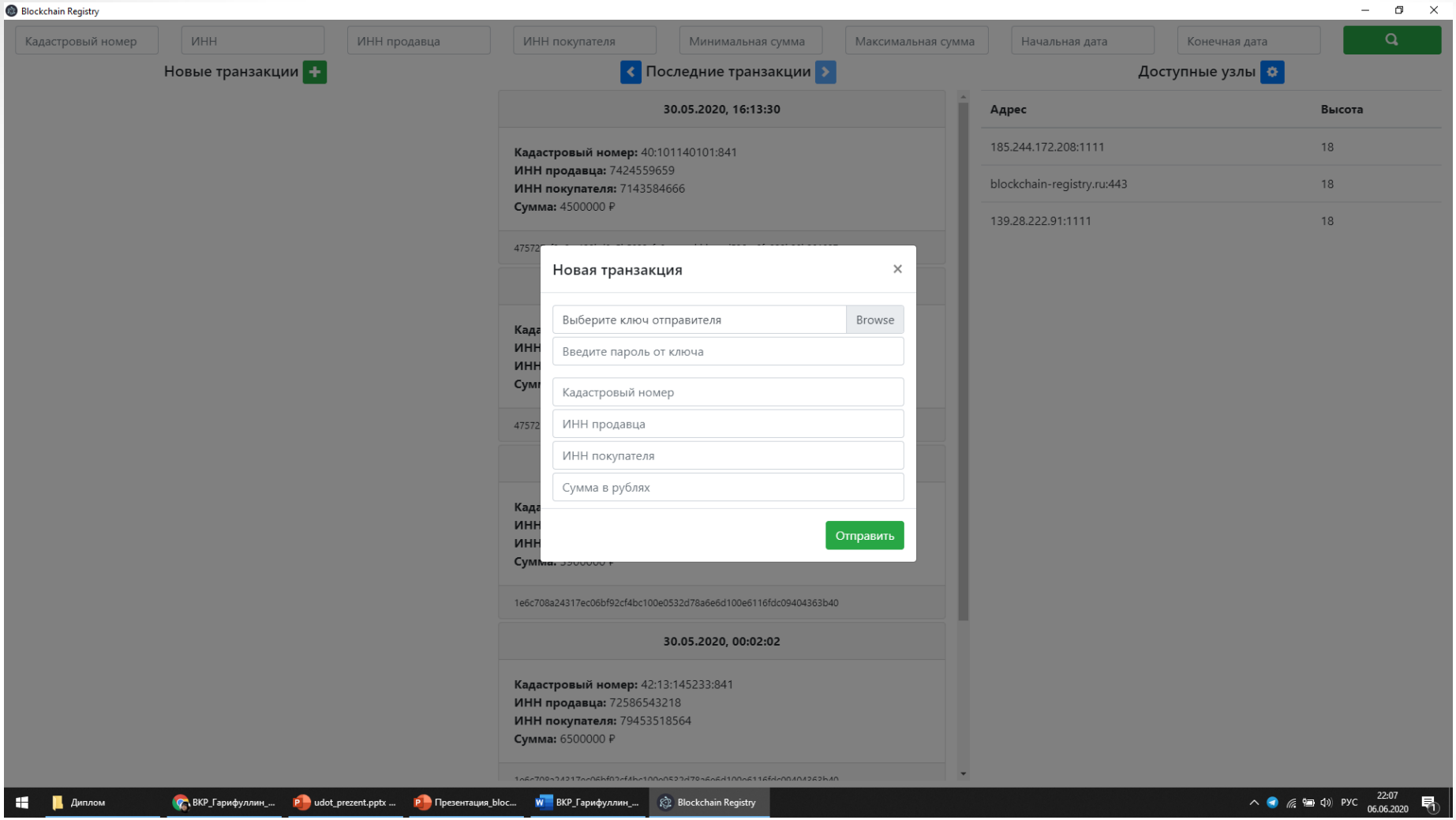
- Все закрытые ключи хранятся в зашифрованном с помощью алгоритма симметричного шифрования AES виде.
- Для создания ключей было разработано консольное и веб-приложение, доступное по адресу https://gval98.github.io



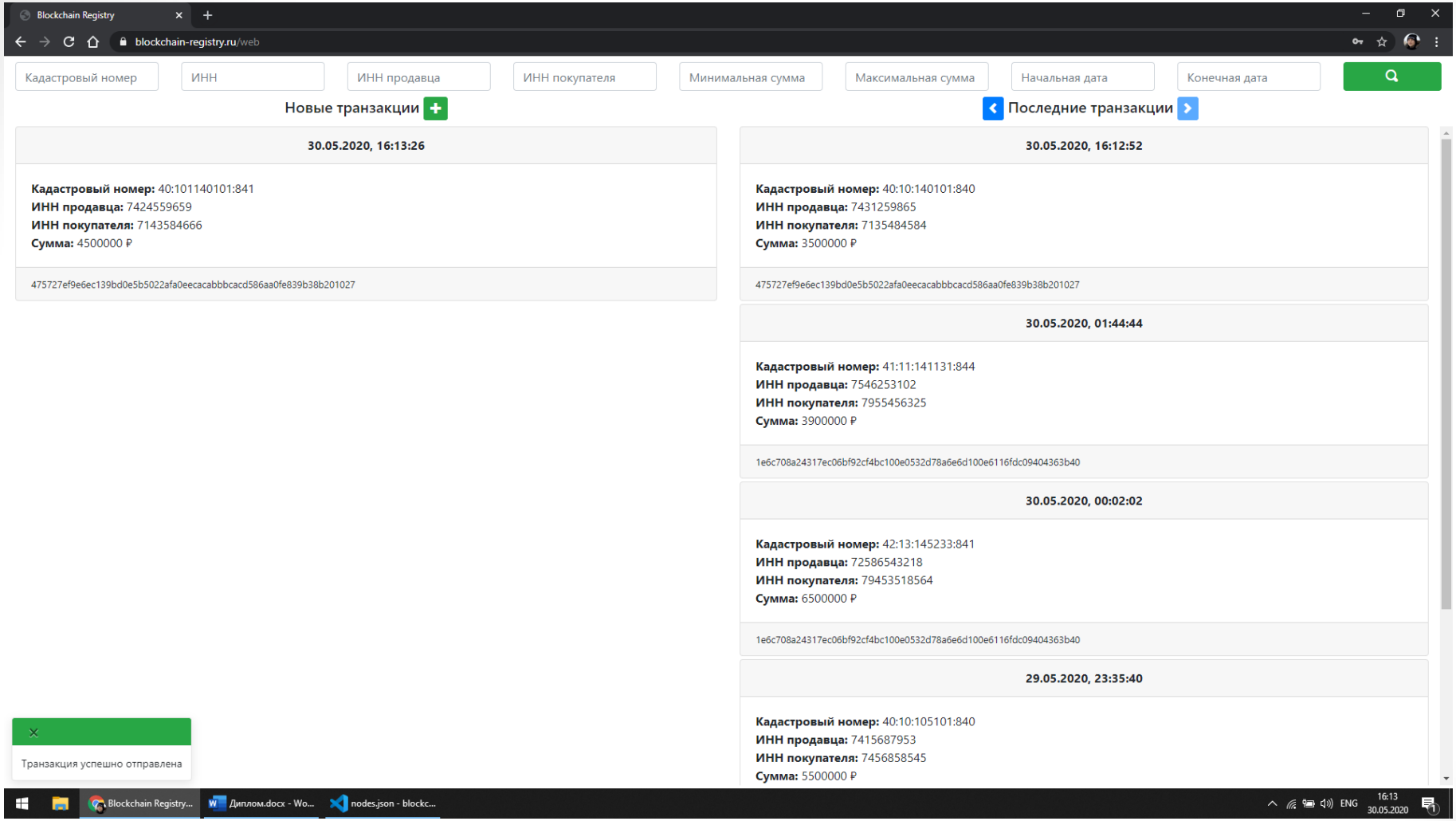
Графическое приложение (1)



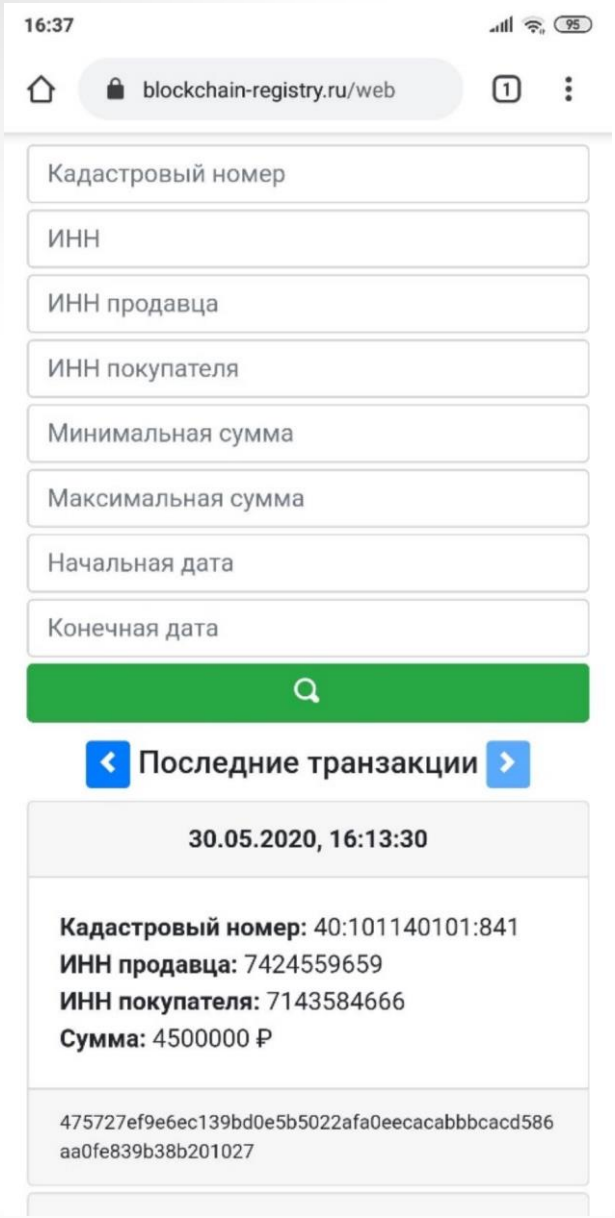
Графическое приложение (2)



Веб-приложение (1)



Веб-приложение (2)



Веб-приложение для уже запущенной сети доступно по адресу: https://blockchain-registry.ru/web

Заключение

- В ходе выполнения работы были решены следующие задачи:
  - Рассмотрены существующие решения для децентрализованного обмена данными, в том числе с использованием технологии blockchain.
  - Произведён сравнительный анализ технологий, используемых в blockchain проектах и децентрализованных сетях.
  - Разработано кроссплатформенное программное обеспечение для организации публичных децентрализованных реестров и графический интерфейс для удобной работы с ним.
- Разработанные приложения могут быть успешно применены для создания публичных реестров на базе технологии Blockchain с целью их децентрализации, увеличения надежности и прозрачности.

Кафедра сетей связи и передачи данных

Спасибо за внимание!

Разработка публичного децентрализованного реестра с использованием технологии blockchain

Выполнил: студент гр. ИКВТ-62 Гарифуллин В.Ф.
Научный руководитель: к.т.н., доцент Владимиров С.С.

Санкт-Петербург 2020