

**Автономная некоммерческая организация высшего образования
«Университет Иннополис»**

**АННОТАЦИЯ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ
(БАКАЛАВРСКУЮ РАБОТУ)
ПО НАПРАВЛЕНИЮ ПОДГОТОВКИ
09.03.01 – «ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА»**

**НАПРАВЛЕННОСТЬ (ПРОФИЛЬ) ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ
«ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА»**

Тема

**Проектирование и реализация языка программирования
Elixirium: Elixir-подобный язык для виртуальной машины
Ethereum**

Выполнил

**Поздняков Никита Михайлович
Филонов Максим Андреевич**

ПОДПИСЬ

Иннополис, Innopolis, 2024

Оглавление

1	Введение	4
2	Обзор литературы	5
2.1	Подглава	6
2.1.1	Подподглава	6
3	Методология	7
4	Реализация	8
5	Выводы	9
	Список использованной литературы	10

Аннотация

В данной диссертации рассматривается необходимость создания нового языка программирования смарт-контрактов в экосистеме Ethereum, в которой в настоящее время доминируют контракты, написанные на Solidity. Это доминирование приводит к таким ограничениям, как монолингвизм¹ и монопарадигмизм². В нашем исследовании представлен Elixirium - функциональный язык с синтаксисом Elixir и динамической типизацией, предназначенный для повышения гибкости и креативности разработчиков децентрализованных приложений (dApp). Компилятор Elixirium был разработан и протестирован на совместимость с виртуальной машиной Ethereum (EVM) и способность поддерживать стандарты ERC-20 и ERC-721. Для сравнения потребления газа Elixirium и Solidity был использован комплексный набор тестов. Результаты показали, что, несмотря на функциональность и совместимость Elixirium с Ethereum, он потребляет больше газа, чем Solidity. Ключевые функции, такие как события и взаимодействие с хранилищем, были реализованы. Однако среди ограничений - высокий расход газа на развертывание, отсутствие внешних библиотек и взаимодействия с внешними смарт-контрактами. Будущая работа должна быть направлена на снижение затрат газа при развертывании и реализацию словарей, которые бы работали во время выполнения. Несмотря на высокое потребление газа, простота разработки и потенциал для более раннего развертывания Elixirium могут

¹Монолингвизм - Этот термин относится к использованию одного языка отдельным человеком или сообществом, без регулярного использования каких-либо дополнительных языков.

²Монопарадигмизм - Этот термин описывает использование только одной парадигмы программирования в среде программирования или программистом, исключая использование нескольких парадигм.

перевесить эти затраты, позиционируя его как жизнеспособную альтернативу в пространстве Web3.

Глава 1

Введение

Мяу [1]—[3]

Глава 2

Обзор литературы

With the widespread of computing systems, information processing, and net working, the practice of replacing paper documentation to electronic documentation has become more and more common. Electronic documentation within the workplace has several advantages over the traditional one, such as being easy to share, copy and edit the document. However, these advantages also present a problem. A malefactor, having access to the system, can easily copy and leak the document, without leaving any trace. Such actions are virtually undetectable in most systems, so, the malefactor goes unpunished. In this thesis, we propose one solution to the problem: digital watermarking.

Every electronic document within the protected system is marked with an invisible digital watermark, containing information about the user, accessing this particular document. Therefore, in case the protected company discovers the leaked document, they will be able to identify the machine of the malicious person and time when the document was leaked. This will allow inflicting punishment on the malefactor, recovering the costs of the leak, and potentially preventing future ones.

This description implies several essential properties of the task at hand:

1. Watermark must contain all necessary information, but still, be placeable and recognizable even on smaller images. The produced watermark must be compact but have the possibility to store enough information.
2. To prevent easy tampering, the watermark must be invisible to the naked eye (and, preferably, to basic image parsing tools). If malefactor does not know about the existence of watermark, they might not even try to remove it and disable it.

2.1 Подглава

Some text

2.1.1 Подподглава

Some text

Глава 3

Методология

Глава 4

Реализация

Глава 5

Выводы

Список использованной литературы

- [1] A. A. Aardvark, «Article title,» *Journal One*, т. 1, № 1, с. 1—8, 1900.
- [2] A. A. Abramson и В. В. Barbie, «Article title,» *Journal Two*, т. 1, № 1, с. 9—17, 1900.
- [3] A. A. Abramson, В. В. Barbie и С. С. Rider, «Article title,» *Journal Three*, т. 1, № 1, с. 192—244, 1900.