

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 ИНТЕРАКТИВНОЙ ПЛАТФОРМЫ ДЛЯ СИМУЛЯЦИИ
ТОРГОВЛИ ЦЕННЫМИ МЕТАЛЛАМИ**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 451 группы
направления 09.03.04 — Программная инженерия
факультета КНиИТ
Мягкова Андрея Владимировича

Научный руководитель
доцент, к. ф.-м. н.

С. В. Миронов

Заведующий кафедрой
к. ф.-м. н., доцент

А. С. Иванов

Саратов 2020

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Постановка задачи	4
2 Анализ, дизайн и программирование веб-приложения	6
ЗАКЛЮЧЕНИЕ	12
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	13

ВВЕДЕНИЕ

Для подготовки к IT-конференции «ЮКОН» компании «Норникель — ОЦО» было необходимо создать различные активности для привлечения посетителей конференции к стенду компании.

Актуальность данной темы состоит в том, что несмотря на широкое распространение электронных валют и интегрированность технологии блокчейн в различные технологические сферы, продажа металлов с использованием смарт-контрактов - это интересная и новая идея. При достаточном развитии платформа может стать востребованной как в нашей стране, так и за рубежом.

Предмет исследования. В ходе написания бакалаврской работы было составлено подробное описание проекта с перечислением аналогов, указанием способов хранения информации, описанием дизайна и адаптивности под различные устройства, API-запросами и системными требованиями.

Цель бакалаврской работы — разработать интерактивную платформу для симуляции торговли цветными металлами и ознакомления с историей и деятельностью компании. Для достижения данной цели были поставлены следующие задачи:

- разработать клиентскую часть приложения (интерфейс и обмен данных с сервером);
- разработать структуру базы данных приложения;
- разработать серверную часть приложения (обработка полученных данных, взаимодействие с клиентом, базой данных и блокчейном, генерация котировок металлов).

Структура и объем работы. Бакалаврская работа состоит из введения, 2 разделов, заключения, списка использованных источников и приложения. Общий объем работы — 93 страницы, из них 43 страницы — основное содержание, включая 18 рисунков и 1 таблицу, цифровой носитель в качестве приложения, список использованных источников информации — 20 наименований.

1 Постановка задачи

Разработать веб-приложение с использованием технологии блокчейн для привлечения посетителей конференции к стенду компании и их знакомства с ее историей и деятельностью. Приложение должно представлять собой платформу для торговли ценными металлами, добываемыми компанией.

В приложении должны быть доступны функции:

- Просмотра табло курса металлов с данными, получаемыми с сервера;
- Покупки и продажи металла через определенные промежутки времени;
- Просмотра истории своих транзакций;
- Ответа на вопросы об истории компании.

Проект является симуляцией процесса покупки и продажи металлов и не оперирует реально существующими ископаемыми и/или денежными валютами. Действительным аналогом в будущем может стать швейцарская платформа Open Mineral, которая собирается задействовать блокчейн в торговле полезными ископаемыми [1].

Хранение информации должно осуществляться:

- в базе данных, расположенной на сервере;
- в блокчейне, разработанном с использованием смарт-контрактов.
- на устройствах пользователей (только данные для аутентификации с разрешения пользователя).

Дизайн приложения должен быть выполнен в корпоративных цветах компании. Верстка веб-приложения должна учитывать смену ориентации экрана (альбомную и портретную), в зависимости от нее размещать основную информацию на странице и/или изменять вид меню навигации. Также оно должно быть адаптировано для корректного отображения на различных устройствах, имеющих разные разрешения экрана.

Для синхронизации изменений веб-приложение должно обмениваться информацией с сервером, который будут хранить часть информации, а остальную отправлять в блокчейн. На сервере будет храниться:

- логин и пароль пользователя;
- ФИО пользователя;
- название организации, которую пользователь представляет на конференции.
- историю операций пользователя в приложении (получение валюты при

регистрации и правильном ответе на вопросы, покупку и продажу металлов);

- данные кошелька пользователя в блокчейн.

В блокчейне будут храниться:

- баланс валюты и металлов пользователя;
- история операций пользователя в системе.

Системные требования к клиенту:

- Разрешение экрана: 960x640 px и более.
- Операционная система: Microsoft Windows XP SP 3.0 (и выше); Mac OS X 10.3 (и выше), Android 2.3 (и выше), iOS 4.0 (и выше).
- Интернет браузер: Firefox 3.1 (и выше); Microsoft Edge 40.15063 (и выше); Safari 3.0 (и выше), Google Chrome 8.0.552 (и выше) и др.

Системные требования к серверу:

- Операционная система Windows Server 2012 R2 (и выше);
- Оперативная память 12GB (и выше);
- Емкость жесткого диска 256 ГБ (и выше);

2 Анализ, дизайн и программирование веб-приложения

Для разработки приложения использованы следующие языки программирования, среды разработки и программные компоненты:

- Python 3.7 (и выше) — интерпретируемый язык программирования общего назначения высокого уровня [2];
- Java (JDK 11.0.7 и выше) — объектно-ориентированный параллельный язык программирования общего назначения, основанный на классах [3];
- Javascript (ECMAScript 2016 и выше) — высокоуровневый мультипарадигменный язык программирования, соответствующий спецификации ECMAScript [4];
- Solidity 0.5.6 — высокоуровневый объектно-ориентированный язык программирования для написания смарт-контрактов платформы Ethereum [5];
- JetBrains IntelliJ IDEA 2019.1.4 — используется для веб-разработки [6];
- JetBrains Pycharm 2019.1.4 — используется для разработки серверной части приложения [7];
- SQLite 3.20.0 (и выше);
- Go Ethereum 1.93 — интерфейс командной строки для запуска узла Ethereum, реализованного в Go.

Подключаемые библиотеки:

- Flask (Python) — микро-фреймворк, написанный на Python;
- Web3 (Python) — библиотека Python для взаимодействия с Ethereum;
- Blockchain (Python) — модуль Python для взаимодействия с API Blockchain.info;
- Bootstrap (HTML/CSS) — веб-фреймворк с открытым исходным кодом для разработки веб-приложений;
- Highcharts (Javascript) — библиотека для построения графиков.

Структура базы данных

- таблица «auth» хранит информацию о пользователе приложения;
- «history» — информацию о транзакциях пользователя ;
- «topicQuestion» — информацию о вопросе ;
- «question» — информацию для доступа к вопросу и о награде за правильный ответ;
- «metals» — информацию о металлах ;
- «points» — информацию о котировках металлов ;

- «personsPruff» — историю ответов на вопросы;
- «listOperations» — список возможных операций, выполняемых в транзакции.

Интерфейс веб-приложения

1. Ознакомительная страница: неавторизованному пользователю при сканировании QR-кода откроется графически оформленная страница с рекламой приложения и стенда компании. На ней присутствует краткая информация (см. рисунок 1):
 - о том, как зарегистрироваться в приложении;
 - для чего нужны QR-коды;
 - о главном функционале приложения — покупке и продаже ценных металлов;
 - о ценных призах, стимулирующих пользоваться приложением.
2. Страница регистрации: процесс регистрации пользователей осуществляется путем ввода логина, пароля, ФИО и названия организации, которую пользователь представляет на конференции.
3. Страница аутентификации: при вводе корректного логина и пароля пользователь будет авторизован в приложении.
4. Личный кабинет (см. рисунок 2):
 - верхняя панель с кнопкой «Выход» — при нажатии на нее доступ к приложению теряется, пока пользователь снова не пройдет процедуру аутентификации;
 - боковое меню со основным списком страниц приложения;
 - блок с выводом информации о балансе валюты и ценных металлов на кошельке пользователя;
 - панель с графиками ценных металлов, имеющая три вкладки (графики обновляются в реальном времени).

В нижней части страницы реализован функционал для совершения операций с металлами (см. рисунок 3). Выбирается необходимый металл, его количество, операция (купить или продать) и время совершения транзакции.

5. История транзакций: просмотр совершенных пользователем транзакций.
6. Страница ответа на вопросы: отображение тестового вопроса, определяемого уникальным ключом, который прописан в QR-коде. Пользователь

- должен выбрать один из вариантов ответа и нажать кнопку «Ответить».
7. Страница вывода сообщений: вывод сообщений, связанных с ответами на вопросы.
 8. Табло с котировками металлов: блок с тремя графиками котировок металлов для показа на стенде компании.

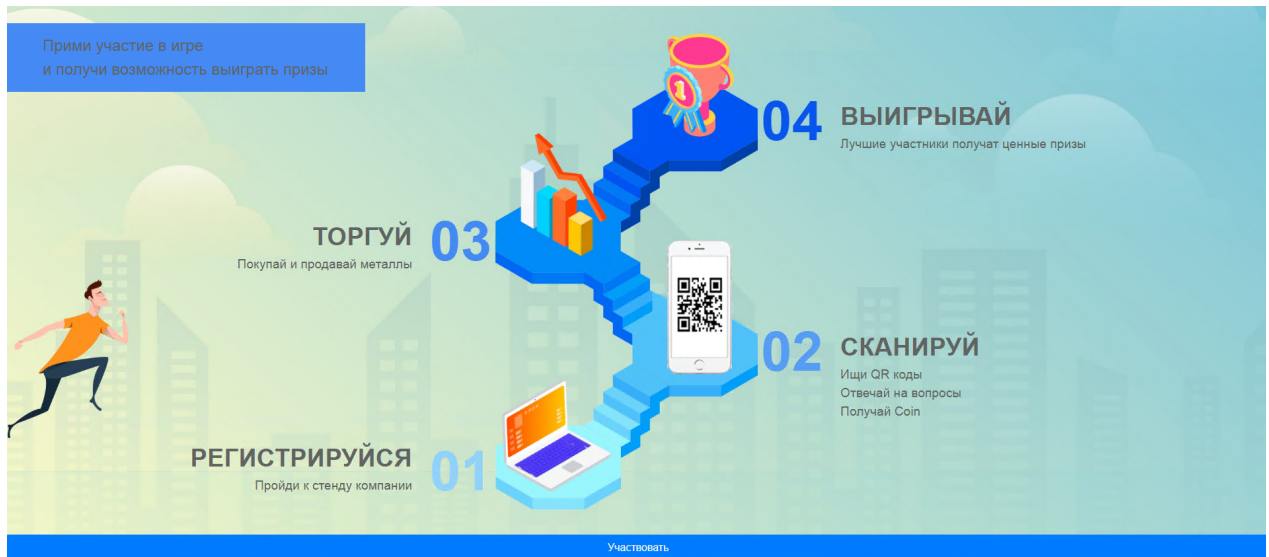


Рисунок 1 – Ознакомительная страница

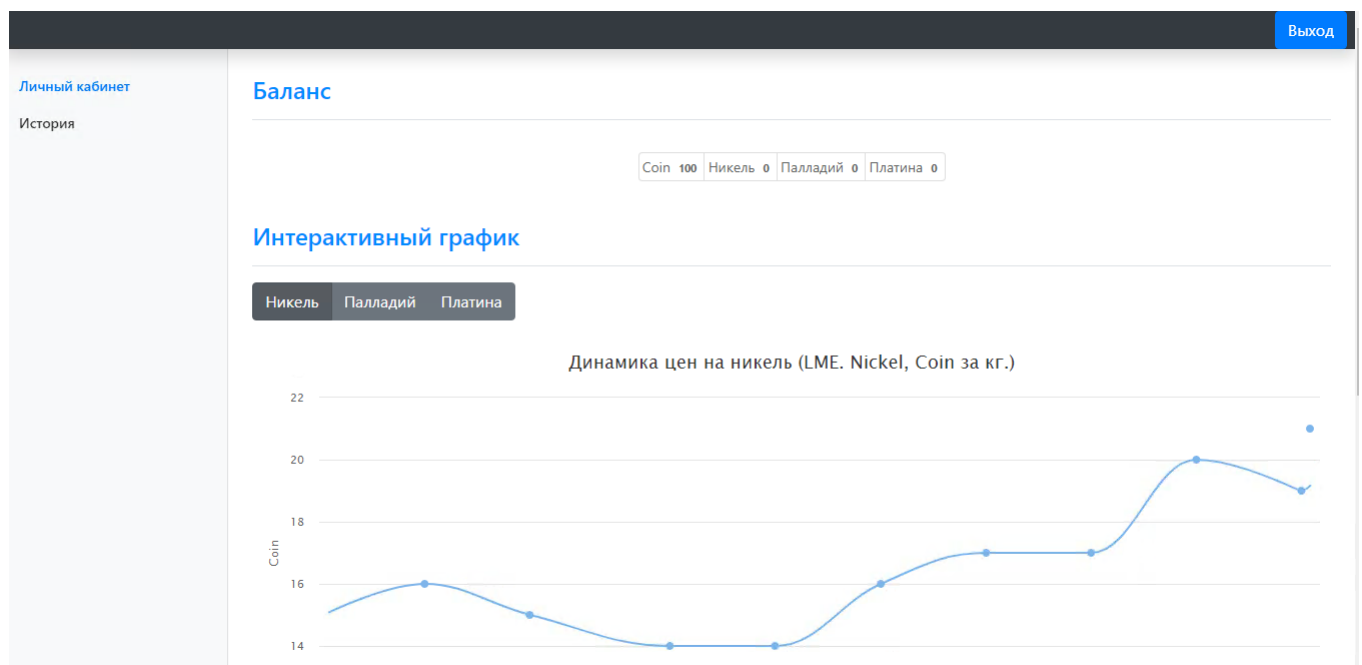


Рисунок 2 – Личный кабинет

API приложения

Веб-приложение взаимодействует с сервером через запросы API, написанном на фреймворке Flask.

Выберите металл

Никель ▾

Количество

1

Операция

Купить ▾

Когда

Через 1 минуту ▾

Отправить запрос

Рисунок 3 – Панель совершения транзакций

- **/get_text** — Обращается к блокчейну, используя зашифрованные и хранящиеся в базе данных данные от кошелька пользователя и получает данные о кошельке пользователя в блокчейне: статус кошелька, баланс валюты и металлов, информацию о последней транзакции.
- **/auth** — Аутентификация пользователя: обращается с базе данных и сравнивает полученные в запросе данные для входа с зашифрованными в базе. При наличии соответствия происходит успешная авторизация.
- **/home** — баннер с приглашением зарегистрироваться;
- **/** — Выполнение операции купли/продажи металлов: функция получает данные о пользователе, его балансе и параметрах операции (количество металла, действие — купля или продажа), отправляет запрос в блокчейн, и, при успешном выполнении транзакции, добавляет запись в историю транзакций пользователя.
- **/register** — Вывод страница регистрации (GET-запрос).
Регистрация пользователя (POST-запрос): при корректности введенной информации (ФИО, логин, пароль, организация) на пользователя создает кошелек в блокчейне, начисляется стартовая валюта, данные пользователя записываются в базу данных.
- **/vopros/<indexed>** — Страница с определенным вопросом об истории компании (GET-запрос).

Обработка ответа на вопрос (POST-запрос): обращение к базе данных и проверка корректности полученного ответа. Если ответ правильный, то отправляется запрос в блокчейн для начисления положенного за ответ количества металлов, добавляются записи в базу данных о начислении металла. В противном случае выводится сообщение о том, что пользо-

ватель ответил неправильно. В независимости от правильности ответа в базу данных записывается сообщение о том, что пользователь уже ответил на данный вопрос, во избежание перебора ответов.

- **/history** — Если пользователь авторизован, то обращается к базе данных для получения списка транзакций пользователя и возвращает страницу с нем. В противном случае возвращает страницу входа в приложение.
- **/closed** — Выход пользователя из аккаунта и возвращает страницу входа в приложение.
- **/dashboard** — Получение топа пользователей по заработанной валюте: функция обращается к блокчейну, получает данные о кошельках с наибольшим количеством валюты, и запрашивает данные об этих кошельках из базы данных.
- **/secondtable** — Страница с графиками котировок металлов в реальном времени.

Взаимодействие с блокчейном

Функция «register» создает новый кошелек для пользователя. Каждому пользователю при регистрации начисляется 100 единиц валюты с главного кошелька. Для того, чтобы пользователь смог совершать транзакции, ему начисляется gas - плата или стоимость, необходимая для успешного проведения транзакции или выполнения контракта на платформе Ethereum.

Функция «allBalance» возвращает баланс валюты всех кошельков.

Функция «getBalance» возвращает баланс валюты определенного кошелька.

Функция «getBalanceMetals» возвращает баланс металлов определенного кошелька.

Функция «settingsContract» совершает разблокировку кошелька перед совершением транзакции и начисляет gas.

Функция «sellMetal» совершает продажу указанного в транзакции металла.

Функция «buyMetal» совершает покупку указанного в транзакции металла.

Функция «openFiles» загружает на сервер файл со смарт-контрактом.

Очередь транзакций

Функция «gup» каждую минуту проверяющая список транзакций и вы-

полняет те, время которых меньше или равно текущего. При это проверяется, хватит ли пользователю валюты для покупки металла или металла для продажи.

Функция «commit» изменяет статус транзакции в истории на «Выполнено».

Функция «restartDaemon» обновляет очередь вновь появившимися транзакциями со статусом «Ожидает».

Функция «checkKey» проверяет наличие элемента с заданным ключом в очереди.

Генерация котировок металлов

Функция «paragraph» получает данные с внешнего сервера и разделяет их на три списка (для каждого из трех металлов) и сохраняет их в базе данных.

Взаимодействие с базой данных

Функция «__init__» открывает соединение с базой данных.

Функция «insertUpdate» вставляет запись в базу данных.

Практическое использование приложения

В Саратове была проведена пятая ежегодная региональная ИТ-конференция «ЮКОН», в которой приняли участие около 1,5 тыс. молодых специалистов, выпускников и студентов саратовских ВУЗов. Главным партнером конференции стала компания «Норникель — ОЦО».

На стенде компании участники участие в различных интерактивных активностях. Стенд Норникеля вызвал большой интерес у участников мероприятия. «Норникель — ОЦО» представил бизнес-игру на платформе блокчейн, которую разработали специально для мероприятия сотрудники Middle-офиса ОЦО. В течение всего дня гости стенда с помощью своих смартфонов в режиме реального времени играли на виртуальной бирже металлов, зарабатывая игровую валюту. В финале бизнес-игры были определены трое лидеров, которые получили призы от компании [8].

ЗАКЛЮЧЕНИЕ

В ходе работы были освоены навыки разработки на языках программирования Python, Java, JavaScript, языке разметки документов HTML и языке описания внешнего вида документа CSS. Была разработана интерактивная платформа для симуляции торговли ценными металлами и знакомства пользователей с компанией.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Биржа open mineral начнет торговать полезными ископаемыми через блокчейн[Электронный ресурс]. — URL: <https://ru.ihodl.com/topnews/2018-07-11/\birzha-open-mineral-nachnet-torgovat-poleznymi-iskopaemymi-\cherez-blokchejn/> (Дата обращения 08.05.2020). Загл. с экр. Яз. рус.
- 2 *Lutz, Mark*. Learning Python / Mark Lutz. — Sebastopol: O'Reilly, 2013.
- 3 Java [Электронный ресурс]. — URL: <https://crispersoft.com/ru/java-programming-language/> (Дата обращения 15.05.2020). Загл. с экр. Яз. рус.
- 4 Javascript [Электронный ресурс]. — URL: <https://developer.mozilla.org/ru/docs/Web/JavaScript> (Дата обращения 15.05.2020). Загл. с экр. Яз. рус.
- 5 Solidity [Электронный ресурс]. — URL: <https://solidity.readthedocs.io/en/v0.6.8/> (Дата обращения 15.05.2020). Загл. с экр. Яз. англ.
- 6 IntelliJ idea [Электронный ресурс]. — URL: <https://www.jetbrains.com/ru-ru/idea/documentation/> (Дата обращения 17.05.2020). Загл. с экр. Яз. рус.
- 7 Pycharm [Электронный ресурс]. — URL: <https://www.jetbrains.com/ru-ru/pycharm/documentation/> (Дата обращения 17.05.2020). Загл. с экр. Яз. рус.
- 8 «НОРНИКЕЛЬ — ОЦО» ВЫСТУПИЛ ПАРТНЕРОМ РЕГИОНАЛЬНОЙ ИТ-КОНФЕРЕНЦИИ [Электронный ресурс]. — URL: <http://ssc.nornik.ru/956/> (Дата обращения 21.05.2020). Загл. с экр. Яз. рус.