Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«БАЙКАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «БГУ»)

Кафедра математических методов и цифровых технологий

Направление подготовки: 09.03.03 Прикладная информатика
Направленность (профиль): Информационные системы и технологии в управлении

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
БАКАЛАВРСКАЯ РАБОТА

на тему: «СИСТЕМА ЦИФРОВОЙ ИДЕНТИФИКАЦИИ ПОЛЬЗОВАТЕЛЕЙ И ОБЪЕКТОВ НА ОСНОВЕ БЛОКЧЕЙН»

Заведующий кафедрой:	22.02.2024 X Родионов Родионов А.В. Заведущий кафедрой Подписано: Родионов Алексей	к.т.н., А.В. Родионов
Руководитель:	22.02.2024 X Родионов Родионов А.В. Заведущий кафедрой Подписано: Родионов Алексей	к.т.н., А.В. Родионов
Нормоконтролер:	22.02.2024 X Юргина Е.С. Юргина Е.С. Нормоконтролер Подписано: Юргина Евгения	Е.С. Юргина
Обучающийся группы: ЗИС—19	22.02.2024 X 0151731 Деркунов М.Ю. Студент ЗИС-19 Подписано: 0151731	Деркунов М.Ю.

Иркутск, 2024 г.

ОГЛАВЛЕНИЕ

СПИСОК СОКРАЩЕНИЙ

1. УИИ – университет искусственного интеллекта;
2. Блокчейн — выстроенная по определённым правилам непрерывная последовательная цепочка блоков, содержащих свои хэш значения и предыдущих [6];
3. LSM – learning management system – система управления обучением;
4. UID – уникальный идентификатор;
5. NFT – non—fungible token – невзаимозаменяемый актив [5];
6. Хэш — хеш—функция, осуществляющая преобразование массива входных данных произвольной длины, в выходную битовую строку установленной длины, выполняемое определённым алгоритмом шифрования;
7. 3НФ – третья нормальная форма базы данных;
8. Токен — единица учёта, не являющаяся крипто—валютой, предназначена для представления цифрового баланса в некотором активе

ВВЕДЕНИЕ

В 2024 году в сфере технологий, несомненно, доминирует разработка нейронных сетей, машинного обучения и всем, что связано с обработкой больших данных, но в сфере защиты данных нет лучшего «инструмента», чем блокчейн технологии.

Выбор темы исследования напрямую связан с созданием безопасной системы идентификации на образовательной платформе университета искусственного интеллекта (УИИ) для обеспечения современных требований пользователей и стремления к цифровой трансформации.

Проводя исследования выявил, что системы на основе блокчейна, способны гарантировать подлинность данных, предотвращать поддельность и обеспечивать высокий уровень защиты личной информации не только пользователей, но доменов, зарегистрированных как участник сети. Такой подход, по—моему, особенно важен в условиях увеличения популярности онлайн образования и расширения образовательной платформы УИИ. В рамках исследования произвел разработку и внедрение системы цифровой идентификации пользователей на базе блокчейна, это обеспечило не только эффективное управление учетными записями, но и повысило общую безопасность.

Целью данной исследовательской работы является разработка и внедрение системы цифровой идентификации на основе блокчейна, с учетом специфики образовательной среды университета. Главной задачей исследования является повышение безопасности и надежности идентификации пользователей, персонализации личных достижений и подтверждения квалификации.

Для достижения поставленной цели предполагается провести анализ текущей организационной структуры университета, выявить ключевые бизнес процессы, а также выявить проблемы, связанные с неэффективностью текущей системы идентификации и хранении персональных данных. Такой анализ позволит учесть особенности учебного процесса и определить требования к будущей системе.

Объектом исследования выступает образовательная среда университета, включая студентов, преподавателей, административные и образовательные подразделения.

Предметом исследования данной дипломной работы является система цифровой идентификации пользователей и объектов на основе блокчейн технологии, применяемая в образовательной среде университета. Под системой цифровой идентификации понимается комплекс программно—технических средств, включающий в себя Смарт контракты на блокчейне (в данном случае, на базе Ethereum), механизмы подписи, транзакций и взаимодействия с базой данных, которые эффективно взаимодействуют с учебным процессом, начиная от момента регистрации студента и заканчивая завершением учебных курсов.

При исследовании предмета и объекта внимание уделяется анализу организационной структуры университета, бизнес—процессам в образовательной среде, а также существующим проблемам в системе идентификации. Особое внимание уделяется теоретическому обоснованию выбора блокчейн технологии, исследованию ее применимости в контексте образования, а также определению требований к системе цифровой идентификации, учитывая особенности учебного процесса.

Разработка системы цифровой идентификации становится основным этапом исследования, включая проектирование архитектуры, создание смарт контрактов, транзакций, механизмов подписи, а также механизмов защиты утечки данных. Работа с базой данных и интеграция системы в учебный процесс также являются составной частью объекта исследования.

Тестирование и анализ разработанной системы цифровой идентификации на основе блокчейн, а также рассмотрение возможности интеграции в сторонние системы образуют завершающий этап исследования.

1.АНАЛИЗ ОРГАНИЗАЦИИ

1.1 Общие сведения об организации

Университет искусственного интеллекта (УИИ) представляет собой образовательное учреждение, специализирующееся на обучении и исследованиях в области искусственного интеллекта (ИИ). Основан с целью удовлетворения растущего спроса на квалифицированных специалистов в данной области. УИИ ориентирован на создание интеллектуальной среды для развития научных исследований и инноваций в области ИИ.

Организационно, УИИ структурирован по принципу факультетов и институтов, специализирующихся в различных областях ИИ. Преподаватели и исследователи научного коллектива являются профессионалами в своей области. Университет также активно сотрудничает с промышленностью и исследовательскими организациями, обеспечивая студентам возможности практического применения полученных знаний.

Целью УИИ является формирование образовательной среды для развития профессиональных навыков в области ИИ и создание общества, осознающего важность данной области для будущего. Университет стремится стать центром интеллектуального роста и развития в области искусственного интеллекта, к повышению уровня безопасности в различных сферах своей деятельности с применением блокчейн технологии, в частности, в финансовых операциях и транзакциях в учебном процессе.

На текущем этапе УИИ успешно внедрил технологию блокчейн, основанную на платформе Ethereum, в область финансовых операций. Это позволяет обеспечить безопасность и прозрачность финансовых транзакций, минимизировать риск мошенничества и предотвращать несанкционированные финансовые операции внутри учебного учреждения.

Однако, следует отметить, что внедрение блокчейн технологии в учебном процессе УИИ все еще находится на начальном этапе. На данный момент блокчейн применяется в ограниченном объеме в учебных финансовых операциях, исключая другие аспекты учебной деятельности. Впредь, планируется расширение

применения технологии блокчейн на различные аспекты управления учебным процессом, в том числе идентификацию студентов, аутентификацию документов и обеспечение целостности данных.

Таким образом, УИИ ориентирован на поэтапное внедрение блокчейн технологии с акцентом на обеспечение безопасности в секторе финансовых операций, и в долгосрочной перспективе — на расширение ее применения для обеспечения безопасности и эффективности в различных аспектах учебного процесса.

УИИ стремится использовать технологию блокчейн не только для обеспечения безопасной идентификации, но и для создания персонализированных портфолио студентов. Целью данной инициативы является предоставление студентам возможности эффективного демонстрирования своих достижений и компетенций, а также обеспечение заказчиков проверяемой информацией о подлинности портфолио и оценке уровня знаний студентов.

С использованием блокчейн технологии, УИИ разрабатывает механизм, который позволяет студентам создавать и поддерживать цифровые портфолио, содержащие информацию о выполненных проектах, полученных оценках, участии в исследовательских работах и других достижениях. Эти данные сохраняются в блокчейн, что гарантирует их целостность, неизменность и доступность для всех заинтересованных сторон.

Заказчики, такие как потенциальные работодатели или образовательные учреждения, могут использовать блокчейн для верификации информации в портфолио студента [4, 12]. Это обеспечивает надежность и достоверность данных, предоставляемых студентом, и упрощает процесс оценки качества его образования и навыков.

Таким образом, УИИ стремится не только повысить безопасность идентификации, но и реализовать инновационный подход к формированию и представлению портфолио студентов с использованием блокчейн технологии. Это позволяет учебному учреждению и его студентам создать доверенное и прозрачное

окружение для обмена информацией о достижениях и компетенциях, способствуя тем самым улучшению процессов проверки и оценки квалификаций студентов.

1.2 Организационная структура

1.2.1 Иерархическая структура

Иерархическая структура УИИ представляет собой организационную структуру, отражающую уровни управленческой и функциональной иерархии внутри учебного учреждения. Эта структура спроектирована с учетом оптимального распределения обязанностей и эффективного взаимодействия между различными уровнями административной и образовательной деятельности (Рис 1.1 — Организационная структура УИИ).

Рис. 1.1 Организационная структура УИИ

На самом высоком уровне иерархии функционирует административное руководство, в лице ректора, который осуществляет общий контроль над стратегическим развитием университета, принимает важные стратегические решения и координирует работу всех подразделений. Помимо этого, координирует профессиональные советы, включая управленческий совет, который играет ведущую роль в формировании стратегических направлений развития университета.

На следующем уровне иерархии находятся факультеты УИИ, каждый из которых специализируется в конкретной области искусственного интеллекта. Руководители факультетов отвечают за академические программы, исследования и обучение в своей области.

Далее идет уровень кафедр и подразделений, ответственных за конкретные дисциплины и направления обучения. Преподаватели и исследователи на этих уровнях занимаются формированием и проведением учебных курсов, научными исследованиями и поддержкой студентов.

Важным элементом иерархической структуры УИИ являются административные и образовательные подразделения, ответственные за организацию учебного процесса, ведение финансовых операций, административные вопросы, информационные технологии и другие функциональные области.

Эта иерархическая структура позволяет УИИ эффективно координировать деятельность различных частей университета, обеспечивая интеграцию административных, образовательных и научных аспектов для достижения общих стратегических целей и успешной реализации миссии университета в области искусственного интеллекта.

1.2.2 Роль административных и образовательных подразделений в системе идентификации

Административные подразделения осуществляют управление и контроль за инфраструктурой системы идентификации. Они занимаются созданием и обслуживанием необходимых технических компонентов, таких как серверы, базы данных и программное обеспечение, гарантируя их бесперебойную работу и защиту от несанкционированного доступа.

Образовательные подразделения внедряют и поддерживают технологии идентификации, такие как смарт—карты, биометрические системы или блокчейн технологии. Они также ответственны за разработку и поддержание процессов аутентификации студентов при доступе к онлайн—ресурсам и учебным материалам, обеспечивая удобство использования и высокий уровень безопасности.

Взаимодействие административных и образовательных подразделений направлено на создание интегрированной и безопасной системы идентификации, обеспечивающей надежность в учебной среде. Этот совместный подход позволяет университету эффективно реализовывать стратегии по улучшению безопасности идентификационных процессов в современных условиях образования. Совместное воздействие административных и образовательных структур нацелено на формирование эффективной и надежной системы идентификации, обеспечивающей безопасность данных и обеспечивающей удобство пользования для пользователей в контексте образовательной среды. Этот симбиоз усилий важен для успешного достижения целей по обеспечению безопасности идентификационных процессов в современном университетском обучении.

1.3 Современные тенденции образования

Современное образование, к которому стремится УИИ, представляет собой динамичную область, претерпевающую непрерывные трансформации под воздействием технологических инноваций и изменений [2, 3], для удовлетворения потребностей общества. Рассмотрим тенденции, оказывающие влияние на развитие современного образования:

1. Гибкость и доступность — с появлением образовательных онлайн платформ, студенты получают гибкость выбора учебных материалов и графика обучения, что соответствует их индивидуальным потребностям и обстоятельствам. Это также позволяет расширить географические границы образования, обеспечивая доступность для студентов из различных регионов мира.

2. Интерактивность и адаптивность — современные образовательные платформы интегрируют технологии, обеспечивающие интерактивное обучение и адаптивные методы преподавания. Это включает в себя использование виртуальной и дополненной реальности, а также алгоритмическое адаптивное обучение для учета индивидуальных уровней знаний студентов.

3. Системы управления обучением (LMS) — интеграция Learning Management Systems (LMS) становится неотъемлемой частью online образования

[4, 8, 12]. Они обеспечивают централизованное управление учебными материалами, административными процессами, и отслеживание активности студентов, обеспечивая эффективность и контроль.

4. Массовые открытые онлайн курсы — предоставляют масштабные возможности обучения, объединяя тысячи студентов со всего мира. Они предлагают бесплатный доступ к курсам от ведущих университетов и организаций, стимулируя глобальное образовательное сообщество.

5. Эволюция оценивания и сертификации — современные тенденции включают пересмотр традиционных методов оценивания в пользу более интерактивных и адаптивных форм, таких как проектные работы, онлайн тестирование, и практические задания. Это также включает в себя эмиссию электронных сертификатов и акцент на признании полученных навыков.

6. Обеспечение безопасности и прозрачности — с ростом онлайн образования возрастает необходимость в обеспечении безопасности данных студентов и прозрачности учебного процесса. Технологии, такие как блокчейн, предоставляют новые возможности для решения этих проблем [8].

Все перечисленные тенденции объединяются в стремлении создать более доступную, интерактивную и эффективную образовательную среду, которая соответствует современным потребностям обучающихся и преподавателей.

Стоит отметить, что формирование портфолио студента представляет собой не мало важный аспект современного образования, играющий значительную роль в процессе оценки, развития и демонстрации достижений учащихся. Рассмотрим фундаментальные аспекты важности формирования портфолио в образовательной среде:

1. Документация академического прогресса — портфолио служит многогранным инструментом для документирования академического прогресса студента. Оно включает в себя различные типы работ, проектов, рефератов, и других академических достижений, предоставляя исчерпывающий обзор интеллектуального и творческого развития.

2. Индивидуализация обучения — формирование портфолио позволяет студентам индивидуализировать свой учебный опыт, выделяться и демонстрировать уникальные навыки, интересы и достижения. Это способствует более глубокому и осмысленному обучению, а также помогает преподавателям лучше адаптировать методы обучения под потребности студентов.

3. Продвижение саморефлексии — формирование портфолио стимулирует саморефлексию и самоанализ. Студенты, составляя свое портфолио, вынуждены критически оценивать свои работы, выявлять сильные стороны и области для улучшения, что способствует развитию метапознания и саморегуляции.

4. Поддержка карьерного роста — портфолио студента является ценным инструментом при поиске работы или поступлении в учебные заведения. Работодатели и вузы получают более полное представление о компетенциях, умениях и опыте кандидата, что повышает шансы на успешное трудоустройство или поступление.

5. Поддержка оценочных процессов — в учебном процессе портфолио играет важную роль в оценке студенческих достижений. Преподаватели могут использовать портфолио для более глубокой и дифференцированной оценки, а также для мониторинга прогресса и адаптации методов обучения.

6. Развитие критического мышления — составление портфолио требует анализа, оценки и выборочного представления материалов. Этот процесс способствует развитию критического мышления, способности выделять значимые аспекты и аргументированно представлять свои идеи.

Перечисленные аспекты, особенно подчеркивают, неотъемлемую важность формирования портфолио в современной образовательной практике, которое является не только инструментом документирования академических достижений, но и ключевым фактором в процессе индивидуализации обучения, стимулируя развитие саморефлексии, критического мышления и уникальных навыков студентов. Более того, портфолио несет в себе потенциал существенного вклада в карьерный рост выпускников, предоставляя работодателям более пол-

ное представление об их компетенциях и опыте. Оценочные процессы становятся более глубокими и дифференцированными, а сам процесс создания портфолио способствует развитию самоанализа и саморегуляции.

Таким образом, формирование портфолио не только улучшает оценку студентских достижений, но и способствует их личностному развитию, что делает его важным инструментом как для учебного процесса, так и для успешной адаптации в профессиональной сфере.

1.4 Анализ текущих бизнес процессов

Анализ текущих бизнес—процессов в УИИ необходим для понимания текущего состояния учебного учреждения и выявления аспектов, подлежащих автоматизации, при внедрении системы на основе блокчейн. На данном этапе оцениваются основные этапы жизненного цикла образовательного процесса: от регистрации студентов до завершения учебных курсов.

1.4.1 Описание жизненного цикла образовательного процесса: от регистрации до завершения курса

Жизненный цикл образовательного процесса в УИИ представляет собой комплексный и много векторный путь, начиная с момента регистрации студентов и завершаясь успешным окончанием учебного курса.

1. Регистрация студентов — процесс начинается с регистрации абитуриентов, которая включает сбор и верификацию их личных данных, предоставление необходимых документов, и создание уникальной учетной записи для каждого студента. Важным этапом является обеспечение безопасности и конфиденциальности передаваемой информации.

2. Выбор учебных программ — после завершения регистрации студенты выбирают учебные программы и курсы в соответствии с их профессиональными интересами. Важно, чтобы процесс выбора был удобным и доступным для студентов, а информация о курсах полной и понятной.

3. Регистрация на курсы и оплата обучения — студенты регистрируются на выбранные курсы и осуществляют оплату. Этот этап включает в себя проверку статуса оплаты и предоставление студентам доступа к необходимым учебным материалам и ресурсам.

4. Учебный процесс — студенты активно участвуют в учебных занятиях, лекциях, практических занятиях и проектах. Процессы фиксации промежуточных и итоговых результатов, а также ведение учебных записей являются важными аспектами, отслеживаемыми на данном этапе.

5. Завершение курса — после прохождения обучения происходит итоговая аттестация и завершается учебный курс. Завершение курса включает в себя аутентификацию достижений студента, выдачу документа об образовании, а также архивацию учебных данных.

Описание жизненного цикла образовательного процесса от регистрации до завершения курса (Рис 1.2 — Схема as—is образовательного процесса) позволяет выявить моменты, в которых цифровая идентификация на блокчейне может существенно улучшить процессы, обеспечив безопасность, прозрачность и эффективность учебного процесса в УИИ.

1.4.2 Недостатки текущей системы идентификации

В рамках анализа текущей системы идентификации в УИИ выявляются недостатки и неэффективности, которые могут подвергнуть угрозе безопасность, прозрачность и эффективность учебного процесса. Исследование эффективности текущей системы идентификации выделяет следующие аспекты:

1. Недостаточная защита личных данных — текущая система идентификации может подвергать личные данные студентов риску утечки или несанкционированного доступа. Отсутствие надежных методов шифрования и аутентификации может привести к возможным нарушениям конфиденциальности [1, 9].

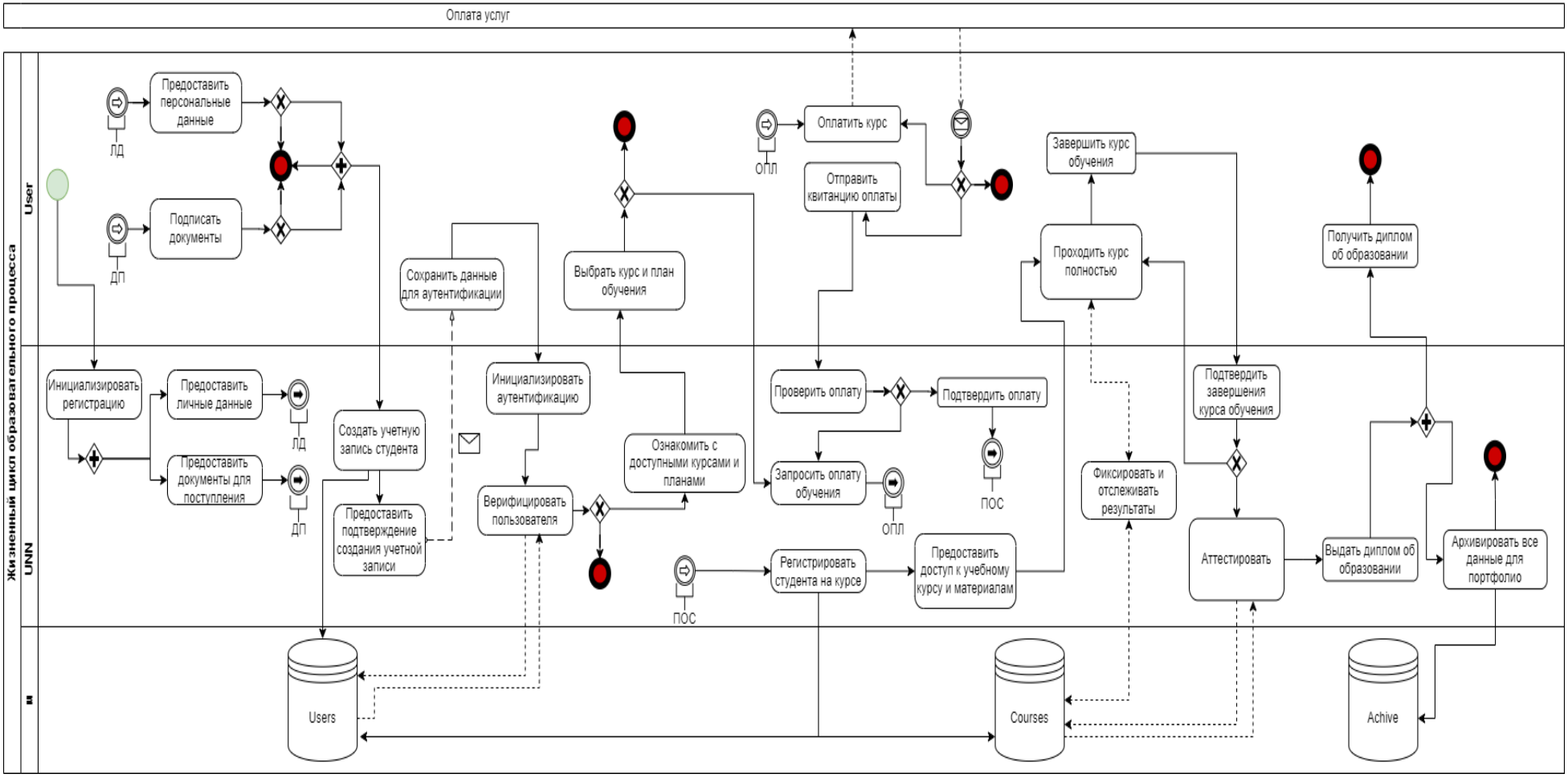


Рис. 1.2 — Схема as—is образовательного процесса

2. Отсутствие прозрачности в управлении доступом — текущие механизмы управления доступом могут быть недостаточно прозрачными, что создает потенциальные уязвимости. Сложность в контроле и мониторинге доступа может привести к несанкционированным действиям и угрозам безопасности [9].

3. Манипуляции с учетными записями — возможность манипулировать учетными записями студентов, такими как создание фальшивых профилей или изменение данных, может подорвать достоверность идентификации студентов.

4. Отсутствие стойкости к мошенничеству — текущая система не обеспечивает достаточной стойкости к мошенничеству, что может отразиться на процессе аттестации и формирования академических достижений [1, 9].

5. Ограниченные возможности восстановления данных — недостаточные механизмы восстановления данных и идентификации могут замедлять процессы реагирования на угрозы безопасности и восстановления после инцидентов.

6. Персонализация портфолио — текущая система идентификации может ограничивать возможности персонализации портфолио студентов. Отсутствие гибкости и настройки в соответствии с индивидуальными потребностями и целями студентов, заказчиков и работодателей, что осложняет процедуру подтверждения квалификации.

Исследование неэффективности текущей системы идентификации в УИИ позволяет выявить необходимость внедрения новых технологий, таких как цифровая идентификация на основе блокчейна. Это обеспечит не только повышение уровня безопасности, но и содействие прозрачности и эффективности учебного процесса в целом.

1.4.3 Ключевые точки взаимодействия с цифровой идентификацией

Взаимодействие направлено на выявление моментов, где внедрение цифровой идентификации на основе блокчейна может эффективно автоматизировать и усовершенствовать текущие бизнес—процессы.

1. Регистрация студентов — цифровая идентификация может быть интегрирована на этапе регистрации студентов, обеспечивая более безопасный и эффективный сбор, и верификацию личных данных. Использование уникальных идентификаторов на блокчейне позволит предотвратить возможные манипуляции с данными и обеспечит стойкость к несанкционированным вмешательствам.

2. Доступ к учебным ресурсам — идентификация с использованием технологии блокчейн может обеспечить безопасный и прозрачный доступ студентов к учебным материалам. Смарт контракты могут регулировать процесс предоставления доступа, гарантируя только авторизованным пользователям возможность взаимодействия с онлайн—ресурсами [10, 13].

3. Система аттестации — внедрение цифровой идентификации в систему аттестации позволяет более надежно идентифицировать студентов и формировать академические достижения. Это создает прозрачность и предотвращает возможные случаи мошенничества.

4. Завершение учебного курса — цифровая идентификация может быть важнейшим фактором при завершении учебного курса, где блокчейн технологии используются для создания надежных и подлинных документов об образовании. Это гарантирует легитимность достижений студента и обеспечивает дополнительный уровень безопасности в процессе выдачи дипломов и формировании персонального портфолио для заинтересованных лиц.

Ключевые точки взаимодействия с цифровой идентификацией подчеркивают потенциал блокчейн технологий для улучшения учебного процесса в УИИ. Автоматизация этих моментов позволит повысить уровень безопасности, прозрачности и эффективности учебного учреждения (Рис 1.3 — Схема to—be образовательного процесса).

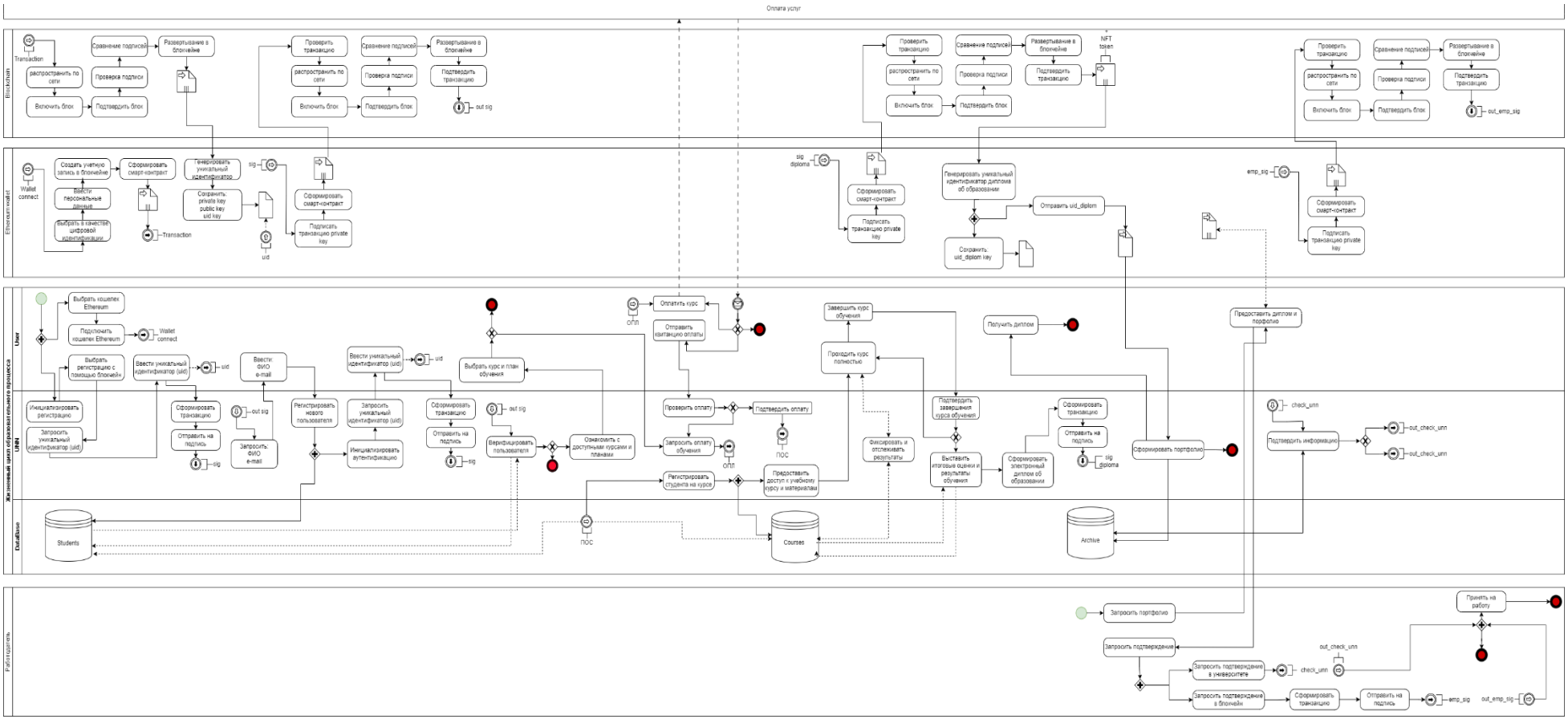


Рис. 1.2 Схема to—be образовательного процесса

1.5 Теоретическая часть

1.5.3 Краткое сравнение существующих подходов построения защищенных систем

В настоящее время существует несколько подходов к решению вопросов идентификации и аутентификации в образовательных средах, таких как использование сервисов, предоставляемых внешними поставщиками (Google, Facebook, Twitter, Yandex, VKontakte), а также технологии блокчейн.

Сервисы внешних поставщиков, такие как Google, Facebook и другие, предлагают удобные механизмы для идентификации и аутентификации пользователей. Однако, они имеют свои ограничения. Во—первых, это централизованные системы, что подразумевает высокий уровень зависимости от одного поставщика. Во—вторых, существует риск утечки личных данных, так как эти сервисы хранят значительное количество конфиденциальной информации.

Технология блокчейн предоставляет децентрализованный и безопасный способ идентификации и аутентификации [14]. Он основан на принципе распределенного хранения данных, что делает его устойчивым к централизованным атакам и обеспечивает высокий уровень прозрачности. Также блокчейн предоставляет пользователям больший контроль над своей личной информацией, поскольку она хранится децентрализовано и защищена криптографически.

Преимущества и недостатки:

- Преимущества сервисов внешних поставщиков — удобство в использовании, быстрая реализация, широкая распространенность.
- Недостатки сервисов внешних поставщиков — централизованный подход, угроза конфиденциальности данных, зависимость от сторонних поставщиков.
- Преимущества блокчейн — децентрализация, высокий уровень безопасности, прозрачность, возможность пользовательского контроля.
- Недостатки блокчейн — более сложная реализация, ограниченная скорость транзакций по сравнению с централизованными системами.

Таким образом, блокчейн предоставляет более безопасный и надежный механизм идентификации и аутентификации в образовательных средах, устраняя ряд проблем, связанных с централизованными системами внешних поставщиков.

1.5.4 Роль блокчейн технологии в усилении безопасности и идентификации

Блокчейн технология представляет собой инновационный подход к обеспечению безопасности и идентификации в образовательных средах. Основываясь на децентрализованной структуре и использовании криптографии, блокчейн вносит значительный вклад в повышение уровня безопасности и подлинности данных в учебных учреждениях [5, 14].

Децентрализация и надежность — является одной из самых важных особенностей блокчейн технологии. Вместо хранения информации в централизованной базе данных, блокчейн распределяет ее по всей сети, что делает систему менее уязвимой к атакам. Каждый блок данных связан с предыдущим с использованием криптографических хэшей, что обеспечивает непрерывную цепь блоков, стойкую к манипуляциям.

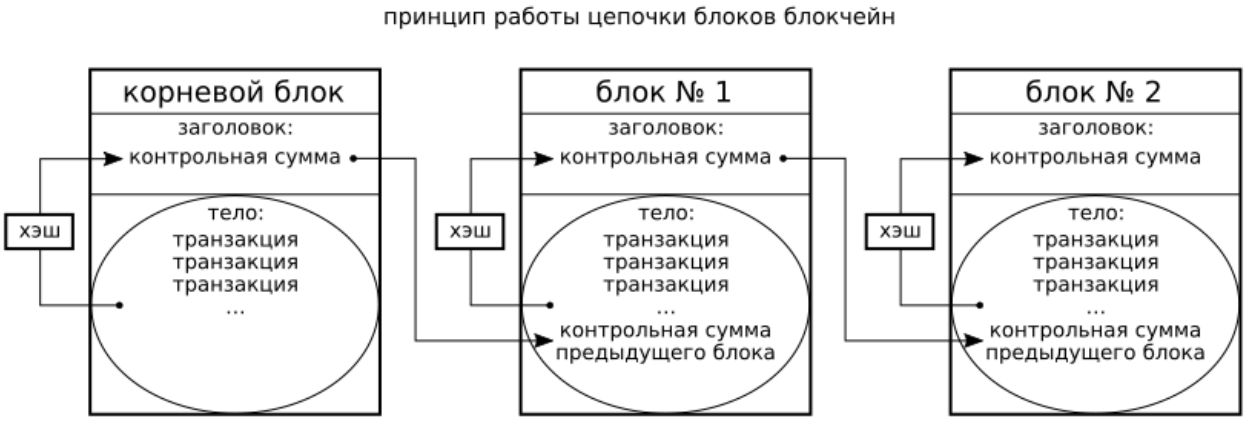


Рис. 1.4 Принцип построения блоков в сети блокчейн

Криптография для защиты идентификационных данных — блокчейн использует современные методы криптографии для защиты идентификационных данных. Каждая транзакция подписывается цифровой подписью, обеспечивая

аутентификацию и авторизацию участников системы. Это гарантирует, что идентификационные данные студентов и других участников остаются защищенными от несанкционированного доступа и фальсификации.

Прозрачность и невозможность подделки — благодаря прозрачности и неизменяемости данных, блокчейн обеспечивает невозможность подделки истории транзакций. Каждый участник сети имеет доступ к одному общему распределенному реестру, и любые изменения данных требуют согласия большинства участников. Это делает блокчейн особенно эффективным в предотвращении поддельных идентификационных данных и обеспечении подлинности информации.

Улучшенная идентификация и трассируемость — блокчейн технология позволяет создавать уникальные идентификаторы (UID) для всех участников образовательного процесса. UID записанные в блокчейн, обеспечивают точную идентификацию личности, а также обеспечивают трассируемость всех связанных с ней транзакций и активностей.

Эффективное управление доступом — блокчейн предоставляет механизм эффективного управления доступом, разграничивая права и предоставляя доступ только тем, кому это необходимо. Смарт контракты, реализованные в блокчейне, могут автоматизировать процессы аутентификации и управления правами доступа, уменьшая риски человеческого фактора [7].

Обобщая, можно сказать, что блокчейн технология играет большую роль в усилении безопасности и идентификации в образовательных учреждениях, предоставляя современные инструменты для эффективного управления данными и обеспечения надежности идентификационной информации.

1.5.5 Особенности Ethereum блокчейна и их применение в системах идентификации

1. Смарт контракты — одной из важнейших особенностей Ethereum блокчейна являются смарт контракты [7]. Это программные коды, которые автоматизируют и обеспечивают выполнение условий без необходимости посредников. В системах идентификации, смарт контракты могут использоваться для создания и

управления цифровыми идентификационными атрибутами. Например, они позволяют эффективно управлять правами доступа и автоматизировать процессы аутентификации.

2. Децентрализованная структура — Ethereum работает на принципах децентрализации, что означает, что нет центрального управляющего органа [10, 11]. Это способствует более высокой стойкости к атакам и обеспечивает более надежную систему идентификации. Децентрализованная структура также содействует обеспечению конфиденциальности данных, поскольку они хранятся на различных узлах сети.

3. Эффективные транзакции — Ethereum блокчейн предлагает быстрые и относительно недорогие транзакции. Это важно для систем идентификации, так как обеспечивает оперативную и эффективную проверку личности. Эффективность транзакций также способствует созданию эффективных систем управления доступом и аутентификации.

4. Криптографическая безопасность — Ethereum блокчейн использует криптографические методы для обеспечения безопасности данных. Хэширование и цифровые подписи применяются для гарантии подлинности и целостности идентификационной информации. Это обеспечивает высокий уровень защиты от манипуляций и несанкционированного доступа [8].

5. Разнообразные токены и стандарты — Ethereum поддерживает стандарты токенов, такие как ERC20 и ERC721 [5, 13], что позволяет создавать уникальные цифровые активы и идентификационные метки. Это расширяет возможности внедрения индивидуальных идентификационных средств, таких как уникальные номера студенческих карт, которые могут быть эффективно отслеживаемы в блокчейне.

6. Открытость и транспарентность — Ethereum блокчейн обеспечивает открытость и прозрачность транзакций. Это важно для систем идентификации в образовательных учреждениях, поскольку стейкхолдеры могут проверять и подтверждать легитимность идентификационных данных [14]. Открытость также способствует доверию участников к системе идентификации.

7. Расширенные возможности программирования — Ethereum блокчейн обладает мощными возможностями программирования, что позволяет создавать сложные сценарии управления идентификацией. Это включает в себя настройку процессов автоматической регистрации, обновления данных и отзыва прав доступа [17].

Таблица 1.1

Блокчейн технологии в контексте систем идентификации

Характеристика	Bitcoin Blockchain	Hyperledger Fabric	Ethereum Blockchain
Тип Блокчейна	Публичный	Разрешенный (Permissioned)	Публичный
Смарт контракты	Отсутствуют	Поддерживаются	Поддерживаются
Скорость Транзакций	Низкая	Высокая	Высокая
Комиссии за Транзакции	Высокие	Зависят от настроек	Умеренные
Приватность Данных	Открытая	Защищенная	Открытая
Токены и Стандарты	Ограничены (Bitcoin)	Поддерживаются	Широкий спектр (ERC—20, ERC—721)
Децентрализация	Высокая	Высокая	Высокая
Программируемость	Ограниченная	Высокая	Высокая
Применение в Системах Идентификации	Ограничено	Эффективное	Идеальное

Применение Ethereum блокчейна в системах идентификации обеспечивает безопасное, эффективное и децентрализованное управление данными (Таблица 1.1 Блокчейн технологии в контексте систем идентификации). Смарт контракты помогают автоматизировать процессы идентификации, а децентрализация и криптография обеспечивают высокий уровень безопасности. Быстрые и недоро-

гие транзакции, в сочетании с возможностью создания уникальных токенов, делают Ethereum блокчейн идеальным выбором для современных систем идентификации в образовательных учреждениях.

1.5.6 Требования к системе цифровой идентификации

Цифровая идентификация в сфере образования предполагает высокие стандарты надежности, безопасности и функциональности системы. В контексте университетской образовательной платформы, базирующейся на блокчейне Ethereum, следует выделить ряд важных требований:

- Надежность и безопасность — обеспечение высокого уровня защиты от несанкционированного доступа и утечки конфиденциальных данных.
- Децентрализация — реализация децентрализованных принципов для обеспечения стойкости к атакам и улучшения устойчивости системы.
- Гибкость и масштабируемость — создание системы, способной адаптироваться к изменениям в образовательном процессе и эффективно масштабироваться под увеличение числа пользователей.
- Совместимость и интеграция — гарантирование беспроблемной интеграции с существующими информационными системами университета.
- Аутентификация и авторизация — предоставление эффективных механизмов аутентификации, использующих блокчейн для обеспечения подлинности и строгой системы авторизации.
- Прозрачность и открытость — обеспечение открытости данных и прозрачности процессов с использованием блокчейн—технологии.
- Эффективность транзакций — обеспечение быстрой и эффективной обработки транзакций для оперативной идентификации пользователей.
- Поддержка смарт контрактов — для автоматизации и эффективного управления правами доступа.
- Удобство пользователя — предоставление удобных и интуитивно понятных средств взаимодействия для повышения комфорта использования системы.

- Соответствие нормативам и законам — соблюдение законодательных норм и регуляций в области защиты данных.
- Адаптивность к технологическим изменениям — спроектированная система должна быть адаптивной к технологическим изменениям, обеспечивая долгосрочную устойчивость и актуальность.

Реализация требований внедрит в образовательную среду, эффективные методы управления личной идентификацией студентов и персонала, соответствуя современным стандартам безопасности и функциональности.

1.5.7 Учет особенностей учебного процесса при определении требований

При разработке системы цифровой идентификации для образовательной платформы, основанной на блокчейне Ethereum, необходимо учитывать уникальные особенности учебного процесса, чтобы обеспечить максимальную эффективность и соответствие потребностям пользователей, выделяя следующие аспекты:

- Разнообразие учебных курсов — УИИ предоставляет широкий спектр учебных курсов, варьирующихся по тематике, сложности и формату. Требования к системе идентификации должны быть гибкими, чтобы адаптироваться к различным потребностям учебных программ.
- Многоуровневая система обучения — учебный процесс охватывает несколько уровней образования, от junior до senior. Требования к идентификации должны поддерживать многоуровневую структуру, обеспечивая точность идентификации на всех уровнях обучения.
- Индивидуализированный подход к обучению — современные образовательные практики подчеркивают важность индивидуализированного обучения. Система идентификации должна поддерживать формирование персонализированных портфолио и обеспечивать точную идентификацию студентов в рамках их индивидуальных учебных планов.

- Эффективный мониторинг прогресса — учет особенностей учебного процесса включает в себя разработку механизмов для эффективного мониторинга академического прогресса студентов. Система должна поддерживать сбор и анализ данных о достижениях студентов, предоставляя необходимую информацию для адекватной оценки успеваемости.
- Безопасность на этапе аттестации — отмечается, что аттестация студентов является ключевым этапом учебного процесса. Система цифровой идентификации должна гарантировать безопасность и подлинность данных при проведении оценочных мероприятий.
- Интеграция с учебными ресурсами — учебные материалы и ресурсы предоставляются в различных форматах и платформах. Требования к идентификации должны предусматривать интеграцию с разнообразными образовательными ресурсами для обеспечения удобства доступа студентов.
- Соблюдение этических и законных норм — в процессе определения требований к системе идентификации, особое внимание уделяется соблюдению этических принципов и соблюдению законодательства о защите данных студентов.

В заключении, результат анализа учета особенностей учебного процесса при определении требований к системе цифровой идентификации для образовательной платформы УИИ, можно сделать несколько выводов.

Прежде всего, необходимо признать, что учебный процесс характеризуется высокой степенью разнообразия и динамичности, требуя гибких и адаптивных решений. Важность индивидуализированного обучения подчеркивает необходимость персонализированных подходов к идентификации студентов и формированию их портфолио.

Эффективность системы идентификации тесно связана с ее способностью интеграции с разнообразными учебными ресурсами, обеспечивая удобный доступ к информации и материалам для студентов и преподавателей [3, 4, 8]. Гарантирование безопасности данных важно на каждом этапе учебного процесса, особенно при проведении оценочных мероприятий.

Особое внимание уделено соблюдению этических норм и законодательства о защите данных, что подчеркивает не только технологическую, но и этическую ответственность при разработке системы цифровой идентификации пользователей и объектов [9].

Таким образом, требования к системе цифровой идентификации для УИИ, должны учитывать множество аспектов, обеспечивая не только функциональную эффективность, но и соответствие уникальным особенностям учебного процесса для создания оптимальной и инновационной системы в образовательной среде.

1.6 Заключение

Глава 1 представляет собой тщательный анализ УИИ, охватывающий как общие сведения об университете, так и детальный разбор его организационной структуры и текущих бизнес—процессов.

Первоначально проведен анализ общих сведений об университете, что включало основные характеристики, направления обучения и основные цели. Этот этап обеспечил контекст для последующего исследования.

Организационная структура университета была подвергнута подробному рассмотрению, включая иерархическую структуру, роль административных и образовательных подразделений в системе идентификации. Это аналитическое изучение позволило выделить ключевые элементы управления и взаимодействия внутри структуры УИИ.

Далее был выполнен анализ образовательного процесса, который выявил важные этапы взаимодействия с цифровой идентификацией. Также выделены ключевые точки взаимодействия, определяющие успешное функционирование системы идентификации УИИ. Обнаружены проблемы в текущей системе идентификации, проведено исследование их неэффективности, что обосновывает необходимость внедрения новой системы.

Особый акцент сделан на роли блокчейн технологии в усилении безопасности и идентификации. Проанализированы особенности Ethereum блокчейна и

его применение в системах идентификации. Сформулированы требования к системе цифровой идентификации, учитывая особенности учебного процесса.

В целом, первая глава представляет комплексный обзор организации, ее бизнес—процессов и теоретических основ, что служит основой для разработки эффективной системы цифровой идентификации в учебном учреждении.

Подробности составленного технического задания, смотреть в Приложении А.

2.РАЗРАБОТКА СИСТЕМЫ ЦИФРОВОЙ ИДЕНТИФИКАЦИИ

2.1 Проектирование архитектуры

2.1.1 Обоснование выбора технологий и алгоритмов

При проектировании системы цифровой идентификации для учебной платформы УИИ на базе блокчейн технологии Ethereum, были выбраны определенные технологии и алгоритмы, обоснование которых представлено ниже.

1. Смарт контракты (играют ключевую роль в автоматизации и безопасности процессов) — представляют собой программные сущности, выполнение которых автоматически происходит в блокчейне Ethereum, при наступлении определенных условий. Основная цель смарт контрактов, заключается в обеспечении прозрачности, непротиворечивости и безопасности всех этапов цифровой идентификации студентов [5, 7]. Они позволяют автоматизировать процессы регистрации, аутентификации и учета активности студентов, исключая возможность вмешательства третьих сторон.

Технически, они реализованы с использованием языка программирования Solidity, специально предназначенного для написания на платформе Ethereum и содержат логику, определяющую условия выполнения, а также взаимодействие с другими контрактами и данными в блокчейне (Рисунок 2.1 — Смарт контракт верификации).

Одним из ключевых аспектов, является их параметризация (Рис 2.2 — Базовый принцип работы смарт контрактов). Это означает, что они могут быть настроены в соответствии с конкретными требованиями, каждого курса или вида активности студента. Такая гибкость обеспечивает адаптивность системы и возможность изменения правил идентификации в соответствии с потребностями УИИ. Они также интегрируются с кошельком Metamask (или любым другим, работающим с Ethereum) для безопасной аутентификации и подписи транзакций.

Это обеспечивает защиту личных данных студентов и подтверждение подлинности транзакций в системе.

Таким образом, смарт контракты в системе цифровой идентификации для УИИ, представляют собой универсальный инструмент, обеспечивающий эффективность, безопасность и прозрачность, в процессах идентификации студентов на основе блокчейн технологии Ethereum.

```solidity
// Указываем версию компилятора Solidity
pragma solidity ^0.8.0;

// Определяем контракт
UnitTest stub | dependencies | uml | funcSigs | draw.io
contract UserVerification {

    // Структура для хранения данных подтверждения пользователя
    struct UserConfirmation {
        address userAddress;
        bool isConfirmed;
    }

    // Маппинг для связи адреса пользователя с его статусом подтверждения
    mapping(address => UserConfirmation) public confirmations;

    // Событие для уведомления о подтверждении пользователя
    event UserConfirmed(address indexed user);

    // Модификатор для проверки, что пользователь подтвержден
    modifier isConfirmed() {
        require(confirmations[msg.sender].isConfirmed, "User not confirmed");
        _;
    }

    // Функция подтверждения пользователя
    ftrace | funcSig
    function confirmUser(address userToConfirm↑) external {
        // Проверка, что подтверждаемый пользователь не является текущим пользователем
        require(userToConfirm↑ != msg.sender, "Cannot confirm yourself");

        // Проверка, что пользователь уже зарегистрирован
        require(confirmations[userToConfirm↑].userAddress == userToConfirm↑, "User not registered");

        // Проверка, что пользователь еще не подтвержден
        require(!confirmations[userToConfirm↑].isConfirmed, "User already confirmed");

        // Подтверждение пользователя
        confirmations[userToConfirm↑] = UserConfirmation(userToConfirm↑, true);

        // Инициирование события о подтверждении пользователя
        emit UserConfirmed(userToConfirm↑);
    }

    // Функция получения статуса подтверждения текущего пользователя
    ftrace | funcSig
    function getConfirmationStatus() external view isConfirmed returns (bool) {
        return confirmations[msg.sender].isConfirmed;
    }
}
```

Рис. 2.1 Смарт контракт верификации

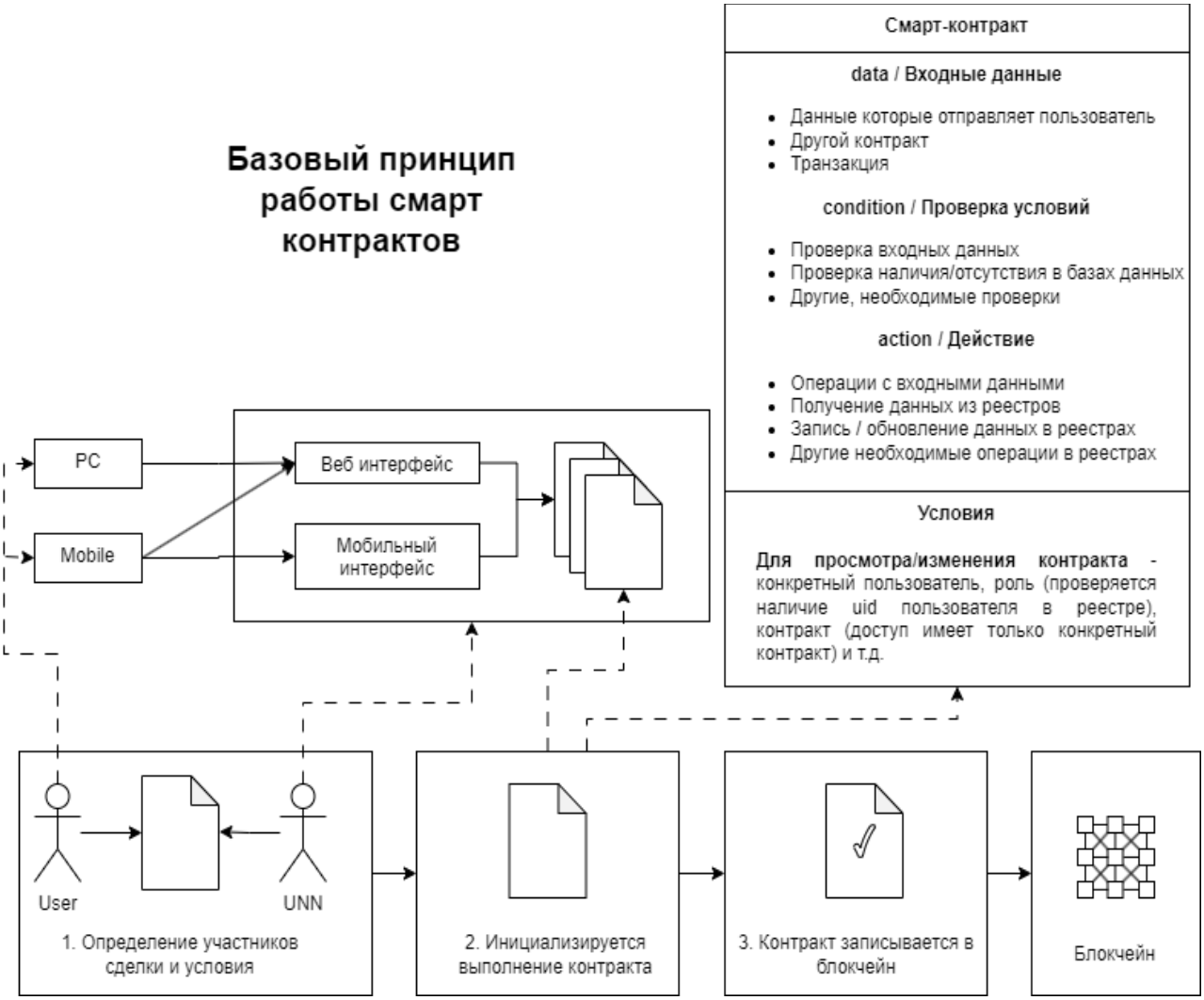


Рис. 2.2 Базовый принцип работы смарт контрактов

2. Публичная блокчейн сеть, такая как Ethereum, предоставляет уникальные преимущества для системы цифровой идентификации в образовательной среде. Её использование обеспечивает децентрализацию и прозрачность системы, что является ключевым аспектом для эффективного функционирования образовательного процесса [10, 13].

В публичном блокчейне Ethereum вся информация о студентах, курсах и сертификатах хранится в распределенном реестре (Рис 2.3 — Распределенный реестр). Это означает, что нет центрального узла управления или единой точки отказа, что повышает устойчивость и надежность системы. Данные, размещенные в блокчейне, не могут быть изменены или подделаны без согласия большинства участников сети, что гарантирует целостность и подлинность информации.

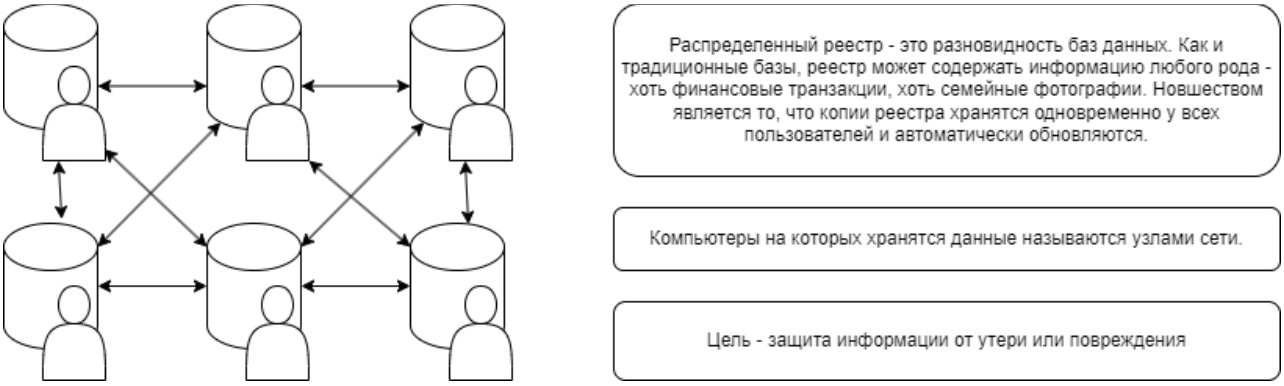


Рис. 2.3 Распределенный реестр

Децентрализация позволяет избежать зависимости от центральных управляющих органов, что особенно важно в образовательных учреждениях. УИИ может полагаться на автономные узлы блокчейна, обеспечивая тем самым управление идентификационными данными студентов без риска централизованных угроз безопасности или одностороннего контроля.

Прозрачность в таком блокчейне означает (Рис 2.4 — Принцип прозрачности блокчейна), что каждый участник сети, включая студентов, преподавателей и администраторов, имеет доступ к общей истории транзакций и идентификационных данных. Это содействует доверию и подтверждению подлинности каждого этапа образовательного процесса.

Рис. 2.4 Принцип прозрачности блокчейна

Таким образом, использование публичного блокчейна Ethereum в системе цифровой идентификации УИИ, обеспечивает не только высокий уровень безопасности и прозрачности, но и поддерживает ценности децентрализации и демократии в образовании.

3. Блокчейн стандарты, такие как ERC20 и ERC721 [5, 13], в Ethereum, играют важную роль в разработке системы УИИ. Эти стандарты предоставляют универсальные протоколы для создания токенов и управления учетными записями студентов, что обеспечивает стандартизацию и совместимость системы с другими проектами, также поддерживающими данные стандарты (Таблица 2.1 — Сводная таблица по ERC стандартам Ethereum).

Стандарт ERC20, изначально созданный для токенов, используемых в крипто валютных проектах, адаптирован для представления учетных данных студентов в виде токенов. Это обеспечивает унификацию представления информации о студентах в блокчейне и создает общий формат для обмена информации с другими образовательными учреждениями, системами.

Стандарт ERC721, известный как стандарт токенов невзаимозаменяемых активов (NFT — non—fungible token), использован для уникальных характеристик студентов, дипломов и их достижений. Например, каждый студент представлен как уникальный NFT, который хранит в себе информацию о его успехах, пройденных курсах, сертификатах, документах об образовании и участия в проектах.

Использование данных стандартов в системе цифровой идентификации УИИ предоставляет не только удобство в управлении данными, но и обеспечивает совместимость с другими блокчейн проектами, которые также придерживаются этих стандартов. Это позволяет взаимодействовать с внешними системами, обмениваться данными и предоставлять подтверждение подлинности идентификационных данных студентов.

4. Выбор языка программирования Solidity для написания смарт контрактов, аргументирован тем, что он специально разработан для работы с блокчейном Ethereum и является официальным языком программирования для этого, обеспечивая максимальную совместимость и поддержку от разработчиков специализированных приложений.

Таблица 2.1

Сводная таблица по ERC стандартам Ethereum

ERC	Название	Описание	Применение УИИ	Дата публикации	Статус	Совместимость	Содержание
ERC20	Базовый стандарт токенов	Определяет набор функций, необходимых для fungible токенов (взаимозаменяемых).	Представление учетных данных пользователей в виде токенов	19.11.2015	Final	Высокая	ФИО Электронная почта Информация о курсах Результаты аттестации
ERC721	Стандарт токенов NFT	Определяет набор функций, необходимых для NFT (невзаимозаменяемых токенов).	Хранение уникальных характеристик студентов, дипломов и их достижений	23.01.2018	Final	Средняя	Диплом Свидетельство Портфолио

Solidity — обладает синтаксисом, близким к языкам программирования, таким как JavaScript, что упрощает процесс разработки и идентификации ошибок в смарт контрактах.

5. Библиотека Web3.py для взаимодействия с блокчейном Ethereum — предоставляет API на языке программирования Python, обеспечивая удобство и гибкость при работе с контрактами и транзакциями.

6. Для хэширования применяется стандартный алгоритм, SHA256(Secure Hash Algorithm 256 bit) — является одним из наиболее распространенных и надежных алгоритмов хэширования. Он обеспечивает высокий уровень стойкости к коллизиям и обеспечивает равномерное распределение хэш—значений, при изменении входных данных (Таблица 2.2 — Свойства алгоритмов SHA).

Применение SHA256 в системе цифровой идентификации позволяет создавать уникальные хэши для каждой транзакции и блока, что обеспечивает уникальность и неизменяемость данных в блокчейне. Одним из основных применений хэширования является обеспечение целостности данных. Каждая транзакция и блок хэшируются перед добавлением в блокчейн, создавая уникальную хэш сумму, зависящую от содержания блока и любое изменение данных в блоке, приведет к изменению хэш значения, что сразу обнаруживается в блокчейне, обеспечивая непреложную подтвержденную целостность информации.

Также, хэширование применяется для подписей транзакций. Хэш создается из данных транзакции, а затем подписывается с использованием криптографического ключа. Это обеспечивает возможность верификации подлинности и неизменяемости данных, так как любое изменение данных приведет к изменению хэша, что сразу станет заметным в блокчейне.

Эти технологические решения были выбраны с учетом их совместимости, надежности, стандартизации и широкой поддержки. Такое обоснованное решение создает прочную и эффективную архитектуру системы на блокчейне Ethereum.

Таблица 2.2

Свойства алгоритмов SHA

Алгоритм	Длина дайджеста сообщения (бит)	Длинна внутреннего состояния (бит)	Длина блока (бит)	Длина сообщения (бит)	Длина слова (бит)	Количество итераций в цикле
SHA1	160	160	512	$< 2^{64}$	32	80
SHA224	224	256	512	$< 2^{64}$	32	64
SHA256	256	256	512	$< 2^{64}$	32	64
SHA384	384	512	1024	$< 2^{128}$	64	80
SHA512	512	512	1024	$< 2^{128}$	64	80

2.1.3 Описание структуры системы идентификации

Проектируемая структура системы представляет собой комплексное решение, направленное на обеспечение безопасности, прозрачности и эффективности идентификации.

Основными элементами системы являются — смарт контракты, база данных и взаимодействие с кошельком Metamask (Рис 2.5 — Подключение к тестовой странице) или любым другим, связанным с блокчейном Ethereum.

Смарт контракты используются для автоматизации регистрации и идентификации студентов, управления доступом и записи активности в блокчейн.

Блокчейн служит основным источником хранения идентификационных данных, интегрируется с учебной платформой и обеспечивает прозрачность и целостность данных. Взаимодействие с кошельком реализуется для безопасной регистрации и подписи транзакций.

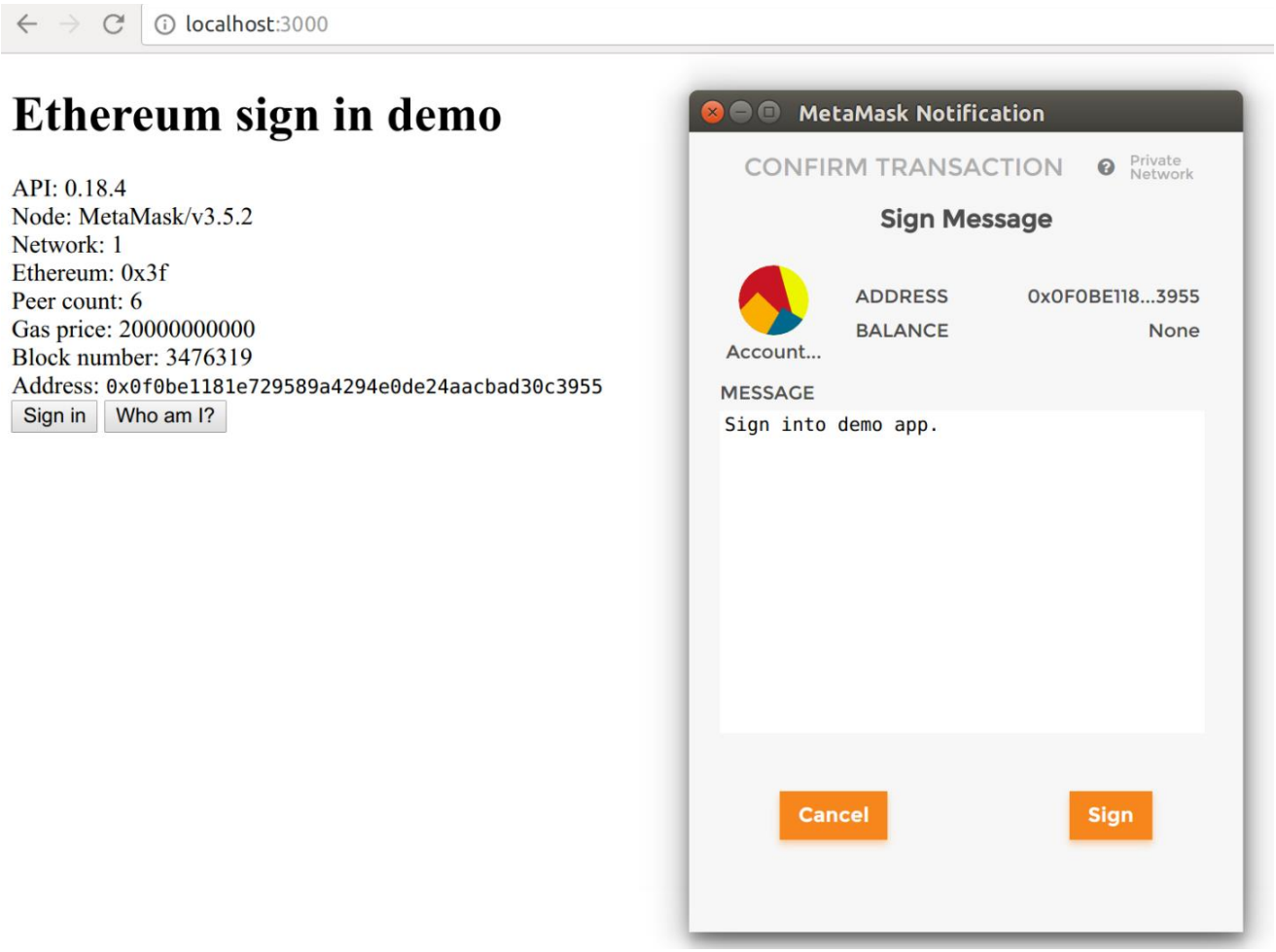


Рис. 2.5 Подключение к тестовой странице

Структура системы включает в себя интерфейс пользователя, предоставляющий web интерфейс (Рис 2.6 — Проверка на тестовой площадке) для взаимодействия пользователей с системой. Этот интерфейс интегрируется с основным web интерфейсом учебной платформы УИИ, обеспечивая единое пространство для пользователей, что предоставляет гибкое, надежное и безопасное решение для идентификации пользователей, поддерживая прозрачность данных и обеспечивая единый пользовательский опыт.

Выбор технологий и алгоритмов в разработке системы цифровой идентификации для УИИ осуществлялся с целью обеспечения высокой степени безопасности, эффективности и удобства использования. Кроме основных компонентов, таких как блокчейн Ethereum, кошелька, смарт контрактов, базы данных и интерфейса пользователя, рассмотрим дополнительные аспекты:

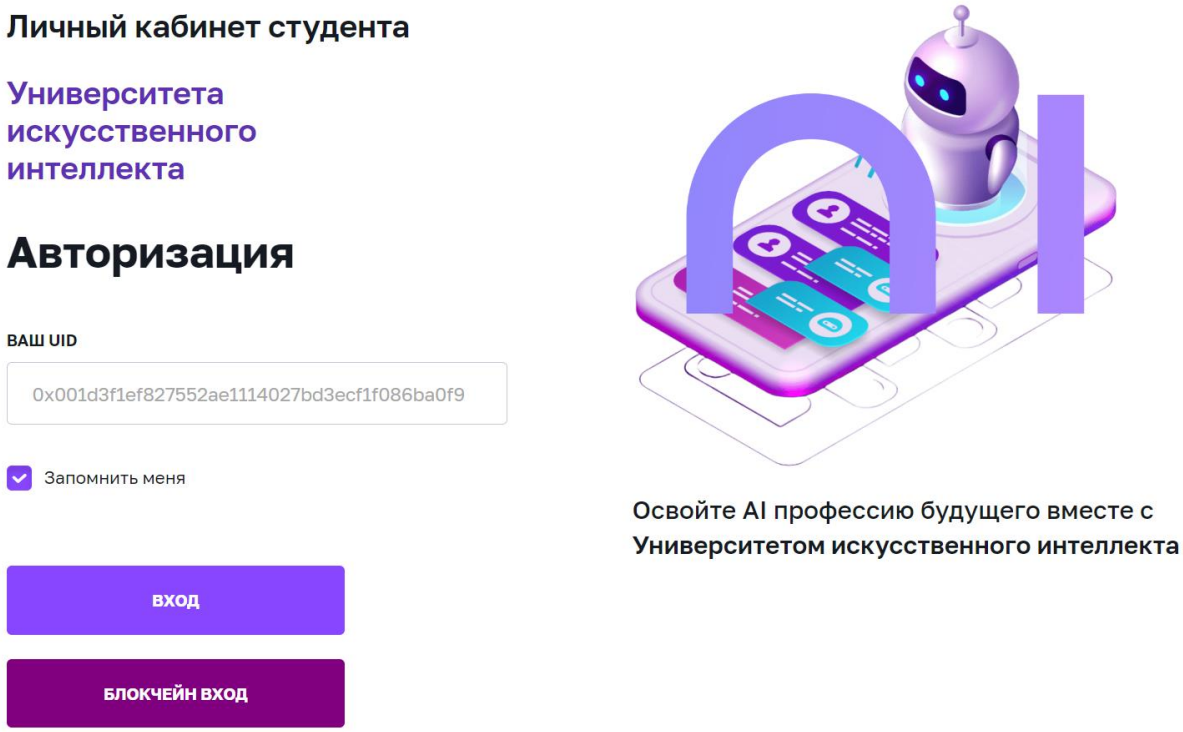


Рис. 2.6 Проверка на тестовой площадке

1. Масштабируемость системы — учитывая возможное увеличение числа студентов и курсов на платформе, выбранные технологии должны обеспечивать масштабируемость системы. Ethereum, с его гибкой архитектурой, обеспечивает возможность расширения без потери производительности.

2. Гибкость в настройке смарт контрактов — предусмотреть использование параметризированных смарт контрактов, которые могут быть легко настроены в соответствии с требованиями конкретного курса или вида активности студента. Это обеспечит гибкость системы и позволит адаптировать под различные потребности.

3. Механизм обновления смарт контрактов — ввести механизмы обновления смарт контрактов для поддержания актуальности системы. Это позволит внедрить новые функциональности, улучшить безопасность и реагировать на изменения в учебной платформе и требования пользователей.

4. Механизм резервного копирования данных — разработать механизм регулярного резервного копирования данных из базы данных. Это направлено на

обеспечение устойчивости к потере данных для соблюдения требований безопасности.

5. Управление идентификационной информацией — предусмотреть механизм управления доступом к идентификационной информации пользователя. Смарт контракты регулируют, какие данные могут быть доступны для просмотра и изменения, обеспечивая приватность и безопасность.

6. Алгоритмы шифрования — для обеспечения конфиденциальности данных и безопасности транзакций в системе используются SHA256.

Таким образом, предусмотрены дополнительные аспекты, обеспечивающие гибкость, масштабируемость и безопасность разрабатываемой системы, что делает ее эффективным и надежным инструментом в образовательном процессе УИИ.

2.2 Детальная разработка

2.2.1 Разработка смарт контрактов

Разработка смарт контрактов является основным этапом в создании системы цифровой идентификации для УИИ, базирующейся на блокчейне Ethereum.

Основной момент в разработке смарт контрактов заключается в детальном анализе требований и бизнес процессов, нацеленных на решение конкретных задач. В дипломной работе, смарт контракты применяются для автоматизации процессов цифровой идентификации студентов, регистрации на курсы, выдачи сертификатов и других операций, связанных с образовательным процессом УИИ.

Безопасность является приоритетным аспектом разработки смарт контрактов, поэтому принято решение использовать стандарты безопасности Ethereum, такие как ERC20 и ERC721, потому как они используются для повышения стойкости к различным видам атак.

Каждый смарт контракт обладает определенной функциональностью, соответствующей конкретной операции с обязательным проведением тестов (модульных, функциональных). Это необходимо для обеспечения надежности работы и их успешной интеграции в систему цифровой идентификации УИИ.

Рассмотрим виды смарт контрактов, применяемых при разработке системы идентификации (Приложение Б — Код программы) в зависимости от задачи:

- Регистрация пользователя — этот смарт контракт отвечает за процесс регистрации на платформе УИИ. Включает в себя проверку идентификационных данных, UID и создание уникальных учетных записей для каждого пользователя и установку прав доступа.
- Управление токенами — обеспечивает создание и управление токенами, которые могут представлять сертификаты, достижения, документы об образовании. Он также регулирует передачу и хранение токенов в блокчейне.
- Выдача документа об образовании — отвечает за генерацию и выдачу NFT (документа об образовании) студентам, по завершении образовательных программ. Включает в себя механизмы подписи и подтверждения подлинности документа.
- Аутентификация —реализует механизм аутентификации пользователя, при входе на образовательную платформу УИИ. Обеспечивает проверку подлинности идентификационных данных и авторизацию, для доступа к персональным данным и образовательным ресурсам.
- Контроля статуса транзакции — следит за статусом проведенных транзакций, связанных с идентификацией и аутентификацией. Обеспечивает корректную обработку ошибок, подтверждение успешного выполнения транзакций и обеспечение целостности данных.

2.2.2 Разработка транзакций с использованием Web3.py

Для эффективной реализации системы цифровой идентификации на основе блокчейна Ethereum, необходим механизм взаимодействия для проведения транзакций (Рисунок 2.7 — Принцип работы транзакций). Для этого используется библиотека Web3.py, предоставляющая API для взаимодействия с Ethereum, через язык программирования Python.

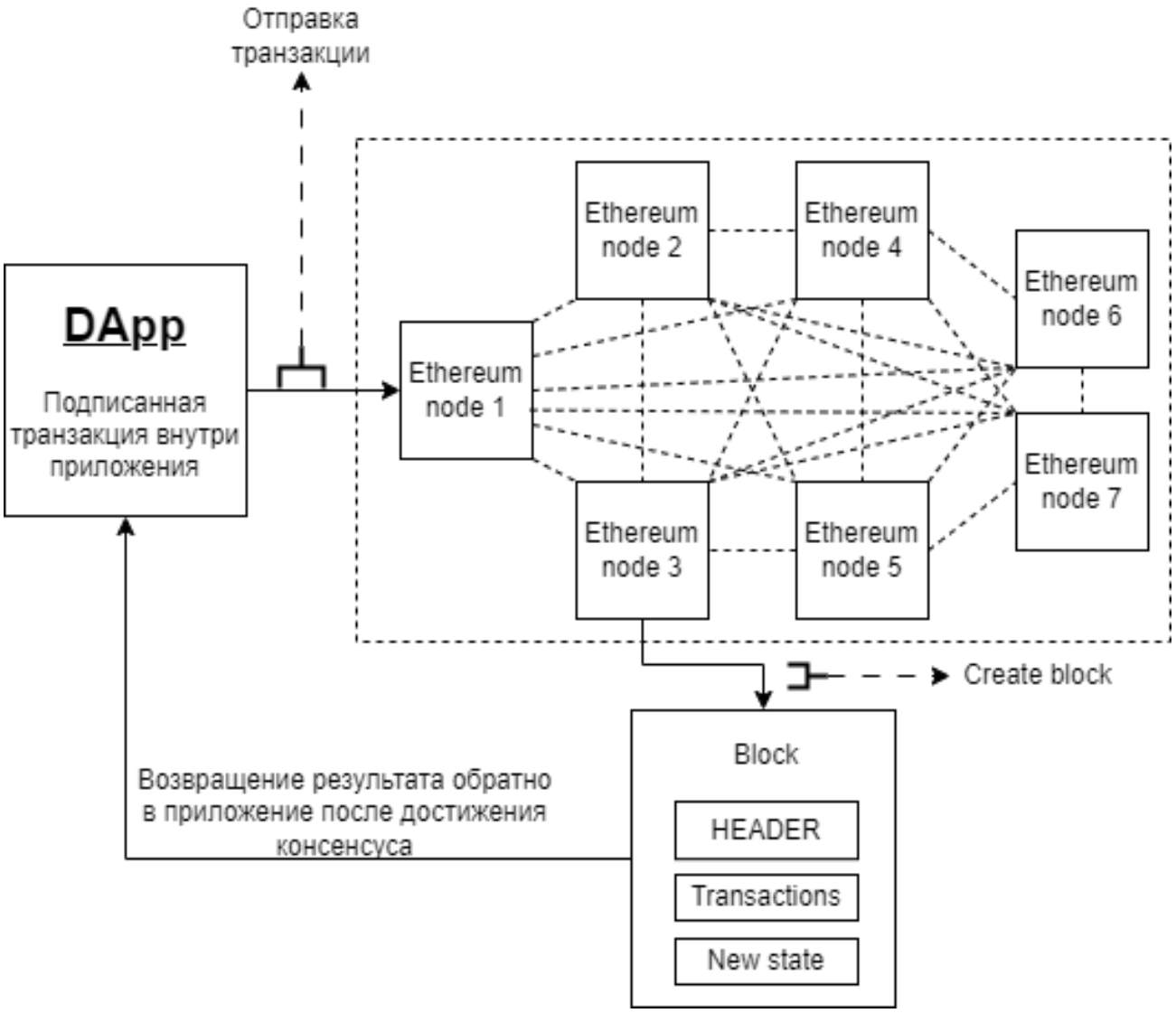


Рис. 2.7 Принцип работы транзакций

Процесс разработки транзакций включает в себя несколько этапов.

Сначала определяются параметры транзакции, такие как адрес получателя, сумма перевода в GAS (это единица измерения вычислительных усилий, для выполнения операций в сети Ethereum – или другими словами, комиссия) и данные, которые будут включены в транзакцию. Затем формируется сама транзакция, с использованием Web3.py.

Важным этапом является подпись транзакции с использованием приватного ключа пользователя (Рис 2.8 — Подписание приватным ключом). Это гарантирует, что транзакция происходит от правильного аккаунта и подтверждается легитимность действия. Web3.py предоставляет удобные методы для подписи транзакции с использованием приватного ключа.

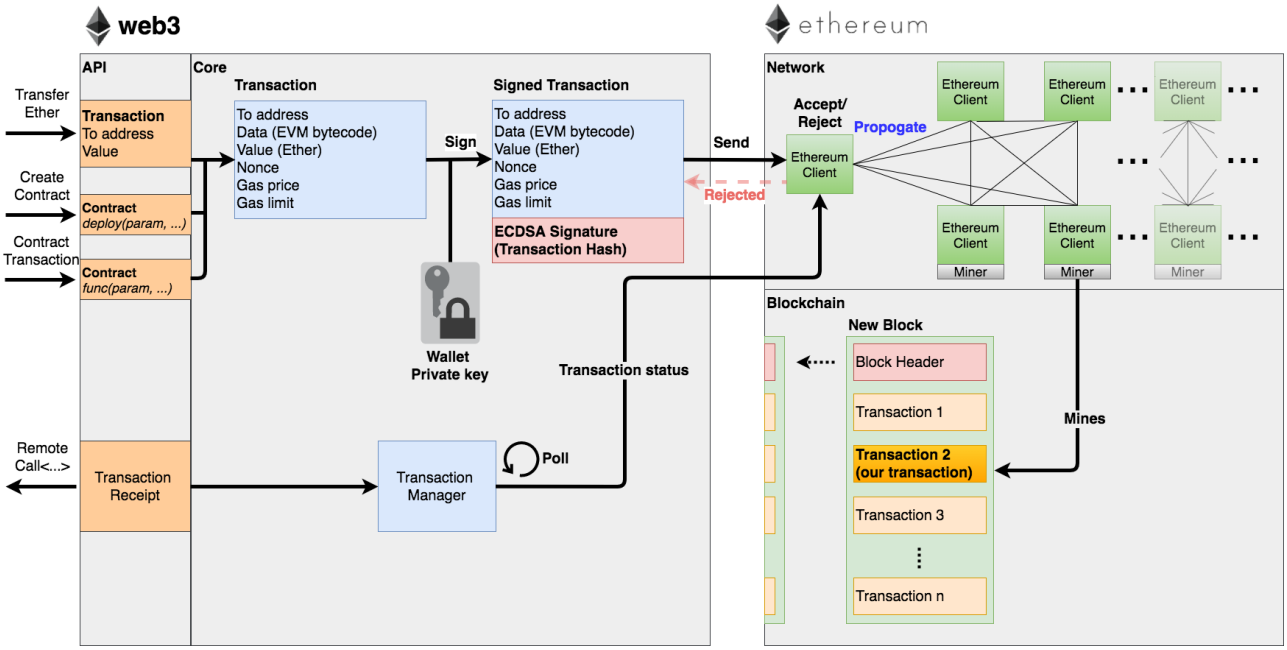


Рис. 2.8 Подписание приватным ключом

Далее, подписанная транзакция отправляется в сеть Ethereum, проходит процесс проверки, подтверждения и включения в блок, что обеспечивает надежность и неизменяемость транзакции в блокчейне.

Необходимо также предусмотреть механизм контроля статуса транзакции и обработки возможных ошибок (Рис 2.9 — Транзакционная структура). Это включает в себя мониторинг включения транзакции в блок, а также обработку возможных сбоев, таких как отсутствие подтверждения или отклонение транзакции.

Рассмотрим виды транзакций, применяемых при разработке системы цифровой идентификации УИИ:

- Транзакция регистрации пользователя — инициирует процесс регистрации нового пользователя в системе. Включает в себя передачу необходимых данных о пользователе, таких как UID, ФИО, электронная почта и адрес кошелька Metamask.
- Транзакция подписи документа — используется для подтверждения подписи электронных документов пользователя. После успешной аутентификации идентичности пользователя, система генерирует хэш документа, который затем подписывается приватным ключом, создавая надежную цифровую подпись.

- Транзакция завершения курса — фиксирует успешное завершение студентом образовательного курса. Включает в себя передачу данных о завершенном курсе, аттестации, документе об образовании и сертификата участия в каком—либо проекте (если участвовал).
- Транзакция запроса верификации навыков — студент может инициировать данную транзакцию для запроса подтверждения навыков, приобретенных в ходе обучения. Это может быть использовано для создания цифрового портфолио, подтверждающего уровень компетенций студента и демонстрации работодателю.
- Транзакция выпуска документа об образовании — при успешном завершении курса, система инициирует транзакцию по выпуску электронного документа об образовании. Включает в себя передачу данных о завершенной программе, статусе, полученной специальности, подписи учебного заведения.

Такие транзакции обеспечивают взаимодействие пользователей с блокчейн системой, поддерживая безопасность и подлинность данных в образовательной среде.

2.2.3 Реализация механизма подписи

Разработанный механизм подписи в системе цифровой идентификации представляет собой устойчивую систему, готовую к эффективному внедрению. В процессе разработки были учтены все ключевые аспекты, обеспечивающие безопасность и надежность цифровых подписей в блокчейне.

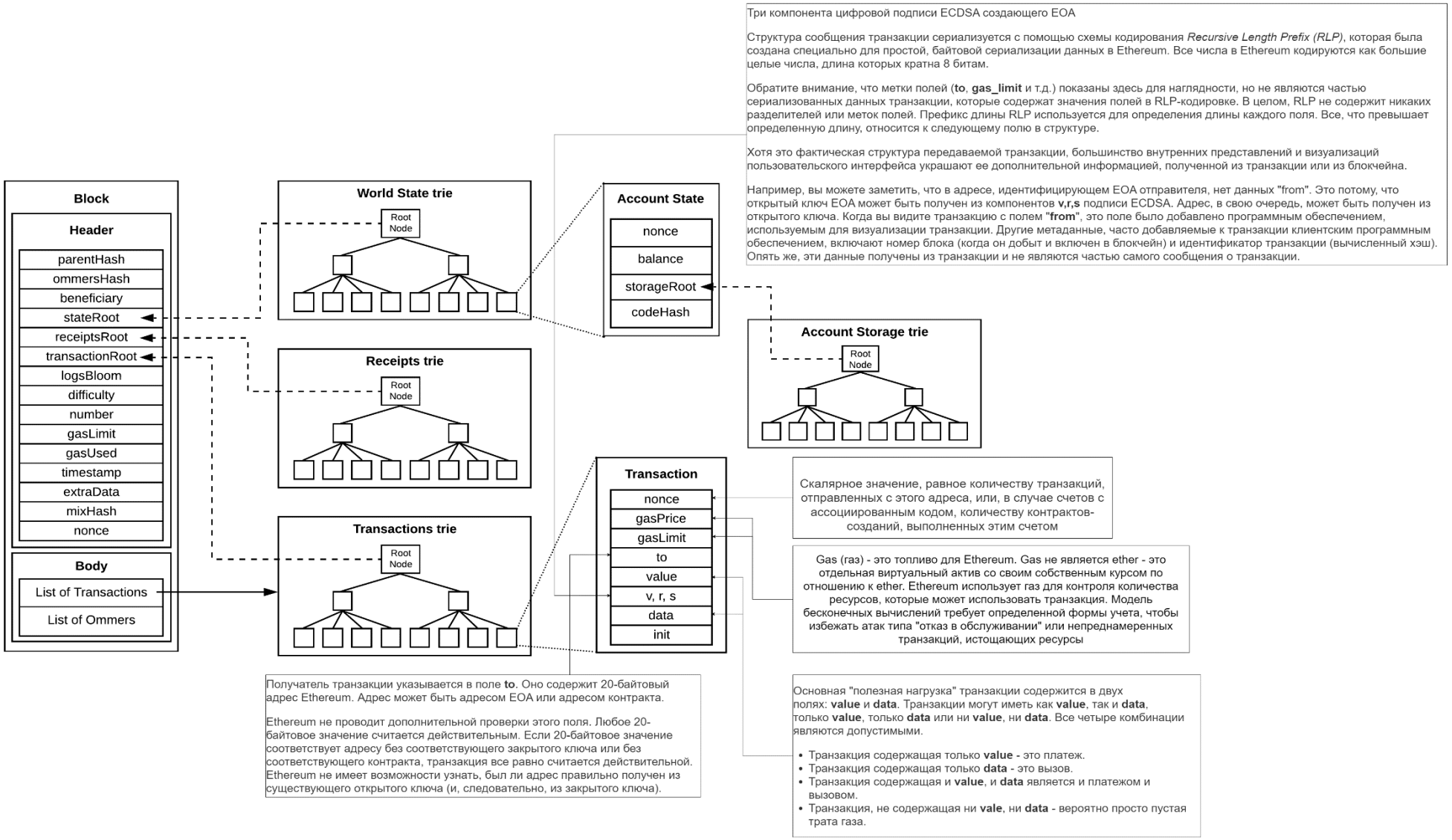


Рис. 2.9 Транзакционная структура

Выбор эллиптической кривой (ECDSA) как криптографического алгоритма гарантирует высокий уровень безопасности при создании подписей [10]. Уникальные ключевые пары, состоящие из открытых и приватных ключей для каждого участника, обеспечивают надежное взаимодействие в рамках блокчейн—сети.

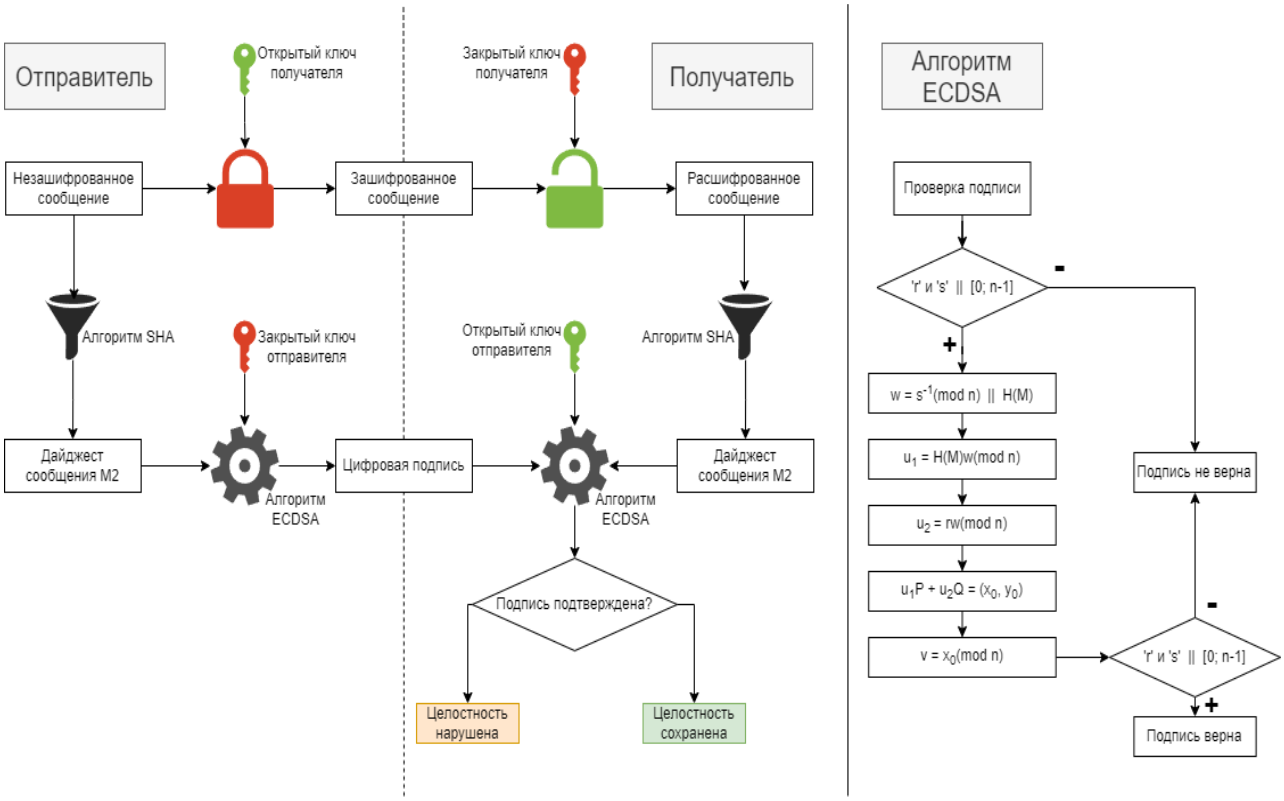


Рис. 2.10 Ассиметричный алгоритм целостности

Процесс подписания данных стандартизирован и интегрирован в транзакции системы цифровой идентификации (Рис 2.10 — Ассиметричный алгоритм целостности), что обеспечивает единообразие и безопасность в каждом этапе взаимодействия. Механизм подписи поддерживает эффективную верификацию подписей, обеспечивая подлинность данных.

Обработка ошибок и исключений стала неотъемлемой частью реализации, гарантируя корректное функционирование системы даже в случаях отсутствия подписей или возникновения проблем с ключами.

В целом, реализованный механизм подписи представляет собой тщательно продуманное решение, готовое к надежному внедрению в систему цифровой идентификации, обеспечивая безопасность и целостность данных в блокчейне.

2.2.4 Разработка механизма контроля статуса транзанзакции и обработки ошибок

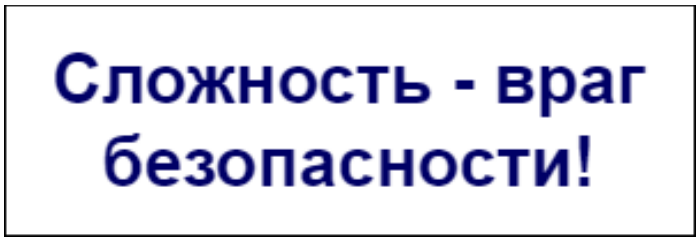
Механизм контроля статуса транзакции и обработки ошибок в системе цифровой идентификации представляет собой важный компонент, обеспечивающий надежность и целостность процесса взаимодействия в блокчейне.

При разработке данного механизма были учтены особенности работы с транзакциями в сети Ethereum [5]. Система обеспечивает мониторинг статуса каждой транзакции, начиная с ее отправки и заканчивая подтверждением включения в блок. Это позволяет пользователям отслеживать текущее состояние своих операций и обеспечивает прозрачность в рамках взаимодействия, так же есть специализированное веб приложение, для отслеживания любой транзакции в сети Ethereum (etherscan).

Механизм обработки ошибок разработан с учетом разнообразных сценариев возможных неудачных событий. В случае, если транзакция не может быть успешно выполнена, система предусматривает соответствующую обработку и уведомление об ошибке. Это включает в себя корректное ведение логов, оповещение участников системы о возникших проблемах, а также возможность повторной отправки транзакции при необходимости.

Разработанный механизм обеспечивает надежную и устойчивую систему контроля статуса транзакции, что является важным аспектом для обеспечения бесперебойной работы цифровой идентификации в блокчейне.

2.2.5 Разработка механизма защиты данных



Разработанный механизм защиты данных представляет собой внимательно спроектированное решение, учтенные все тонкости и нюансы. В его основе лежит комплексный подход, гарантирующий высокий уровень безопасности и конфиденциальности в рамках системы цифровой идентификации на блокчейне.

1. Использование современных алгоритмов шифрования: Чувствительные данные, включая идентификационную информацию студентов и результаты обучения, защищены с использованием передовых алгоритмов шифрования. Методы хэширования и асимметричного шифрования обеспечивают надежную защиту от несанкционированного доступа.

2. Система мониторинга и обнаружения аномалий: Внедрена система мониторинга, которая постоянно отслеживает действия в системе и выявляет любые аномалии. Автоматизированные реакции на необычные события обеспечивают оперативное противодействие угрозам безопасности.

3. Строгая система управления доступом: в системе реализована детализированная система управления доступом с использованием ролевой модели и многоуровневой идентификации. Это обеспечивает контроль над различными уровнями доступа и предотвращает несанкционированный доступ.

4. Регулярные аудиты безопасности: в рамках механизма предусмотрены периодические аудиты безопасности для выявления и устранения потенциальных уязвимостей. Постоянное обновление системы гарантирует соответствие современным стандартам безопасности.

5. Превентивные меры по уменьшению рисков утечек данных: разработаны и внедрены превентивные меры, включая строгий контроль доступа, механизмы резервного копирования данных и обеспечение физической безопасности серверов.

Этот цельный механизм обеспечивает надежную защиту данных, предоставляя безопасное окружение для системы цифровой идентификации на блокчейне.

2.2.6 Лучшие практики безопасности смарт контрактов

1. Минимализм и простота — чем проще код и чем меньше он делает, тем меньше вероятность возникновения ошибки или непредвиденного эффекта. Когда приступаешь к программированию смарт контрактов, возникает соблазн написать много кода. Вместо этого нужно смотреть код и искать способы сделать

его короче, с меньшим количеством строк кода, меньшей сложностью и меньшим количеством функций. Если кто—то говорит, что его проект «тысячи строк кода» для своих смарт контрактов, следует усомниться в безопасности. Проще — значит надежнее!

2. Повторное использование кода — стараться не изобретать колесо. Если уже существует библиотека или контракт, который делает большую часть того, что нужно, использовать его повторно. В своем коде следую паттерну DRY (don't repeat yourself): «Не повторяйтесь». Если есть повторяющийся фрагмент кода более одного раза, тогда лучше записать в отдельную функцию. Код, который уже многократно используется и тестируется, более безопасен, чем любой новый код.

3. Качество кода — не надо относиться к программированию смарт контрактов так же, как к программированию общего назначения. Написание DApps на Solidity не похоже на создание веб виджета на JavaScript.

4. Читаемость и контролируемость — код должен быть понятным и простым для восприятия. Чем проще он читается, тем проще его контролировать. Смарт контракты являются публичными, поэтому каждый может прочитать байткод и любой может провести обратный инжиниринг. Писать хорошо документированный и легко читаемый код, следуя стилю и соглашениям об именовании, принятым в сообществе Ethereum.

5. Покрытие тестов — тестировать все, что можно. Смарт контракты работают в публичной среде исполнения, где любой может выполнить их с любыми входными данными. Никогда не полагать, что входные данные, такие как аргументы функций, хорошо сформированы, правильно ограничены или имеют доброкачественное назначение. Проверить все аргументы, чтобы убедиться, что они находятся в ожидаемых диапазонах и правильно отформатированы, прежде чем разрешить выполнение кода.

2.3 Работа с базой данных

2.3.1 Проектирование структуры базы данных для хранения информации о пользователях.

Проектирование базы данных для системы цифровой идентификации УИИ, является важным этапом, направленным на обеспечение эффективного и

безопасного хранения информации о пользователях. В основе структуры лежит цель обеспечения высокой производительности, надежности и гибкости при обработке данных о студентах и других участниках образовательного процесса (Рис 2.11 — Схема базы данных УИИ).

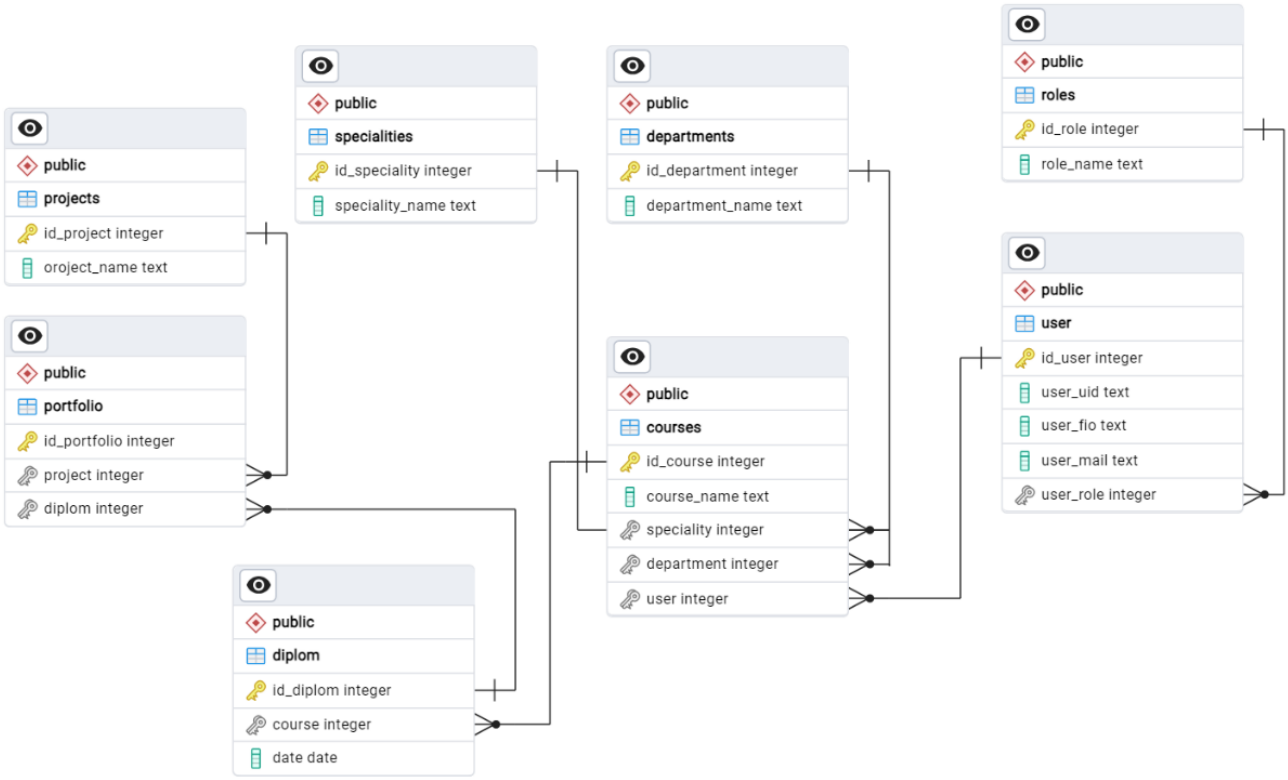


Рис. 2.11 Схема базы данных УИИ

Опишем схему базы данных системы цифровой идентификации платформы УИИ, на примере сущностей и связей.

1. Таблица «user»:

id_user (integer, PRIMARY KEY) — идентификатор пользователя (целое число), первичный ключ.

user_uid (text, NOT NULL) — уникальный идентификатор пользователя (текстовое поле), формируется Ethereum кошельком.

user_fio (text, NOT NULL) — ФИО пользователя (текстовое поле).

user_mail (text, NOT NULL) — электронная почта пользователя (текстовое поле).

user_role (integer) — внешний ключ, ссылается на «id_role» в таблице «roles».

2. Таблица «roles»:

id_role (integer, PRIMARY KEY) — идентификатор роли (целое число), первичный ключ.

role_name (text, NOT NULL) — название роли (текстовое поле).

3. Таблица «departments»:

id_department (integer, PRIMARY KEY) — идентификатор кафедры (целое число), первичный ключ.

department_name (text, NOT NULL) — название кафедры (текстовое поле).

4. Таблица «specialities»:

id_speciality (integer, PRIMARY KEY) — идентификатор специальности (целое число), первичный ключ.

speciality_name (text, NOT NULL) — название специальности (текстовое поле).

5. Таблица «portfolio»:

id_portfolio (integer, PRIMARY KEY) — идентификатор портфолио (целое число), первичный ключ.

project (integer) — внешний ключ, ссылается на «id_project» в таблице «projects».

diplom (integer) — внешний ключ, ссылается на «id_diplom» в таблице «diplom».

6. Таблица «projects»:

id_project (integer, PRIMARY KEY) — идентификатор проекта (целое число), первичный ключ.

project_name (text) — название проекта (текстовое поле).

7. Таблица «diplom»:

id_diplom (integer, PRIMARY KEY) — идентификатор диплома (целое число), первичный ключ.

course (integer, NOT NULL) — внешний ключ, ссылается на «id_course» в таблице «courses».

date (date, NOT NULL) — дата получения диплома (дата).

8. Таблица «courses»:

id_course (integer, PRIMARY KEY) — идентификатор курса (целое число), первичный ключ.

course_name (text, NOT NULL) — название курса (текстовое поле).

speciality (integer) — внешний ключ, ссылается на «id_speciality» в таблице "specialities".

department (integer) — внешний ключ, ссылается на «id_department» в таблице «departments».

user (integer) — внешний ключ, ссылается на «id_user» в таблице «user».

Приведенная структура базы данных системы идентификации УИИ, приведена к третьей нормальной форме (3НФ), означая, что каждый атрибут нетривиально зависит от первичного ключа и не зависит от других не ключевых атрибутов. Таким образом устраняется избыточность, повышается целостность и упрощается работа с данными.

2.3.2 Реализация механизма взаимодействия с базой данных для идентификационных данных.

Эффективная работа системы цифровой идентификации УИИ на основе блокчейна, обуславливает необходимость реализации специализированного механизма взаимодействия с базой данных, предназначенного для хранения и обработки идентификационных данных пользователей (Приложение Б — Программный код).

1. Подключение к базе данных:

- Подключение к защищенной базе данных PostgreSQL версии 16 посредством драйвера Psycopg2.
- Обеспечение устойчивой и безопасной связи между приложением и базой данных.
- Создание основы для последующих операций с данными.

2. Разработка запросов к базе данных:

- Разработка запросов для выполнения различных операций с базой данных.

- Обеспечение:
  - Добавления новых пользователей;
  - Актуализации их данных;
  - Извлечения информации о пользователях;
  - Удаление пользователя;
  - Добавление информации о проектах пользователю
- Проектирование запросов с учетом требований к эффективности и безопасности.

3. Механизм обработки данных:

- Вставка идентификационных данных при регистрации новых пользователей.
- Проверка существующих данных при аутентификации пользователей.
- Обеспечение:
  - Достоверности и актуальности данных;
  - Корректности и безопасности идентификационного процесса.

4. Безопасность и защита данных:

- Применение современных алгоритмов шифрования для защиты данных от утечек и несанкционированного доступа.
- Обеспечение:
  - Высокого уровня безопасности системы;
  - Соответствия современным стандартам информационной безопасности.

5. Оптимизация производительности:

- Оптимизация структуры базы данных:
  - Создание индексов;
  - Настройка таблиц.
- Оптимизация сложных запросов.
- Обеспечение:
  - Высокой эффективности работы с данными;

- Отзывчивости системы.

6. Логирование и мониторинг:

- Внедрение механизмов логирования для фиксации всех операций с базой данных.
- Мониторинг работы базы данных:
  - Выявление узких мест;
  - Оптимизация запросов;
  - Контроль за производительностью;
  - Отслеживание нагрузки на базу данных.
- Обеспечение:
  - Прозрачности работы системы;
  - Управляемости и отладки системы;
  - Анализа и оптимизации работы базы данных.

Представленный механизм взаимодействия с базой данных обеспечивает надежное, безопасное и эффективное хранение и обработку идентификационных данных пользователей в рамках системы цифровой идентификации УИИ.

2.4 Генерация NFT токена из документов об образовании

Генерация NFT (Non—Fungible Token) из данных документа об образовании, представляет собой, процесс формирования уникального, не заменяемого токена на блокчейне Ethereum (для этого применяется стандарт ERC721), который соответствует конкретной информации о полученном образовании и не может быть изменен (Рисунок 2.12 — Базовый принцип создания NFT). Этот механизм добавляет цифровую уникальность и подтверждение подлинности образовательным документам, выдаваемых УИИ [5, 7, 13].

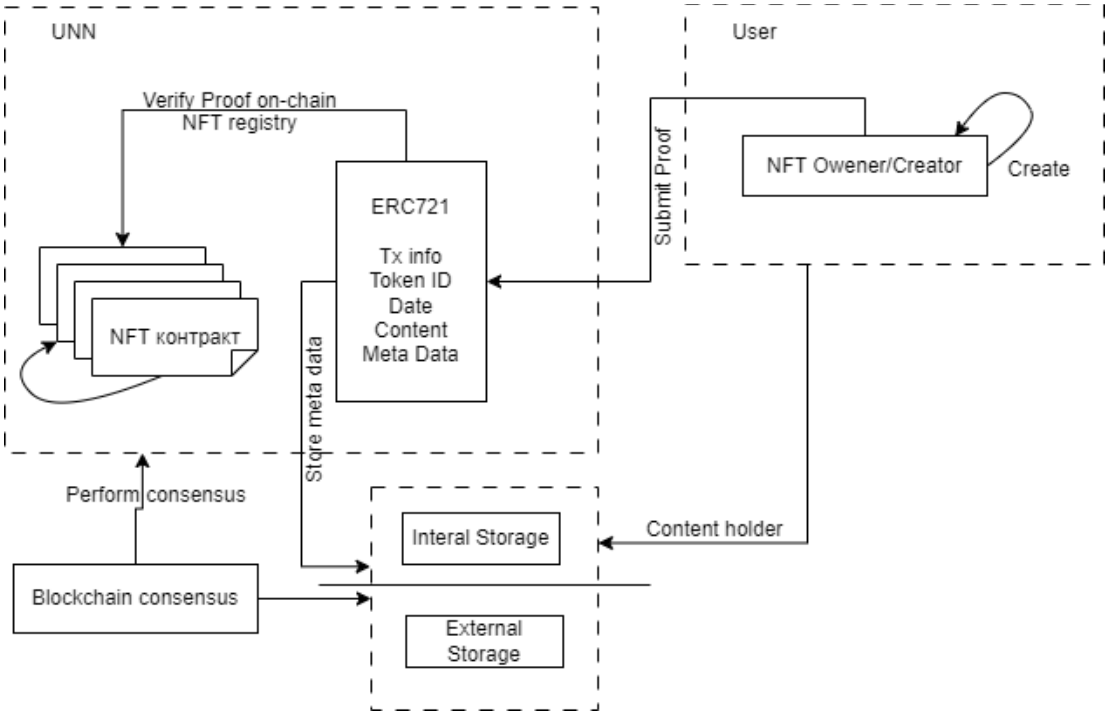


Рис. 2.12 Базовый принцип создания NFT

Информация из документа об образовании (название курса, специальность, факультет, ФИО, дата выдачи, университет) преобразуется в цифровой формат, затем создается уникальный NFT, связанный с этими данными, при этом пользователь навсегда останется его владельцем. Это обеспечивает неповторимость и неизменяемость информации, что делает токен уникальным представителем образовательного достижения.

Процесс генерации NFT включает в себя использование уникальных хэшей, цифровой подписи, идентификации владельца, а также метаданных, описывающих содержание и характеристики образовательного документа. Этот уникальный токен можно проверить, для подтверждения информации.

Такая функциональность не только обеспечивает безопасность и надежность цифровых образовательных документов, но и открывает новые перспективы для их использования в цифровых экосистемах, таких взаимодействие с работодателями и клиентами.

2.5 Заключение

В заключении к второй главе, продемонстрировал сжатое обобщение выполненной работы, включая базу данных, тестирование, NFT токены, смарт контракты и другие элементы системы, как шифрование и ECDSA [16].

Создал комплексную, интегрированную систему, охватывающей разнообразные аспекты цифровой идентификации и образования. Разработка базы данных обеспечивает эффективное хранение и управление идентификационными данными, а также быстрый и надежный доступ к информации. Тестирование смарт контрактов гарантирует их корректность и безопасность функционирования.

Добавил функционал NFT токенов, к образовательным документам, который придает им цифровую уникальность и подтверждение подлинности, открывая новые возможности для их использования. Реализовал шифрование для обеспечения защиты данных.

Смарт контракты и транзакции предоставляют дополнительные инструменты для автоматизации процессов, улучшения прозрачности и поддержания децентрализованности. Эти элементы в совокупности формируют целостную и инновационную систему цифровой идентификации пользователей в УИИ.

Следует отметить, что вся работа этой главы направленна на создание современной и функциональной цифровой платформы университета, не только для УИИ, но и для всех, кто захочет внедрить эту систему себе.

Таким образом, вторая глава представляет собой уникальный технический вклад в область цифровых технологий УИИ и демонстрирует широкий спектр навыков и компетенций в разных областях программирования [16, 17].

3.ЭКОНОМИЧЕСКИЕ ПРЕИМУЩЕСТВА

3.1. Экономические преимущества

Внедрение системы цифровой идентификации пользователей и предметов на основе блокчейн Ethereum в УИИ, приносит ряд экономических выгод.

Снижение расходов:

- Уменьшение затрат на бумажные носители (печать дипломов, свидетельств, справок, хранение и транспортировка);
- Автоматизация административных процессов (проверка подлинности документов, уменьшение времени на обработку запросов);
- Сокращение расходов на ИТ отдел (централизованное хранение данных, необходимость в дублировании записей).

Повышение доходов:

- Создание новых услуг (платная проверка подлинности документов, предоставление доступа к образовательным данным третьим лицам);
- Увеличение конкурентоспособности УИИ (привлечение новых студентов за счет инновационных технологий, повышение имиджа).

3.2 Доказательство экономической выгоды

3.2.1 Расчет стоимости регистрации

Система УИИ:

- Затраты на носители — 50 руб. (за студента);
- Затраты на персонал — 1 ч. (работы сотрудника) 500 руб.;
- Время регистрации — 15 мин. (0,25 ч.);

Итого: 50 + 500 * 0,25 = 175 руб.

Система блокчейн:

- Затраты на GAS: 0,001 ETH = 20 рублей (при текущем курсе, на 20.02.2024).
- Затраты на вычислительные мощности: 0,0001 ETH = 3 руб.;

Итого: 20 + 3 = 23 руб.

Экономия: 175 — 23 = 152 руб.

3.2.2 Расчет стоимости аутентификации

Система УИИ:

- Затраты на персонал — 1 ч. (работы сотрудника) = 500 руб.;
- Время аутентификации — 5 мин. = 0,083 ч.;

Итого: 500 * 0,083 = 41,5 руб.

Система блокчейн:

- Затраты на газ Ethereum — 0,0005 ETH = 5 руб.;

Итого: 5 руб.

Экономия: 41,5 — 5 = 36,5 руб.

3.2.3 Расчет стоимости подписи транзакций

Система УИИ:

- Затраты на носители: 5 руб.;
- Затраты на персонал: 1 ч. (работы сотрудника) 500 руб.;
- Время подписи: 10 мин. (0,167 ч.);

Итого: 5 + 500 * 0,167 = 83,35 руб.

Система на основе блокчейн:

- Затраты на газ Ethereum: 0,001 ETH = 10 руб.;

Итого: 10 руб.

Экономия: 83,35 — 10 = 73,35 руб.

3.2.4 Расчет стоимости сервера

Система УИИ:

- Затраты на серверное оборудование: 87500 руб. в мес.;
- Затраты на электроэнергию: 5000 руб. в мес.;
- Затраты на обслуживание: 3750 руб. в мес.;

Итого: 87500 + 5000 + 3750 = 96250 руб. в мес.

Система на основе блокчейн:

- Затраты на электроэнергию: 700 руб. в мес.;

Итого: 700 руб.

Экономия: 96250 — 700 = 95550 руб.

Подведем общий итог: $Э = 36{,}5 * 30 + 73{,}35 * 30 + 95550 = 1095 + 2200{,}5 + 95550 = 98845{,}5$ руб., таки образом минимальная экономия в месяц, составляет 98845,5 руб. (при расчете 1 студент в месяц).

3.3 Заключение

Рассмотрел экономическое преимущество, которое привело к конкретному результату, после внедрения системы блокчейн Ethereum в УИИ. Исследование охватило широкий спектр факторов, начиная, от сокращения расходов на бумажные носители, до автоматизации административных процессов, повышение конкурентоспособности и создание новых услуг. Внимательный анализ стоимости операций, таких как регистрация, аутентификация, подписание транзакций и создание новых блоков, подтвердил экономические выгоды.

Общая экономия, в размере 98845,5 руб. в месяц, выделяет эффективность предложенной системы (в год 1186146 руб.), на 1 студента в месяц. Результаты отражают финансовые выгоды, но и подчеркивают ценность ИТ, для повышения эффективности административных и управленческих процессов в УИИ. Так же, подчеркиваю существенный вклад блокчейн, в уменьшение затрат и повышение производительности университета.

Эти результаты открывают большие перспективы, которые возникают при широком внедрении инновационных технологий УИИ.

ЗАКЛЮЧЕНИЕ

В ходе выполнения исследования по дипломной работе, проведено глубокое исследование области цифровой идентификации пользователей и объектов на платформе УИИ, а также разработана и реализована библиотека, предназначенная для эффективного и безопасного управления данными. Обобщая результаты исследования, сформировал следующие ключевые выводы:

- Разработка цифровой идентификационной системы — создана система, предоставляющая эффективные механизмы идентификации пользователей с использованием современных технологий блокчейн Ethereum и Python библиотеки Web3.py. Разработанный программный продукт включает в себя модули для регистрации, аутентификации и управления идентификационными данными.
- Тестирование и безопасность — проведено тестирование смарт контрактов и механизмов обработки идентификационных данных. Результаты подтвердили надежность и безопасность системы, её способность эффективно защищать конфиденциальную информацию и отслеживать изменения.
- Интеграция и использование — система успешно интегрирована в образовательную среду УИИ. Полученные результаты свидетельствуют о практической значимости и применимости разработанной системы.
- Научная инновационность — исследование внедряет инновационные технологии в область цифровой идентификации, предлагая подходы, учитывающие современные требования к безопасности и эффективности систем управления идентификацией.

Обобщение результатов исследования в этой дипломной работе, подчеркивает значимость проделанной работы и её вклад в развитие сферы цифровой идентификации. Полученные результаты представляют собой основу для дальнейшего улучшения и развития систем управления идентификацией в образовательных учреждениях и других сферах применения.

Но это не всё, мои разработки, проведенные в данной дипломной работе, размещены в официальных общественных репозиториях. Этот шаг осуществлен

с целью стимулирования и развития open source движения в России. Предоставление открытого доступа к коду и проектам способствует коллективной работе, обмену опытом и знаниями, а также создает благоприятную среду для совместного технологического развития. Этот подход не только способствует прозрачности и доступности разработок, но и поощряет внесение вклада со стороны сообщества, способствуя инновациям, созданию устойчивой экосистемы в области цифровых технологий и развития ИТ сектора в России.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Генкин, А.С. Блокчейн для всех: Как работают криптовалюты Baas, NFT, DeFi и другие финансовые технологии. Москва: Альпина Паблишер, 2023. — 588 с.
2. Могайар, У. Блокчейн для бизнеса. Москва: Эксмо, 2018. — 224 с.
3. Башир, Имран Блокчейн: архитектура, криптовалюты, инструменты разработки, смарт—контракты. Москва: Трэнтэкс, 2019. — 538 с.
4. Свон, Мелони Блокчейн: схема новой экономики. Москва: Олимп—Бизнес, 2018. — 240 с.
5. Иванов, А.Ю. Блокчейн на пике хайпа Правовые риски и возможности (мВШЭ) Иванов. Москва: ВШЭ, 2017. — 237 с.
6. Хата, Е. Блокчейн для бабушки за 60 минут. СПб: Пальмира, 2018. — 84 с.
7. Сажина, М.А. Блокчейн в системе управления знанием. Монография. Москва: Форум, 2019. — 90 с.
8. Сушков, В.А. Метавселенные, DeFi и NFT. Путеводитель в мир инноваций. Москва: Перо, 2022. — 156 с.
9. Прасти, Нарьян Блокчейн. Разработка приложений. СПб: БХВ, 2019. — 256 с.
10. Ищукова, Е.А. Криптографические основы блокчейн—технологий. Москва: ДМК пресс, 2022. — 302 с.
11. Богданова, И.А. Конституционное право в Московском университете. Хрестоматия. Учебное пособие: учеб. пособие / Д.Г. Шустрова. — Москва: Проспект, 2023. — 464 с.
12. Макаров, А.Н. Методология научных исследований в университетах и промышленных компаниях. Вологда: Инфра — Инженерия, 2023. — 276 с.
13. Петросянц, Д.В. Современные методики оценки эффективности деятельности университетов. Монография. Москва: Проспект, 2022. — 176 с.
14. Сидорова, А.А. Сотрудничество университета и бизнеса в цифровую эпоху. Москва: Аргамак—Медиа, 2021. — 240 с.

15. Васильев, Юлий Python для Data science. СПб: Питер, 2023. — 272 с.

16. Чоу, Эрик Python для сетевых инженеров. Автоматизация сети, программирование и DevOps. СПб: Питер, 2022. — 528 с.

17. Ленс, Мориц Python. Непрерывная интеграция и доставка. Москва: ДМК пресс, 2020. — 168 с.

18. Янка, Таня Безопасность веб—приложений. Исчерпывающий гид для начинающих разработчиков. Москва: Бомбора, 2023. — 464 с.

19. Скабцов, Никита Kali Linux в действии. Аудит безопасности информационных систем. Москва: Прогресс книга, 2024. — 384 с.

20. Мартынов, А.П. Информационная безопасность и защита информации. Москва: Ай Пи Эр Медиа, 2024. — 130 с.

21. Левашев, П.Ю. Киберкрепость: всестороннее руководство по компьютерной безопасности. Москва: Прогресс книга, 2024. — 544 с.

22. Максимов, А.Е. Изучаем процессы проектирования, обеспечения безопасности, применения механизации, определения стоимости строительства. Москва: Инфра—Инженерия, 2024. — 136 с.

23. Балаков, А.Н. Комплексная информационная безопасность. Полный справочник специалиста. Москва: Инфра—Инженерия, 2024. — 156 с.

24. Граймс, Роджер Как противостоять хакерским атакам. Уроки экспертов по информационной безопасности. Москва: Бомбора, 2023. — 368 с.

25. Бирюков, А.А. Информационная безопасность: защита и нападение. Москва: ДМК Пресс, 2023. — 440 с.

ПРИЛОЖЕНИЕ 1

Техническое задание

1. Введение
1.1. Наименование проекта: Система идентификации пользователей и объектов с применением блокчейн
1.2. Цель проекта: Создание системы, автоматизирующей образовательные процессы в УИИ.
1.3. Заказчик: Администрация УИИ
1.4. Исполнитель: Деркунов Михаил Юрьевич
1.5. Дата: 16.11.2023
2. Общие положения
2.1. Общие сведения:

- Проект включает в себя разработку системы идентификации пользователей и предметов с применением блокчейн технологий, а также токенизацию учебных достижений.
- Система должна быть интегрирована с существующей информационной системой УИИ.
- Система должна быть масштабируемой и надежной.

2.2. Термины и определения:

- Блокчейн: Распределенный реестр, обеспечивающий неизменность и безопасность записей.
- Токен: Цифровой актив, который может быть использован для представления различных ценностей, таких как права доступа, баллы или предметы.
- ERC20: Стандарт токена Ethereum, который используется для создания взаимозаменяемых токенов.
- ERC721: Стандарт токена Ethereum, который используется для создания уникальных токенов.

3. Требования к системе
3.1. Функциональные требования:

- Интеграция с блокчейн Ethereum.
- Создание программного модуля для взаимодействия с блокчейн.
- Реализация стандартов ERC20 и ERC721.
- Обеспечение транзакционных возможностей.
- Передача токенов между пользователями.
- Создание и выдача токенов.

- Сжигание токенов.
- Обеспечение возможности токенизации учебных достижений.
- Выдача токенов за успешное прохождение курсов.
- Использование токенов для оплаты обучения.
- Использование токенов для доступа к образовательным материалам.

3.2. Требования к безопасности:

- Разработка механизма аутентификации и авторизации пользователей.
- Шифрование конфиденциальных данных.
- Установка системы мониторинга безопасности.
- Обеспечение защиты от несанкционированного доступа к данным.
- Обеспечение целостности данных.

3.3. Требования к производительности:

- Обеспечение высокой отзывчивости системы.
- Тестирование системы на производительность при максимальной нагрузке.
- Обеспечение возможности обработки большого количества транзакций.

3.4. Требования к интерфейсу:

- Обеспечение удобного и понятного интерфейса для пользователей.
- Обеспечение поддержки различных устройств:
- Компьютеры.
- Планшеты.
- Смартфоны.

3.5. Требования к масштабируемости:

- Создание расширяемой архитектуры для подготовки к увеличению числа пользователей и объема данных.
- Возможность горизонтального масштабирования системы.

3.6. Требования к тестированию:

- Разработка плана и сценариев тестирования.
- Проведение тестирования на всех этапах разработки:
- Единичное тестирование.

- Интеграционное тестирование.
- Системное тестирование.
- Приемочное тестирование.

3.7. Требования к документации:

- Подготовка технической документации:
- Описание системы.
- Руководство пользователя.
- Руководство администратора.

3.8. Требования к обеспечению совместимости:

- Гарантирование совместимости системы с основными браузерами:
- Chrome.
- FireFox.
- Safari.
- Edge.
- Яндекс.
- Гарантирование совместимости системы с основными операционными системами:
- Linux.
- MacOS.
- Windows.

3.9. Требования к дополнительному программному обеспечению:

- Использование библиотеки Web3.py.
- Язык программирования: Python.

3.10. Требования к гарантированному сопровождению:

- Обеспечение гарантированного сопровождения системы в течение 12 месяцев.
- Организация службы поддержки пользователей.

4. Этапы разработки

4.1. Анализ требований:

- Определение функциональных и нефункциональных требований к системе.
- Разработка технического задания.

4.2. Проектирование:

- Разработка архитектуры системы.
- Разработка интерфейса пользователя.
- Разработка программного обеспечения.

4.3. Разработка:

- Реализация программного обеспечения.
- Тестирование программного обеспечения.

4.4. Внедрение:

- Установка и настройка системы.
- Обучение пользователей.

4.5. Сопровождение:

- Исправление ошибок.
- Доработка системы.
- Обновление системы.

5. Календарный план разработки

- Анализ требований — 2 недели.
- Проектирование — 4 недели.
- Разработка — 8 недель.
- Тестирование — 4 недели.
- Внедрение — 2 недели.
- Сопровождение — 12 месяцев.

6. Стоимость разработки

Разработка на добровольной основе, для поддержания open-source движения ИТ России.

7. Контакты

Заказчик:

- Наименование: Университет искусственного интеллекта
- ФИО: Романова Ирина Сергеевна
- Должность: директор

Исполнитель:

- ФИО: Деркунов Михаил Юрьевич
- Email: giocatory@yandex.ru

8. Приложения

- Техническое задание

- Код программы

ПРИЛОЖЕНИЕ 2

Код программы

Смарт контракты:
Регистрация пользователей

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
ontract UserRegistration {
    address public owner;
    mapping(address => bool) public registeredUsers;
    mapping(address => string) public userNames;
    event UserRegistered(address indexed user, string userName);
    modifier onlyOwner() {
            require(msg.sender == owner, "Вызвать эту функцию может только владе-
лец контракта");
            _;
    }
    modifier notRegistered() {
        require(!registeredUsers[msg.sender], "User is already registered");
        _;
    }
    constructor() {
        owner = msg.sender;
    }
    function registerUser(string memory userName) external notRegistered {
        require(bytes(userName).length > 0, "Username cannot be empty");
        registeredUsers[msg.sender] = true;
        userNames[msg.sender] = userName;
        emit UserRegistered(msg.sender, userName);
    }
    function getUserInfo(address userAddress) external view returns (string memory) {
        require(registeredUsers[userAddress], "User is not registered");
        return userNames[userAddress];
    }
    function changeOwnership(address newOwner) external onlyOwner {
        require(newOwner != address(0), "Invalid new owner address");
        owner = newOwner;
    }
}
```

Аутентификация

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract AuthenticationContract {
    address private owner;
```

```
    mapping(address => bool) private authorizedUsers;
    event UserAuthorized(address indexed user);
    modifier onlyOwner() {
        require(msg.sender == owner, "AuthenticationContract: Only the contract owner
can call this function");
        _;
    }
    modifier onlyAuthorized() {
        require(authorizedUsers[msg.sender], "AuthenticationContract: User is not au-
thorized");
        _;
    }
    constructor() {
        owner = msg.sender;
    }
    function authorizeUser(address userAddress) external onlyOwner {
        require(userAddress != address(0), "AuthenticationContract: Invalid user ad-
dress");
        authorizedUsers[userAddress] = true;
        emit UserAuthorized(userAddress);
    }
    function deauthorizeUser(address userAddress) external onlyOwner {
        require(userAddress != address(0), "AuthenticationContract: Invalid user ad-
dress");
        authorizedUsers[userAddress] = false;
    }
    function authenticate() external onlyAuthorized view returns (bool) {
require(ecrecover(hash, v, r, s) == msg.sender, "AuthenticationContract: Invalid sig-
nature");
require(validateSessionKey(msg.sender), "AuthenticationContract: Invalid session
key");
require(validateTwoFactorAuth(msg.sender), "AuthenticationContract: Two-factor
authentication failed");
return true;
    }
    function changeOwnership(address newOwner) external onlyOwner {
        require(newOwner != address(0), "AuthenticationContract: Invalid new owner
address");
        owner = newOwner;
    }
}
```

Управление токенами

ERC-20:

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
```

```
import "@openzeppelin/contracts/token/ERC20/ERC20.sol";
contract MyToken is ERC20 {
    constructor(string memory name, string memory symbol) ERC20(name, symbol) {
        // Развертывание контракта с указанием имени и символа токена
        _mint(msg.sender, 1000000 * 10 ** decimals());
        // Выдача начального количества токенов создателю контракта
    }
    function mint(address to, uint256 amount) external {
        // Функция для эмиссии новых токенов
        require(hasRole(DEFAULT_ADMIN_ROLE, msg.sender), "MyToken: must
have admin role to mint");
        // Проверка наличия у отправителя роли администратора
        _mint(to, amount);
        // Создание новых токенов
    }}
```

ERC-721:

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
import "@openzeppelin/contracts/token/ERC721/ERC721.sol";
import "@openzeppelin/contracts/access/Ownable.sol";
contract MyNFT is ERC721, Ownable {
    uint256 private _tokenIdCounter;
    constructor(string memory name, string memory symbol) ERC721(name, symbol)
{
        // Развертывание контракта с указанием имени и символа NFT
    }
    function mint(address to) external onlyOwner {
        // Функция для эмиссии нового уникального токена NFT
        uint256 tokenId = _tokenIdCounter;
        _safeMint(to, tokenId);
        _tokenIdCounter++;
        // Создание нового токена NFT и увеличение счетчика
    }}
```

Выдача документа об образовании

```
// Указываем версию компилятора
pragma solidity ^0.8.0;
// Импортируем стандарт ERC721
import "@openzeppelin/contracts/token/ERC721/ERC721.sol";
import "@openzeppelin/contracts/access/Ownable.sol";
// Объявляем контракт и наследуем его от ERC721 и Ownable
contract EducationTokenContract is ERC721, Ownable {
    // Событие, срабатывающее при выдаче нового документа об образовании
    event EducationDocumentIssued(address indexed recipient, uint256 tokenId);
    // Задаем имя и символ токена
    constructor() ERC721("EducationToken", "EDU") {}
```

```
    // Маппинг для отслеживания выданных документов
    mapping(uint256 => bool) private _documentsIssued;
    // Функция для выдачи нового документа об образовании
    function issueEducationDocument(address recipient) external onlyOwner {
        require(recipient != address(0), "EducationTokenContract: Invalid recipient ad-
dress");
        // Генерируем уникальный tokenId
        uint256 tokenId = totalSupply() + 1;
        // Создаем новый токен
        _safeMint(recipient, tokenId);
        // Отмечаем, что документ был выдан
        _documentsIssued[tokenId] = true;
        // Срабатывает событие
        emit EducationDocumentIssued(recipient, tokenId);
    }
    // Функция для проверки, был ли выдан документ с указанным tokenId
    function isDocumentIssued(uint256 tokenId) external view returns (bool) {
        return _documentsIssued[tokenId];
    }
}
```

Контроль статуса транзакции

```
// Указываем версию компилятора
pragma solidity ^0.8.0;
// Контракт токена
contract EducationToken {
    // Адреса пользователей и их статусы
    mapping(address => bool) public userStatus;
    // Событие для отслеживания изменения статуса
    event StatusChanged(address indexed user, bool newStatus);
    // Модификатор для проверки, что отправитель — владелец контракта
    modifier onlyOwner() {
        require(msg.sender == owner, "Only the owner can call this function");
        _;
    }
    // Адрес владельца контракта
    address public owner;
    // Конструктор контракта
    constructor() {
        owner = msg.sender;
    }
    // Функция для изменения статуса пользователя
    function changeUserStatus(address _user, bool _newStatus) external onlyOwner {
        // Обновляем статус пользователя
        userStatus[_user] = _newStatus;
        // Вызываем событие для отслеживания изменения статуса
```

```
        emit StatusChanged(_user, _newStatus);
    }
}
```

Транзакции

Регистрация пользователя

```
from web3 import Web3
# Подключение к локальному узлу Ethereum или удаленному узлу
web3 = Web3(Web3.HTTPProvider('http://localhost:8545'))
# Адрес смарт-контракта UserRegistration
contract_address = ' 0x001d3f1ef827552ae1114027bd3ecf1f086ba0f9 '
contract_abi = [
    {
        "constant": False,
        "inputs": [{"name": "username", "type": "string"}],
        "name": "registerUser",
        "outputs": [],
        "payable": False,
        "stateMutability": "nonpayable",
        "type": "function",
    }
]
# Адрес кошелька отправителя
sender_address = '0x001d3f1ef827552ae1114027bd3ecf1f086ba0f9'
# Приватный ключ кошелька отправителя (не делитесь им!)
private_key = '0x021d3f1aef905552ae1165027bd3ecf1f116ba0f9'
# Создание объекта контракта
contract = web3.eth.contract(address=contract_address, abi=contract_abi)
# Подготовка данных для вызова функции контракта
username = 'Giocatory'
transaction_data = contract.functions.registerUser(username).buildTransaction({
    'from': sender_address,
    'gas': 1000000,
    'gasPrice': web3.toWei('30', 'gwei'),
    'nonce': web3.eth.getTransactionCount(sender_address),
})
# Подпись транзакции
signed_transaction = web3.eth.account.signTransaction(transaction_data, pri-
vate_key)
# Отправка транзакции
transaction_hash = web3.eth.sendRawTransaction(signed_transaction.rawTransac-
tion)
print(f'Transaction sent: {transaction_hash.hex()}')
```

Завершение курса обучения

```
from web3 import Web3
# Подключение к ноде Ethereum
```

```
w3 = Web3(Web3.HTTPProvider('http://localhost:8545'))
# Адрес кошелька отправителя
sender_address = '0x001d3f1ef827552ae1114027bd3ecf1f086ba0f9'
# Приватный ключ кошелька отправителя (не делитесь им!)
private_key = '0x021d3f1aef905552ae1165027bd3ecf1f116ba0f9'
# Адрес вашего смарт-контракта UserRegistration
contract_address = '0x1234567890123456789012345678901234567890'
contract_abi = [
    {
        "constant": False,
        "inputs": [{"name": "username", "type": "string"}],
        "name": "registerUser",
        "outputs": [],
        "payable": False,
        "stateMutability": "nonpayable",
        "type": "function",
    }
contract_abi = contract_abi
# Создание объекта контракта
contract = w3.eth.contract(address=contract_address, abi=contract_abi)
# Определение значения газа
gas_estimate = contract.functions.completeCourse().estimateGas()
# Подготовка данных для транзакции
transaction = contract.functions.completeCourse().buildTransaction({
    'from': sender_address,
    'gas': gas_estimate,
    'gasPrice': w3.toWei('50', 'gwei'),
    'nonce': w3.eth.getTransactionCount(sender_address),
})
# Подписание транзакции
signed_transaction = w3.eth.account.signTransaction(transaction, private_key)
# Отправка транзакции
transaction_hash = w3.eth.sendRawTransaction(signed_transaction.rawTransaction)
# Ожидание подтверждения транзакции
transaction_receipt = w3.eth.waitForTransactionReceipt(transaction_hash)
print(f'Transaction confirmed with status: {transaction_receipt["status"]}')
Запрос верификации навыков
from web3 import Web3
# Адрес вашего смарт-контракта
contract_address = '0x001d3f1ef827552ae1114027bd3ecf1f086ba0f9'
contract_abi = [
    {
        "constant": False,
        "inputs": [{"name": "username", "type": "string"}],
        "name": "registerUser",
```

```
        "outputs": [],
        "payable": False,
        "stateMutability": "nonpayable",
        "type": "function",
    }
contract_abi = contract_abi
# Адрес кошелька отправителя
sender_address = '0x001d3f1ef827552ae1114027bd3ecf1f086ba0f9'
# Приватный ключ кошелька отправителя (не делитесь им!)
private_key = '0x2a5bc342ed616b5baf827552ae1114027bd3ecf1f086ba0f9'
# Подключение к узлу Ethereum
web3 = Web3(Web3.HTTPProvider('https://mainnet.infura.io/v3/YourIn-
furaApiKey'))
# Создание экземпляра смарт-контракта
contract = web3.eth.contract(address=contract_address, abi=contract_abi)
# Создание транзакциитransaction = contract.functions.requestSkillVerifica-
tion().buildTransaction({
    'from': sender_address,
    'gas': 2000000,
    'gasPrice': web3.toWei('50', 'gwei'),
    'nonce': web3.eth.getTransactionCount(sender_address),
})
# Подписание транзакции
signed_transaction = web3.eth.account.signTransaction(transaction, private_key)
# Отправка транзакции
transaction_hash = web3.eth.sendRawTransaction(signed_transaction.rawTransac-
tion)
print(f'Transaction sent: {transaction_hash.hex()}')
Выпуск документа об образовании
from web3 import Web3
# Адрес вашего смарт-контракта
contract_address = '0x001d3f1ef827552ae1114027bd3ecf1f086ba0f9'
contract_abi = [
    {
        "constant": False,
        "inputs": [{"name": "username", "type": "string"}],
        "name": "registerUser",
        "outputs": [],
        "payable": False,
        "stateMutability": "nonpayable",
        "type": "function",
    }
contract_abi = contract_abi
# Адрес кошелька отправителя
sender_address = '0x001d3f1ef827552ae1114027bd3ecf1f086ba0f9'
```

```
# Приватный ключ кошелька отправителя (не делитесь им!)
private_key = '0x2a5bc342ed616b5baf827552ae1114027bd3ecf1f086ba0f9'
# ID студента и другие параметры для выпуска документа
student_id = 123
document_data = "Данные о документе"
# Подключение к узлу Ethereum
web3 = Web3(Web3.HTTPProvider('https://mainnet.infura.io/v3/YourIn-
furaApiKey'))
# Создание экземпляра смарт-контракта
contract = web3.eth.contract(address=contract_address, abi=contract_abi)
# Создание транзакции для выпуска документа
transaction = contract.functions.issueEducationDocument(student_id, docu-
ment_data).buildTransaction({
    'from': sender_address,
    'gas': 2000000,
    'gasPrice': web3.toWei('50', 'gwei'),
    'nonce': web3.eth.getTransactionCount(sender_address),
})
# Подписание транзакции
signed_transaction = web3.eth.account.signTransaction(transaction, private_key)
# Отправка транзакции
transaction_hash = web3.eth.sendRawTransaction(signed_transaction.rawTransac-
tion)
print(f'Transaction sent: {transaction_hash.hex()}')Формирование NFT из доку-
мента об образовании
// Импорт стандарта ERC721
import "@openzeppelin/contracts/token/ERC721/ERC721.sol";
import "@openzeppelin/contracts/access/Ownable.sol";
// Определение смарт-контракта
contract EducationNFT is ERC721, Ownable {
    // Счетчик для присвоения уникальных идентификаторов NFT
    uint256 private tokenIdCounter;
    // Маппинг для отображения идентификатора NFT на данные документа об
образовании
    mapping(uint256 => EducationDocument) private educationDocuments;
    // Структура для представления данных документа об образовании
    struct EducationDocument {
        string studentName;
        string courseName;
        uint256 completionYear;
        // Другие данные по вашему усмотрению
    }
    // Конструктор смарт-контракта
    constructor() ERC721("EducationNFT", "EDNFT") {}
    // Функция для создания NFT из сведений о документе об образовании
```

```
    function createEducationNFT(
        string memory _studentName,
        string memory _courseName,
        uint256 _completionYear
    ) external onlyOwner {
        // Увеличиваем счетчик идентификатора NFT
        tokenIdCounter++;
        // Создаем экземпляр данных документа об образовании
        EducationDocument memory newEducationDocument = EducationDocument({
            studentName: _studentName,
            courseName: _courseName,
            completionYear: _completionYear
            // Другие данные по вашему усмотрению
        });
        // Присваиваем данные документа об образовании идентификатору NFT
educationDocuments[tokenIdCounter] = newEducationDocument;
        // Создаем NFT с помощью функции из стандарта ERC721
        _safeMint(msg.sender, tokenIdCounter);
    }
    // Функция для получения данных документа об образовании по идентифика-
тору NFT
    function getEducationDocument(uint256 _tokenId) external view returns (Educa-
tionDocument memory) {
        return educationDocuments[_tokenId];
    }
}
```