

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХЕРСОНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
Факультет комп'ютерних наук, фізики та математики
Кафедра комп'ютерних наук та програмної інженерії

**РОЗРОБКА КАЛЬКУЛЯТОРУ СТАВКИ ДЛЯ БЛОКЧЕЙН
СИСТЕМ**

Кваліфікаційна робота (проект)
на здобуття ступеня вищої освіти «бакалавр»

Виконав: здобувач
спеціальності: 121 Інженерія
програмного забезпечення
Освітньо-професійної (наукової)
програми: Інженерія програмного
забезпечення
Чобулда Данило Олександрович
Керівник: канд. пед. н., доцент
Вінник М. О.
Рецензент: доктор ф.-м. н.,
професор Савченко О. Г.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ	3
ВСТУП	4
РОЗДІЛ 1 КОНЦЕПТУАЛЬНА МОДЕЛЬ КАЛЬКУЛЯТОРУ СТАВКИ: ТЕОРЕТИЧНІ ОСНОВИ ТА ВІДОМОСТІ	5
1.1 Огляд технології блокчейн та її принципів роботи	5
1.2 Огляд ставки та її різновидів	15
1.3 Аналітичний підрозділ ставки: розрахунок реальної доходності	20
РОЗДІЛ 2 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ: КАЛЬКУЛЯТОР СТАВКИ	25
2.1. Огляд технологій, що використовуються у розробці калькулятора ставки	25
2.2. Архітектура додатку та схеми його компонентів	28
2.3. Опис розробки калькулятора ставки	30
ВИСНОВКИ	36
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	37
ДОДАТКИ	40
Додаток А	40

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

DeFi - Decentralized Finance

PHP - Hypertext Preprocessor

HTML - HyperText Markup Language

MVC – Model-View-Controller

ORM - Object-Relational Mapping

MySQL – My Structured Query Language

ВСТУП

Актуальність теми: калькулятор ставки є дуже актуальним в контексті сучасних тенденцій у галузі фінансів та блокчейн технологій. Розробка такого веб-додатку для блокчейн систем має важливе значення для фінансової сфери, так як дозволяє користувачам легко і швидко розрахувати свій потенційний дохід з вкладення коштів у блокчейн проекти.

Об'єкт дослідження: калькулятор ставки.

Предмет дослідження: технології створення веб-додатку.

Мета і завдання дослідження: метою дослідження є розробка програмного забезпечення, яке дозволить користувачам блокчейн системи зручно та ефективно розраховувати потенційний дохід від ставок. Завдання дослідження:

- 1) Аналіз функціональних та нефункціональних вимог до калькулятора ставки для блокчейн-систем.
- 2) Виявлення потреб та можливостей для створення калькулятора ставок.
- 3) Розробка технічної архітектури та програмного забезпечення.

РОЗДІЛ 1

КОНЦЕПТУАЛЬНА МОДЕЛЬ КАЛЬКУЛЯТОРУ СТАВКИ: ТЕОРЕТИЧНІ ОСНОВИ ТА ВІДОМОСТІ

1.1 Огляд технології блокчейн та її принципів роботи

Блокчейн (Blockchain) - це розподілена база даних, яка зберігає інформацію в блоках, зв'язаних між собою цифровим підписом. Це означає, що кожен блок містить інформацію про певну кількість транзакцій, а також хеш попереднього блоку, що забезпечує послідовність блоків у ланцюжку. Блокчейн можна порівняти з глобальним реєстром, який зберігає всі транзакції, які здійснюються у мережі[1].

Головна перевага блокчейн технології - це децентралізація, тобто відсутність центрального контролю та залежності від окремих осіб чи організацій. Кожен учасник мережі має доступ до всіх транзакцій, що здійснюються в мережі, та може перевірити їх автентичність та цілісність за допомогою криптографічних методів.

Блокчейн застосовується у різних галузях, включаючи фінансові послуги, медичну та логістичну галузі, громадські послуги та багато інших. Його використання дозволяє покращити безпеку, ефективність та надійність процесів, а також забезпечити децентралізацію та відкритість взаємодії між учасниками мережі.

Звичайний блокчейн - це суцільний ланцюжок блоків, що включає усі попередні блоки, кожен з яких посилається на попередній за допомогою хеш-функції. Кожен блок містить кілька елементів, включаючи заголовок, транзакції та підписи. У блокчейні немає централізованої управлінської структури, а кожен вузол (комп'ютер, що підключений до мережі) має копію ланцюжка. Вся мережа працює за протоколом

консенсусу, що дозволяє досягти згоди щодо правильності нових блоків, що додаються до ланцюжка.

Блокчейн забезпечує безпеку даних та їх надійність за допомогою криптографії, що дозволяє забезпечити конфіденційність та невідворотність транзакцій. Також блокчейн дозволяє створювати "розумні контракти", що є програмами, що автоматизують процеси, що пов'язані з переведенням активів між сторонами.

Блокчейн також має безліч застосувань у різних галузях, включаючи фінанси, логістику, медицину та багато інших. У сукупності з технологіями штучного інтелекту та Інтернету речей, блокчейн може допомогти створити нові екосистеми, що забезпечують безпеку, надійність та ефективність обміну даними і активами між різними сторонами[2].

Основні принципи роботи блокчейн-систем можна умовно розділити на такі:

- розподілена база даних;
- криптографічна безпека;
- децентралізована система;
- автоматичні смарт-контракти;
- надійність та стійкість до вторгнень.

Ці принципи є взаємопов'язаними та спільно забезпечують працездатність технології блокчейн. Кожен з них грає важливу роль у забезпеченні безпеки, надійності та функціональності мережі.

Перший принцип блокчейн технології - це "Розподілена база даних". Це означає, що дані, збережені в блокчейн, розподілені між багатьма вузлами (комп'ютерами) в мережі, а не зберігаються централізовано на одному сервері. Кожен вузол мережі має повну копію всіх даних, що зберігаються в блокчейні. Така система забезпечує безпеку та надійність

даних, оскільки жоден вузол не може змінити дані без згоди інших вузлів. Якщо один з вузлів стає несправним, інші вузли продовжують працювати і зберігають копії даних, тому їх не втрачається.

Крім того, така система забезпечує прозорість та відкритість, оскільки будь-який учасник мережі може перевірити всі дані, що зберігаються в блокчейні, і переконатися у їхній правильності. Це дозволяє підтверджувати легітимність транзакцій та підтримувати високу довіру до мережі.

Отже, розподілена база даних є важливим принципом блокчейн технології, який забезпечує безпеку, надійність, прозорість та відкритість мережі.

Другий принцип, на якому ґрунтується технологія блокчейн, це криптографічна безпека. Безпека є важливим аспектом у будь-якій системі, особливо в такій, яка працює з фінансовими транзакціями. Тому в блокчейні застосовується криптографічне шифрування для забезпечення конфіденційності, цілісності та автентичності даних.

Кожен блок містить у собі хеш-значення, яке є унікальним для кожного блоку і залежить від даних, що містяться в блоку. Це означає, що будь-які зміни даних в блоку автоматично відображаються в його хеш-значенні, що у свою чергу впливає на хеш-значення наступного блоку. Це створює ланцюжок хеш-значень, який забезпечує імутабельність даних, тобто унеможлиблює зміну даних, які вже були збережені в блоках.

Крім того, використання криптографічних алгоритмів дозволяє забезпечити конфіденційність даних, тому що транзакції зашифровуються з використанням публічних та приватних ключів. Публічні ключі використовуються для шифрування даних, тоді як приватні ключі використовуються для їх розшифрування. Це дозволяє

лише власнику приватного ключа здійснювати транзакції та забезпечує автентичність даних.

Таким чином, криптографічна безпека забезпечує захист від змін даних, їх крадіжки та несанкціонованого доступу до них, що робить блокчейн безпечним для зберігання та обміну фінансовими даними.[3]

Третій принцип, на якому базується робота блокчейн, - це децентралізація. Цей принцип полягає у тому, що мережа блокчейн не має централізованого керівництва. Замість цього, керування мережею здійснюється за допомогою протоколу, який забезпечує автоматичне виконання функцій, необхідних для роботи мережі. Це означає, що вся інформація про транзакції, блоки та стан мережі зберігається та перевіряється в різних вузлах мережі. Кожен вузол мережі може здійснювати операції з обробки транзакцій та створення нових блоків. На відміну від централізованих систем, де вся інформація зберігається на серверах, в мережі блокчейн кожен вузол мережі має свою власну копію бази даних.

Така система дозволяє забезпечити високий рівень безпеки та стійкості мережі, оскільки жоден вузол не є однією точкою входу для хакерів або зловмисників. Крім того, така децентралізація забезпечує певний рівень анонімності та приватності користувачів, оскільки немає централізованого джерела, яке може відстежувати транзакції або контролювати їх.

Однак, децентралізована система має свої недоліки. Наприклад, швидкість обробки транзакцій може бути повільною в порівнянні з централізованими системами, оскільки потрібен час на синхронізацію баз даних між вузлами мережі. Також, відсутність централізованого контролю може створити проблему при вирішенні конфліктів інтересів в мережі, якщо не передбачені відповідні механізми. Крім того,

децентралізована система може бути вразливою до атак більш потужних вузлів мережі, які можуть мати перевагу в обчислювальних можливостях і використовувати їх для здійснення атак на мережу.

Проте, не зважаючи на ці недоліки, децентралізована система має свої переваги. Зокрема, вона забезпечує більшу безпеку та незалежність від централізованого контролю, що дозволяє уникнути можливих помилок та порушень, що можуть стати причиною великих проблем у централізованих системах. Крім того, децентралізовані системи можуть бути більш стійкими до кібератак, оскільки немає одного центру, який може бути вразливим для атак.

Загалом, децентралізовані системи, такі як блокчейн, стають все більш популярними в різних сферах, таких як фінанси, технології, охорона здоров'я та багато інших, оскільки вони забезпечують більшу безпеку, прозорість та незалежність від централізованого контролю.

Четвертий принцип блокчейн - це використання автоматизованих смарт-контрактів. Смарт-контракти - це програми, які автоматично виконують умови, задані їх створювачем. Вони запускаються, коли виконуються певні умови, які можуть бути задані виключно точними алгоритмами.

Смарт-контракти дозволяють автоматизувати виконання угод, зменшуючи витрати на опосередкування та забезпечуючи більшу надійність, оскільки вони працюють на основі криптографічних протоколів і не можуть бути змінені без дозволу всіх учасників.

Смарт-контракти можуть бути використані для різних цілей, від автоматичного виконання угод між підприємствами до створення децентралізованих додатків (DApps). Вони можуть забезпечувати автоматичне розподілення коштів, перевірку ідентичності, безпечну передачу активів і багато іншого.

Проте, слід мати на увазі, що смарт-контракти також мають свої обмеження і можуть містити помилки в коді, які можуть призвести до некоректного виконання угод. Тому, перед використанням смарт-контрактів необхідно ретельно їх перевіряти і тестувати[4].

Останнім принципом блокчейн є надійність та стійкість до вторгнень. Блокчейн є досить надійною системою, оскільки будь-яка спроба змінити дані вже збережені в блокчейні стикається з технічними перешкодами. Кожен блок в блокчейні містить хеші попереднього блоку, що робить будь-які зміни в блоках попереднього ланцюжка майже неможливими. Більше того, такі зміни потребують великої обчислювальної потужності, що робить їх економічно не вигідними.

Крім того, блокчейн має досить високий рівень стійкості до вторгнень, оскільки кожен вузол мережі має свою власну копію бази даних, що зберігається на різних машинах в різних локаціях. Це ускладнює завдання хакерам, які намагаються здійснити атаку на мережу, оскільки для успішної атаки потрібно одночасно скомпрометувати більше ніж половину вузлів мережі.

Для забезпечення ще більшої безпеки та захисту від вторгнень, деякі блокчейн-мережі використовують додаткові заходи безпеки, такі як мультипідписи та розумні контракти. Мультипідписи вимагають підпису кількох користувачів для здійснення транзакцій, тоді як розумні контракти є автоматизованими програмами, які забезпечують виконання умов транзакцій, що робить їх більш безпечними та надійними.

Існує кілька типів блокчейн систем, кожен з яких має свої унікальні особливості та застосування. Основні типи блокчейн систем:

- публічний блокчейн;
- приватний блокчейн;
- консорціумний блокчейн;

- гібридний блокчейн[5].

Публічний блокчейн - це вид блокчейну, доступний для всіх користувачів мережі із відкритим вихідним кодом. Це означає, що будь-який користувач може бути учасником мережі, створювати нові блоки, переводити кошти та перевіряти транзакції. В цьому виді блокчейну немає центральної влади, яка контролює мережу, і кожен учасник є рівноправним. Цей вид блокчейну використовується для створення криптовалют, таких як Bitcoin та Ethereum, а також для реалізації інших децентралізованих проектів. Оскільки публічний блокчейн відкритий для всіх, він забезпечує високий рівень довіри, оскільки будь-який користувач може перевірити транзакції та переконатись у їх справедливості.

Проте публічний блокчейн також має свої недоліки. Один з них - повільна швидкість обробки транзакцій, оскільки кожна транзакція має пройти через багато вузлів мережі, перш ніж бути підтвердженою. Крім того, він може бути вразливим до атак 51%, коли одна група користувачів здобуває контроль над більшістю вузлів мережі та може контролювати транзакції та відправлення коштів від імені інших користувачів.

Незважаючи на ці недоліки, публічний блокчейн залишається одним з найбільш популярних інструментів для реалізації децентралізованих проектів, які забезпечують високий рівень довіри та безпеки у мережі.

Приватний блокчейн - це блокчейн, який призначений для внутрішнього використання певною організацією або компанією. Цей вид блокчейну відрізняється від публічного блокчейну тим, що доступ до нього обмежений тільки для визначеного кола користувачів. Також, у приватному блокчейні зазвичай не існує токенів, які можуть бути виторговані на біржі.

Приватний блокчейн може бути корисним для компаній або організацій, які хочуть використовувати технологію блокчейну для внутрішньої обробки даних, контролю за транзакціями та підтвердження прав на доступ до інформації. Він також може бути використаний в умовах, коли важливо забезпечити конфіденційність даних та захист від зловживань або вторгнень.

Однак, приватний блокчейн може мати свої недоліки. Наприклад, обмежене коло користувачів може призвести до недостатньої децентралізації, що може знизити ступінь надійності мережі. Також, може виникнути проблема управління блокчейном в разі, якщо він використовується декількома сторонами, і виникають конфлікти між ними.

Третій тип блокчейну, який називається консорціумним блокчейном, поєднує особливості публічного та приватного блокчейнів. У цьому типі мережі доступ до блокчейну обмежений групою учасників, які погодилися на певні правила та умови співпраці. Ця група може складатися з організацій або осіб з певної індустрії, які мають спільний інтерес у використанні блокчейну.

Консорціумний блокчейн забезпечує більш високий рівень конфіденційності, оскільки доступ до мережі мають тільки погоджені учасники, що дозволяє зберігати конфіденційну інформацію та даних від сторонніх осіб. Крім того, такий тип блокчейну дозволяє керувати правами доступу до даних, що дає можливість керувати рівнем доступу до інформації відповідно до ролей та обов'язків учасників мережі.

Консорціумний блокчейн може бути використаний у різних галузях, таких як фінанси, логістика, охорона здоров'я та інші. Наприклад, у фінансовій галузі консорціумний блокчейн може бути використаний для створення системи для обміну валют між банками, що забезпечує більш

ефективний та безпечний процес обміну валют. У логістичній галузі блокчейн може допомогти забезпечити більш точну та ефективну доставку товарів, а в галузі охорони здоров'я - забезпечити безпеку та цілісність медичної інформації.

Однак, консорціумний блокчейн також має свої недоліки. По-перше, у такій системі існує потенційна загроза зловживання владою з боку групи учасників, які контролюють більшість голосів. По-друге, процес прийняття рішень може бути повільнішим у порівнянні з публічним блокчейном, оскільки потрібно досягти консенсусу між всіма учасниками. Також, консорціумні блокчейни можуть бути менш відкритими та прозорими, оскільки правила та умови участі можуть бути встановлені обмеженою групою учасників.

Проте, консорціумний блокчейн може бути більш ефективним та безпечним рішенням для бізнес-середовища, де необхідно дотримуватись законодавства щодо обмеження доступу до конфіденційної інформації, але при цьому забезпечувати можливість декількох сторін працювати разом над однією базою даних. Також, консорціумні блокчейни можуть бути більш енергоефективними, оскільки не потребують такої великої кількості обчислювальної потужності, як публічні блокчейни[6].

Гібридний блокчейн - це комбінація публічного та приватного блокчейнів. Він забезпечує можливість розподілу даних між різними членами мережі, але водночас може контролювати рівень доступу до цих даних. Така гнучкість дозволяє використовувати гібридний блокчейн у різних сферах.

У гібридному блокчейні, приватна частина може бути використана для збереження конфіденційної інформації, такої як особисті дані чи фінансова інформація, тоді як публічна частина може бути використана

для збереження публічної інформації, такої як транзакції або інформація про товари.

Інша важлива особливість гібридного блокчейну - це можливість зміни протоколу консенсусу. Це означає, що в різних частинах мережі можуть бути різні протоколи консенсусу, в залежності від їх потреб. Наприклад, приватна частина мережі може використовувати протокол консенсусу Proof-of-Authority, який дозволяє швидко підтверджувати транзакції, тоді як публічна частина може використовувати більш складний протокол консенсусу Proof-of-Work.

Гібридний блокчейн може бути використаний у сферах, де потрібні як переваги публічного, так і приватного блокчейнів. Наприклад, у сфері логістики гібридний блокчейн може забезпечити надійність та безпеку даних про перевезення вантажів між різними компаніями, а також дозволити публічний доступ до інформації про рух вантажу під час транспортування.

Гібридний блокчейн також може бути використаний у медичній сфері для збереження та обміну медичних даних. Приватний блокчейн може забезпечити конфіденційність пацієнтів, тоді як публічний блокчейн може забезпечити надійність та доступність даних для медичних працівників.

Однак, гібридний блокчейн також має свої недоліки, такі як складність розробки та підтримки, а також питання щодо безпеки та прозорості даних, які зберігаються в різних частинах блокчейну. Також, використання гібридного блокчейну може призвести до збільшення витрат на інфраструктуру та обслуговування.

1.2 Огляд ставки та її різновидів

Ставка - це механізм винагороди, який стимулює користувачів зберігати свої монети або токени на певний час, щоб захистити мережу і підтвердити транзакції. Це можна вважати схожим на майнінг, але без потреби в спеціалізованому обладнанні; все, що вам потрібно - це монети, які ви готові тримати протягом певного періоду часу. Чим довше ви тримаєте монети, тим вищим буде відсоток вашої винагороди[7]

Як було сказано вище, ставка передбачає блокування вашої криптовалюти в якості застави в обмін на отримання більшої кількості тих самих монет/токенів.

Ставка дозволяє власникам криптоактивів отримувати пасивний дохід без необхідності переводити свої активи з біржі.

Після закінчення зазначеного в контракті часу стейкінг-платформа переведе вкладені кошти в спотове сховище. Хоча вкладені монети захищені, їхня вартість все одно може знизитися через волатильність обмінного курсу. Крім того, дохідність може змінюватися через волатильність криптовалют.

Для пересічного користувача ставку можна порівняти з банківською інвестицією. Ви переказуєте кошти на рахунок і отримуєте пасивний дохід, зберігаючи їх. Чим більше криптовалют ви покладете на рахунок, тим більшим буде прибуток.

Деякі блокчейни також мають спеціальні умови ставки на додаток до цієї базової схеми роботи.

Існують три види ставки:

- фіксована ставка.
- безстрокова ставка.

- DeFi-ставка[8].

Фіксована ставка - це процес блокування валюти або токенів на певний період для отримання винагороди в мережі блокчейн. Цей процес дозволяє учасникам мережі використовувати свої активи для забезпечення безпеки та стабільності мережі, отримуючи винагороду за свою участь.

Існують є два варіанти фіксованої ставки:

- Стейк монет/токенів на Proof of Stake (PoS). Як правило, прибуток, отриманий від ставки, прямо пропорційний кількості стейкннутих токенів. Користувачі, які вкладають більшу кількість токенів, швидше за все, отримають більший прибуток.
- Стейк інших монет. Вкладаючи токени, не пов'язані з PoS, наприклад, для забезпечення ліквідності або маржі, користувачі можуть отримувати прибуток. Однак, як правило, відсоток винагороди за такі монети нижчий у порівнянні з класичною ставкою PoS, де прибуток залежить від кількості внесених токенів[9].

Фиксированный стейкинг

Токен	Расч. доходность % годовых	Срок (дней) ⌄	Минимальная сумма	
NEAR	54.49%	10 30 60 90	1NEAR	Добавить активы
CTSI	65.25%	10 30 60 90	100CTSI	Добавить активы
AXS	152.18%	10 30 60 90	0.1AXS	Продано Проверить
PNT	18.56%	30 60 90	1PNT	Добавить активы
LUNA	3.85%	30 60 90	1LUNA	Добавить активы

Рис. 1.1 – Фіксована ставка

Однією з основних переваг фіксованої ставки є стимулювання учасників мережі зберігати свої токени або валюту в мережі на довгий термін. Це може допомогти знизити ризики для мережі, оскільки це

забезпечує більш стабільний обсяг активів, які не будуть викинуті на ринок, знижуючи ризики для спекуляцій.

Окрім того, фіксована ставка може допомогти підтримувати ціну токена або валюти, оскільки чим більша кількість активів блокується в мережі, тим менше їх є доступно на вільному ринку. Це може допомогти запобігти зниженню ціни через перевантаження ринку з надлишковими активами.

Однак, існують певні ризики для учасників, які заблоковують свої активи в мережі. Основним ризиком є можливість зменшення вартості токенів або валюти під час періоду блокування. Якщо ціна токенів спаде, то винагорода за ставку може не компенсувати втрату вартості активів. Також, фіксована ставка може бути менш гнучкою в порівнянні з іншими видами, оскільки активи блокуються на довший період.

У будь-якому випадку, фіксована ставка може бути вигідною для учасників мережі, які мають значну кількість монет і хочуть забезпечити безпеку мережі.

Безстрокова ставка, як і фіксована, є однією з видів ставки в блокчейні. Цей вид ставки означає, що учасник блокчейн мережі зобов'язується зберігати свої монети в мережі на необмежений термін. За внесення ставки він отримує винагороду у вигляді криптовалюти, яку він ставкував.

Однією з переваг безстрокової ставки є те, що учасники мережі не обмежені терміном зберігання своїх монет. Це дозволяє їм зберігати монети у мережі на необмежений термін, забезпечуючи стабільність мережі та отримуючи при цьому винагороду за внесену ставку.

Іншою перевагою безстрокової ставки є те, що він дозволяє уникнути ризику втрати винагороди за внесену ставку, що може статися при

фіксованій ставці, якщо учасник забере свої монети з мережі раніше визначеного терміну.

Звичайно, безстрокова ставка не є ідеальною і має свої недоліки. Наприклад, учасник мережі не може використовувати свої монети, якщо він зберігає їх у мережі для ставки на необмежений термін. Крім того, він не може продавати або обмінювати свої монети, поки вони знаходяться у мережі для ставки.

DeFi-ставка (Decentralized Finance staking) - це процес, коли користувачі блокчейн мережі можуть ставити свої криптовалюти в децентралізовані фінансові протоколи (DeFi-протоколи) з метою отримання прибутку[10].

Децентралізовані фінансові протоколи дозволяють ставити гроші у депозитні пули, а також брати участь в механізмах ліквідності, які дозволяють обмінювати криптовалюти без посередницької участі банків чи інших фінансових установ. Зазвичай, користувачі отримують винагороду за ставки від комісій, що сплачуються учасниками мережі за доступ до фінансових послуг.

Для того, щоб взяти участь в DeFi-ставці, користувачі повинні знати, які DeFi-протоколи є доступними для ставок, а також, які криптовалюти можуть бути використані для ставок. Зазвичай, для DeFi-ставки використовують популярні криптовалюти, такі як Bitcoin, Ethereum, а також стейблкоїни, такі як USDT та USDC.

Одним з найбільш відомих DeFi-протоколів є Compound, де користувачі можуть ставити свої криптовалюти у депозитні пули та отримувати винагороду за ставки. Іншими відомими DeFi-протоколами є Aave, MakerDAO, Curve Finance та інші.

Однією з переваг DeFi-ставки є її децентралізований характер, який забезпечує безпеку та надійність транзакцій, а також зменшує можливість

фінансових обманів чи втручання управління з боку централізованих установ. Крім того, якщо порівняти звичайну і DeFi-ставку, остання може бути більш вигідною. У DeFi-ставці ви можете отримати вищу дохідність, оскільки тут зазвичай застосовуються вищі процентні ставки. Крім того, DeFi-ставка може бути більш гнучкою і надавати більше можливостей для диверсифікації портфеля.

Однак, DeFi-ставка має свої ризики, такі як висока волатильність ринку та можливість втрати коштів у разі виникнення помилок у смарт-контрактах. Тому, як і в будь-якому іншому вкладенні, важливо розуміти ризики та проводити додаткову досліджувану роботу, перш ніж інвестувати свої кошти.

Токен	Расч. APR	Срок (дней)	Минимальная сумма	
LTC	1.4%	Гибкая блокировка	0.001LTC	Добавить активы
XRP	1.39%	Гибкая блокировка	0.001XRP	Добавить активы
BUSD	13.33%	Гибкая блокировка 120	0.01BUSD	Добавить активы
ETH	10.12%	Гибкая блокировка 120	0.001ETH	Добавить активы
BTC	8.19%	Гибкая блокировка 60	0.0001BTC	Добавить активы
DYDX	16.39%	30	0.1DYDX	Добавить активы

Рис. 1.2 – DeFi ставка

Ставка залишається прибутковим способом заробітку на криптовалютах у 2023 році з низькими ризиками. Однак кінцева прибутковість залежить від конкретної обраної криптовалюти і тривалості зберігання монет у стейкінг-гаманці. Таким чином, прибутковість може відрізнитися від користувача до користувача: одні заробляють менше, а інші – більше[11]

1.3 Аналітичний підрозділ ставки: розрахунок реальної доходності

Аналіз фактичного прибутку від ставки є важливим етапом для тих, хто бажає інвестувати свої кошти у ставку. Цей аналіз дає можливість оцінити потенційний дохід, який може отримати інвестор. Для аналізу фактичного прибутку від ставки необхідно враховувати ряд факторів, таких як процентна ставка, період ставки, кількість власних токенів та вартість токенів на ринку.

Процентна ставка - це процент, який виплачується за ставку валюти. Цей процент може змінюватись в залежності від вибраного протоколу ставки та кількості власних токенів, що були заблоковані.

Період ставки - це час, протягом якого токени знаходяться у стейкінгу. Час може бути фіксованим або безстроковим. Зазвичай, чим довший період стейкінгу, тим вищий процентна ставка.

Кількість власних токенів - це кількість токенів, які власник заблокував для ставки. Чим більше токенів заблоковано, тим більший прибуток можна отримати.

Вартість токенів на ринку - це ціна, за якою токени можна купити або продати на ринку. Вартість токенів може змінюватись, що впливає на фактичний дохід від ставки.

За допомогою аналізу фактичного прибутку можна оцінити потенційну вигоду від ставки та зробити розумний вибір щодо інвестування своїх коштів.

Обрахунок номінальної та реальної ставок є важливим етапом у розрахунку фактичного прибутку від стейкінгу. Номінальна ставка - це процентна ставка, яка встановлюється для визначення прибутку від стейкінгу. Наприклад, якщо номінальна ставка становить 10%, а ви вклали 1000 доларів, то ви отримаєте 100 доларів за рік.

Проте, реальна ставка може відрізнятись від номінальної ставки, оскільки враховується інфляція. Якщо інфляція становить 3%, то реальна ставка буде меншою за номінальну ставку, тому що ваші гроші втратять на вартості у результаті інфляції. У такому випадку, реальна ставка може бути приблизно 7% замість 10%.

Обрахувати реальну ставку можна за формулою: реальна ставка = $((1 + \text{номінальна ставка}) / (1 + \text{інфляція})) - 1$. Наприклад, якщо номінальна ставка дорівнює 10%, а інфляція - 3%, то реальна ставка буде: реальна ставка = $((1 + 0.10) / (1 + 0.03)) - 1 = 6.80\%$.

Отже, реальна ставка в цьому прикладі буде 6.80%, що менше, ніж номінальна ставка, оскільки враховується вплив інфляції на вартість вкладених коштів. Для точнішого обрахунку фактичного прибутку від стейкінгу необхідно враховувати як номінальну, так і реальну ставки[12].

Staking Rewards, популярний сайт-агрегатор, пропонує широкий вибір номінальних ставок. Крім того, важливо враховувати комісію, яку стягує стейкінгова платформа. Якщо рік тому лише кілька спеціалізованих сайтів пропонували послуги ставки, то зараз на цей ринок вийшли такі великі криптовалютні біржі, як Binance, Poloniex та Waves.Exchange[13].









 IOST IOST	\$ 0.00542 (-7.99%)	10.20%	6.20%
 Waves WAVES	\$ 1.19 (-7.03%)	5.72%	2.64%
 Zcoin XZC	\$ 4.56 (-4.60%)	12.95%	-13.79%
 V Systems VSYS	\$ 0.0241 (-6.28%)	12.87%	7.78%
 Alon AION	\$ 0.1157 (-8.32%)	5.88%	3.63%
 Harmony ONE	\$ 0.00366 (-7.13%)	11.47%	4.91%
 Ark ARK	\$ 0.2524 (2.97%)	8.74%	3.46%
 Eirond ERD	\$ 0.00281 (-3.64%)	29.76%	29.76%

Рис. 1.3 – Список номінальних ставок

Розглянемо фактичні дані для 5 найпопулярніших криптовалют за період з 1 грудня 2021 року по 1 червня 2022 року. Ми розглянемо лише номінальну прибутковість монети та коливання цін, не беручи до уваги комісійні платформи.

1) Cosmos (ATOM)

"Інтернет блокчейнів" - це основна концепція цього проекту. Протокол має на меті досягти інтероперабельності між різними блокчейнами, що є значним викликом у криптоіндустрії. У разі успіху ціна монети могла б злетіти до небес, але у 2020 році спостерігалася негативна тенденція. Маючи високу капіталізацію, АТОМ є другою за величиною стейкінг-монетою, доступною на ринку. Протокол працює, але є ще багато питань, які потребують вирішення. АТОМ можна стейкувати на декількох платформах.

Фактична прибутковість

$$P = 0,7326$$

Номінальна дохідність: 8,34%

$$K = 1,0834$$

$$ROI = 0,7937 \text{ (збиток у розмірі } 20,6\%)$$

2) Tezos (XTZ)

XTZ - єдиний стейблкоїн, який входить до топ-10 за капіталізацією, в його стейкінг інвестовано понад 2 мільярди доларів. Tezos став відомим завдяки залученню понад 230 мільйонів доларів під час ІСО. Унікальною особливістю цього протоколу є те, що стейкхолдери, або власники монет, можуть змінювати його правила за допомогою децентралізованого процесу голосування, що дозволяє блокчейну розвиватися без хардфорків. Майже всі стейкінгові платформи підтримують XTZ.

Фактична прибутковість

$$P = 2,23$$

Номинальна дохідність: 5,65%

$$K = 1,0565$$

$$ROI = 2,356 \text{ (прибуток у розмірі 135,6\%)}$$

3) USD Neutrino (USDN)

Стейблкоїн - єдиний стейкінговий актив серед популярних. Вартість USDN прив'язана до долара США з незначними відхиленнями, і він повністю забезпечений активом WAVES. Тому питання волатильності не стоїть: інвестори отримують стабільний щоденний дохід від стейкінгу USDN. Однак норма прибутковості не є фіксованою і залежить від динаміки активу WAVES (який за 6 місяців виріс на 85%) і частки від загальної кількості монет USDN, інвестованих в стейкінг. За даними Waves.Exchange, середній прибуток складе 10%, в той час як KuCoin заявляє про реальну прибутковість на рівні 8-15%. Крім того, Waves.Exchange дозволяє інвесторам купувати монети USDN без комісії з банківської картки, що робить її привабливою платформою для деяких.

Фактична прибутковість

$$P = 1$$

Номинальна дохідність: 10,1%

$$K = 1,101$$

$$ROI = 1,101 \text{ (прибуток у розмірі 10,1\%)}$$

4) Synthetix (SNX)

Synthetix працює як платформа на базі Ethereum, що дозволяє створювати та обмінюватися синтетичними активами. Ці синтетичні

активи являють собою токенизовані версії різноманітних товарів, таких як акції, валюти, криптовалюти, дорогоцінні метали та фондові індекси. З більш ніж 40 торговими парами, Synthetix пропонує широкий вибір варіантів, включаючи синтетичне золото, срібло, долар США, швейцарський франк та інші.

Фактична прибутковість

$$P = 0,6139$$

Номинальна дохідність: 53,66%

$$K = 1,5366$$

$$ROI = 0,9433 \text{ (збиток у розмірі 5,7\%)}$$

5) V.Systems (VSYS)

V.Systems - це інфраструктурний проект, метою якого є розробка баз даних і хмарних сервісів для ринку децентралізованих фінансів (DeFi), хоча повноцінного продукту ще не запущено. Цікаво, що засновником проекту є Сонні Кінг, видатний теоретик блокчейну, який створив протокол Proof-of-Stake.

Фактична прибутковість

$$P = 0,471$$

Номинальна дохідність: 12,89%

$$K = 1,1289$$

$$ROI = 0,532 \text{ (збиток у розмірі 46,8\%)}$$

РОЗДІЛ 2

РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ: КАЛЬКУЛЯТОР СТАВКИ

2.1. Огляд технологій, що використовуються у розробці калькулятора ставки

PHP - це мова програмування, яка використовується для розробки веб-додатків із динамічним контентом. Ця мова дозволяє взаємодіяти з базами даних, зчитувати і записувати файли, створювати PDF-файли, відправляти електронні листи та інше.

Основні особливості PHP:

- PHP є мовою з відкритим кодом, тобто вона є безкоштовною та доступною для всіх;
- Ця мова є легкою в освоєнні і має велику кількість документації та прикладів коду;
- PHP є популярною мовою для розробки веб-додатків, тому вона має велику спільноту розробників та безліч корисних бібліотек;
- PHP може бути вбудований в HTML-код, що дозволяє створювати динамічні сторінки змінюючи HTML-код на льоту.

PHP підтримує об'єктно-орієнтоване програмування, що дозволяє створювати класи, об'єкти та методи. Вона також підтримує анонімні функції, замикання та ітератори. PHP може взаємодіяти з багатьма іншими технологіями, такими як AJAX, XML та JSON, що робить її популярним вибором для веб-розробки.

Основними перевагами PHP є простота використання, велика кількість функцій та бібліотек, а також широка підтримка серед веб-хостинг-провайдерів.

PHP дуже гнучка мова програмування, що дозволяє створювати різні веб-додатки. Для розробки калькулятора ставки ми будемо використовувати PHP для обчислення прибутку та інших розрахунків, що потрібні для нашого додатку[14].

Symfony - це відкрите програмне забезпечення, яке надає ряд інструментів і бібліотек для розробки веб-додатків на PHP. Він базується на концепції Model-View-Controller (MVC) та використовує ряд різноманітних компонентів, таких як роутер, контролер, ORM та багато інших, які дозволяють розробникам створювати високоякісні веб-додатки швидко та ефективно.

Symfony має широкий спектр інструментів та можливостей, які роблять його одним з найбільш популярних фреймворків PHP. Деякі з його основних переваг:

- Він має потужний роутер, який дозволяє налаштовувати маршрутизацію веб-запитів.
- Symfony пропонує широкий набір компонентів та бібліотек, які можна використовувати окремо або разом.
- Він має потужну ORM (Object-Relational Mapping), яка дозволяє працювати з базами даних швидко та ефективно.
- Symfony пропонує багато інструментів для тестування веб-додатків, що дозволяє розробникам впевнено виконувати рефакторинг коду та виявляти помилки.

Узагальнюючи, Symfony є потужним та гнучким фреймворком, який надає багато можливостей для розробки веб-додатків на PHP. Його широкий спектр інструментів та бібліотек дозволяє розробникам швидко та ефективно створювати високоякісні веб-додатки[15]

MySQL - це відкрита реляційна система управління базами даних, яка використовує мову SQL для управління даними. Вона є однією з найбільш популярних баз даних, що використовується у веб-розробці.

MySQL забезпечує можливість створення баз даних та таблиць, додавання, видалення і редагування даних, а також запитів до даних, використовуючи мову SQL. Вона є дуже надійною та швидкою базою даних, що може обробляти велику кількість даних.

MySQL підтримує множину типів даних, таких як числа, рядки, дати та багато інших. Вона також має можливість створювати зв'язки між таблицями, що забезпечує зручний доступ до даних.

У розробці веб-додатків MySQL зазвичай використовується для зберігання даних, таких як інформація про користувачів, їх транзакції та інші дані, пов'язані з додатком. MySQL є стандартом для багатьох веб-додатків та інших проектів у сфері веб-розробки[1617].

JavaScript є скриптовою мовою програмування, яка використовується для створення динамічного контенту на веб-сторінках. Вона може бути використана для додавання ефектів, анімацій, валідації даних, обробки подій та багато іншого. У веб-розробці JavaScript часто використовується разом з HTML та CSS для створення інтерактивних сторінок та додавання функціональності до веб-додатків.

Основні фреймворки JavaScript включають React, Angular та Vue.js. Ці фреймворки дозволяють легко створювати інтерактивні інтерфейси та додавати складну логіку до веб-додатків.[17

У нашому калькуляторі ставки ми можемо використати JS для взаємодії з користувачем та динамічної зміни відображення результатів обчислень. Наприклад, ми можемо створити кнопки для додавання або видалення стейкінг пулів, валідувати введені користувачем дані та

автоматично розраховувати результати на стороні клієнта без необхідності перезавантаження сторінки.

2.2. Архітектура додатку та схеми його компонентів

Для розробки веб додатку була обрана архітектура Model-View-Controller(MVC).

Цей тип досить поширений для веб-додатків, і він добре підходить для калькулятора ставки.

У MVC архітектурі додаток розділений на три основні компоненти:

1. Model (Модель) - це компонент, який відповідає за дані, з якими працює додаток. У випадку калькулятора ставки модель буде відповідати за розрахунок прибутку від стейкінга на основі введених користувачем даних. Вона також може зберігати дані про користувача і його стейкінг операції.
2. View (Вид) - це компонент, який відповідає за відображення даних на сторінці. У випадку калькулятора ставки він буде відповідати за відображення форми введення даних та результатів розрахунку.
3. Controller (Контролер) - це компонент, який відповідає за обробку запитів користувача та взаємодію з моделлю та виглядом. У випадку калькулятора ставки він буде відповідати за обробку даних, введених користувачем, та передачу їх в модель для розрахунку. Крім того, контролер може взаємодіяти з видом, щоб відображати результати розрахунку користувачу.

Загалом, архітектура MVC дозволяє зручно організувати код додатку та розділити його на логічні компоненти. Це сприяє більш ефективному та швидкому розробленню, а також полегшує підтримку та розширення додатку в майбутньому.[18]

Діаграма класів для калькулятора ставки, побудованої на основі архітектури MVC:

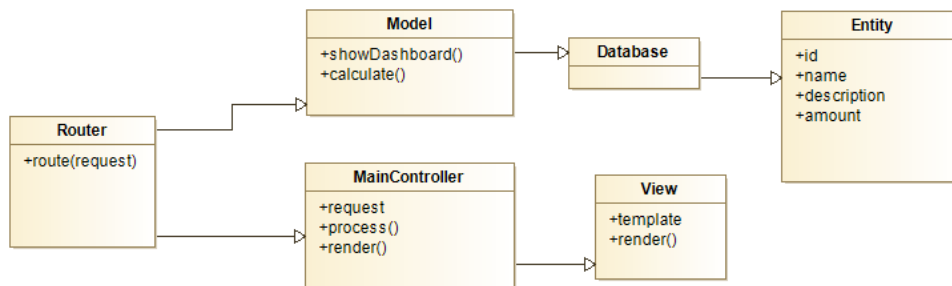


Рис 2.1 – Діаграма класів

Ця діаграма показує, що є чотири головні компоненти додатку: Router, Controller, Model та View, які взаємодіють між собою для обробки запитів користувачів та повернення відповіді.

Також на діаграмі можна побачити, що Model використовує базу даних (Database), а саме таблицю Entity, яка містить дані про мережу стейкінгу.

Діаграма активності для обчислення ставки на калькуляторі:

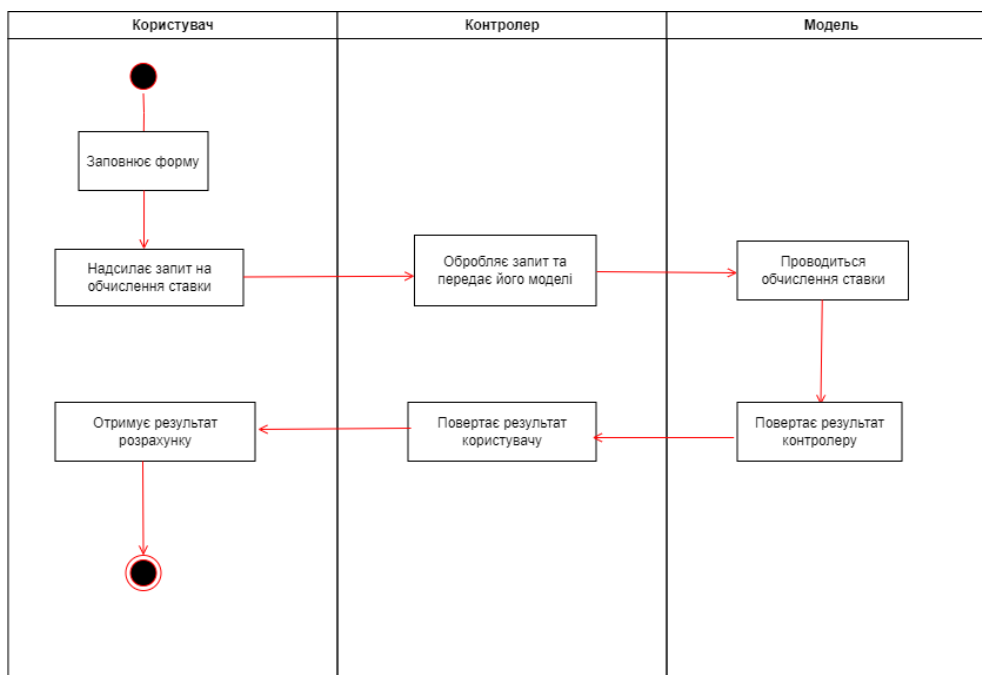


Рис. 2.2 – Діаграма активності

У цій діаграмі користувач відкриває сторінку калькулятора та надсилає запит на обчислення ставки. Контролер приймає запит, передає його до моделі, де проводиться обчислення, і повертає результати контролеру. Контролер повертає результати користувачеві.

2.3. Опис розробки калькулятора ставки

Список етапів розробки веб-додатку:

- 1) Налаштування середовища розробки та встановлення необхідних залежностей.
- 2) Створення базової структури додатку на основі Symfony.
- 3) Розробка моделі для калькулятора, яка міститиме логіку розрахунку ставки.
- 4) Розробка контролеру для обробки запитів користувачів та передачі даних в модель.
- 5) Розробка шаблонів вигляду (views) для відображення результатів розрахунків та форм для введення даних.
- 6) Налаштування маршрутів для коректної обробки запитів.
- 7) Тестування додатку та відлагодження помилок.
- 8) Оптимізація та покращення продуктивності додатку[19].

Перший етап розробки калькулятора полягає в налаштуванні середовища розробки та встановленні необхідних залежностей. Це дуже важливий етап, оскільки без правильного налаштування середовища розробки та встановлення необхідних залежностей подальша розробка може бути ускладненою.

Серед основних кроків цього етапу можна виділити:

- 1) Встановлення та налаштування локального веб-сервера (наприклад, Apache чи Nginx) та СУБД (наприклад, MySQL чи

PostgreSQL). Це дозволить запустити та тестувати додаток локально на комп'ютері.

- 2) Встановлення та налаштування залежностей проекту, таких як фреймворки (наприклад, Symfony), бібліотеки та пакети, які необхідні для розробки калькулятора.
- 3) Створення проекту та налаштування конфігурації, включаючи налаштування з'єднання з СУБД, розташування файлів, встановлення параметрів проекту тощо.

Ці кроки допоможуть забезпечити правильне налаштування середовища розробки та встановити всі необхідні залежності для подальшої розробки калькулятора.

Другий етап розробки полягає в створенні базової структури додатку на основі Symfony, яка буде використовуватися протягом всієї подальшої розробки. Основними завданнями на цьому етапі є:

- 1) Створення нового проекту Symfony. Для цього використовується команда “`symfony new`”, яка створює новий проект з необхідними налаштуваннями та структурою директорій.
- 2) Налаштування базової конфігурації додатку. У файлі “`config/packages/framework.yaml`” встановлюються основні параметри фреймворку, такі як шлях до кешування, режим відладки, мова за замовчуванням, інтервал часу сесії та інші.
- 3) Створення базових шаблонів. У папці “`templates`” створюються базові шаблони, які будуть використовуватися для створення сторінок додатку. Ці шаблони містять загальну структуру HTML-сторінки та посилання на необхідні CSS- та JS-файли.
- 4) Додавання бібліотек та компонентів. Symfony має велику кількість готових бібліотек та компонентів, які можна використовувати у додатку. Наприклад, для роботи з формами

можна використовувати компонент “Form”, а для роботи з БД – бібліотеку “Doctrine”.

- 5) Створення базових контролерів. У папці “src/Controller” створюються базові контролери, які будуть відповідати за обробку запитів та відображення відповідних сторінок.

Після виконання цих завдань, ми отримуємо базову структуру додатку, яка включає в себе деякі згенеровані Symfony файли та папки, такі як каталог “config” з налаштуваннями додатку, каталог “src” з базовими контролерами та класами моделей, каталог “templates” з шаблонами відображення, та каталог “public” з файлами ресурсів[20].

На третьому етапі необхідно розробити модель для калькулятора. Модель буде містити всю логіку розрахунку ставки та розрахунку прибутку. Для цього можна створити окремий клас, який буде містити методи для розрахунку.

Наприклад, для розрахунку прибутку можна створити метод, який буде приймати на вхід поточну ціну монети, кількість монет, що зберігаються від користувача, а також різні параметри стейкінгу, такі як ставка, термін стейкінгу тощо. Після цього метод буде розраховувати прибуток та повертати його.

Крім того, модель може містити методи для розрахунку ставки, перевірки коректності введених даних тощо.

Четвертий етап передбачає розробку контролера, який буде взаємодіяти з моделлю калькулятора та обробляти запити користувачів. Контролери в Symfony зазвичай реалізуються як PHP-класи, які унаслідуються від базового контролера Symfony - Controller.

У цьому етапі розроблюється контролер для обробки запитів користувачів. Він містить методи, які обробляють окремі запити. У методах контролера зазвичай проводиться валідація вхідних даних,

передача даних в модель та форматування відповіді користувачу. Контролер може також взаємодіяти з сервісами та іншими компонентами Symfony, які необхідні для виконання запитів.

Після розробки контролеру, додаток можна використовувати для обробки запитів користувачів та передачі даних в модель для розрахунку стейкінгу.

П'ятий етап розробки калькулятора полягає у розробці шаблонів вигляду (views), які будуть відповідати за відображення результатів розрахунків та форм для введення даних користувачем.

Для цього потрібно створити необхідні шаблони з використанням технології шаблонізації Symfony, яка дозволяє створювати HTML-шаблони за допомогою змінних та блоків, що дає змогу зробити відображення динамічним та змінювати зміст сторінок в залежності від введених даних користувачем.

Основний шаблон для калькулятора буде містити форму для введення даних та результат розрахунку. Також можуть бути додані додаткові шаблони для відображення допоміжної інформації, такої як інструкції користування калькулятором, пояснення до розрахунків тощо.

У цьому етапі також потрібно врахувати дизайн та стилізацію шаблонів, щоб забезпечити їх зручне відображення на різних пристроях та браузерах. Для цього можна використовувати CSS-фреймворки або стилізувати шаблони вручну.

Важливо пам'ятати, що шаблони вигляду повинні бути інтуїтивно зрозумілими та зручними для користувачів, щоб вони могли легко здійснювати необхідні розрахунки та отримувати потрібну інформацію.

Шостий етап передбачає налаштування маршрутів в додатку, щоб коректно обробляти запити користувачів і повертати їм відповідні

результати. У Symfony маршрути визначаються в файлі “routes.yaml”, де для кожного маршруту вказується шлях до нього, метод, який він приймає, та контролер, який обробляє запит.

Наприклад, маршрут для відображення головної сторінки калькулятора може мати шлях “/”, метод “GET” і контролер “App\Controller\MainController::index”. Цей контролер може обробляти запит і повертати відповідний шаблон вигляду (view) для відображення головної сторінки калькулятора.

Також, маршрути можна налаштовувати для обробки запитів з параметрами, наприклад, якщо користувач вводить дані для розрахунку в формі на іншій сторінці. У цьому випадку, маршрут може містити певні параметри, які передаються в контролер для обробки запиту.

Після налаштування маршрутів, Symfony автоматично забезпечує коректну маршрутизацію запитів користувачів і передачу їх відповідним контролерам для обробки.

Сьомий етап "Тестування додатку та відлагодження помилок" включає в себе проведення різноманітних тестів додатку для перевірки правильності його роботи та виявлення помилок, які потребують відлагодження.

У процесі тестування додатку можна використовувати різні види тестів, такі як юніт-тести, інтеграційні тести та енд-ту-енд тести. Юніт-тести перевіряють роботу окремих компонентів додатку, інтеграційні тести перевіряють взаємодію між компонентами, а енд-ту-енд тести перевіряють роботу всього додатку як єдиного цілого.

Після проведення тестів можна перейти до відлагодження помилок, які були виявлені. Відлагодження може включати в себе пошук причини помилки, виправлення помилки та перевірку роботи додатку після виправлення помилки.

Після завершення цього етапу, додаток має бути підготовлений до наступних етапів розробки, які включають в себе додавання нових функцій та покращення функціональності додатку.

Останній етап "Оптимізація та покращення продуктивності додатку" передбачає ряд дій, спрямованих на покращення продуктивності, швидкодії та ресурсоемності додатку.

На цьому етапі проводились наступні дії:

- 1) Оптимізація коду - перевірка коду на наявність неефективних частин, зайвих операцій, повільних алгоритмів та їх заміна на більш оптимальні.
- 2) Кешування даних - збереження результатів обчислень у кеші для їх подальшого використання, замість повторного розрахунку.
- 3) Перевірка на наявність "гарячих точок" - ідентифікація часто використовуваних ділянок коду та їх оптимізація.
- 4) Тестування продуктивності - проведення тестів, щоб перевірити, наскільки швидко працює додаток за різних умов.
- 5) Використання кращих інструментів - використання оптимізованих бібліотек, фреймворків та інструментів розробки, що дозволяють покращити продуктивність додатку.

Оптимізація та покращення продуктивності додатку є важливим етапом розробки, оскільки вона дозволяє підвищити швидкодію та ефективність додатку, що в свою чергу позитивно вплине на користувачів.

ВИСНОВКИ

Розробка калькулятора ставки для блокчейн систем є важливим і актуальним завданням, яке може бути вирішене з використанням сучасних технологій та інструментів. Розроблений калькулятор має можливість розрахунку ставок для блокчейн-майнінгу, стейкінгу та інших подібних процесів, що дозволяє користувачам легко визначити потенційні прибутки та ризики від цих операцій. Однак, в дослідженні були виявлені деякі обмеження та перспективи для подальшого розвитку калькулятора, що може бути розглянуто у майбутніх дослідженнях.

Мета досягнута, поставлені завдання виконані:

- 1) Проведений функціональних та нефункціональних вимог до калькулятора ставки для блокчейн-систем.
- 2) Виявлені потреби та можливості для створення калькулятора ставок.
- 3) Розроблено прототип сайту.

Проект вирішено реалізувати з використанням фреймворку Symfony, мов програмування PHP, JavaScript. Всі цілі і завдання були виконані.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Що таке блокчейн? URL: <https://finances.in.ua/shcho-take-blokchejn/>
2. Як працює технологія блокчейн? URL: <https://guland.com.ua/kryptovalyuta/blockchain/shcho-take-blokcheyn.htm#STB3>
3. Криптографічний захист інформації: поняття та практичне застосування. URL: <https://ips.ligazakon.net/document/situation-doc/SX170078>
4. Смарт контракти – чи дійсно договори та чи дійсно розумні? <https://unba.org.ua/publications/print/3169-smart-kontrakti--chi-spravdi-dogovori-ta-chi-dijsno-rozumni.html#:~:text=%D0%A1%D0%BC%D0%B0%D1%80%D1%82%20%D0%BA%D0%BE%D0%BD%D1%82%D1%80%D0%B0%D0%BA%D1%82%D0%B8%20%E2%80%93%D1%86%D0%B5%20%D1%83%D0%B3%D0%BE%D0%B4%D0%B8%2C%20%D0%BD%D0%B0%D0%BF%D0%B8%D1%81%D0%B0%D0%BD%D1%96,%D1%83%D0%BC%D0%BE%D0%B2%D0%B8%2C%20%D1%89%D0%BE%20%D0%B2%D0%B8%D0%BA%D0%BE%D0%BD%D1%83%D1%8E%D1%82%D1%8C%D1%81%D1%8F%20%D1%81%D1%82%D0%BE%D1%80%D0%BE%D0%BD%D0%B0%D0%BC%D0%B8%20%D1%83%D0%B3%D0%BE%D0%B4%D0%B8>.
5. Класифікація блокчейнів. URL: <https://www.bitbon.space/ua/knowledge-base/distributed-ledger-technologies-blockchain/technological-aspects-of-blockchain/classification-of-blockchains>
6. У чому різниця між публічними, приватними та консорціумними блокчейнами? URL: <https://academy.binance.com/uk/articles/private-public-and-consortium-blockchains-whats-the-difference>

7. Що таке стейкінг? URL:

<https://academy.binance.com/uk/articles/what-is-staking>

8. Стейкінг криптовалют – що це таке, як працює, види крипто. URL:

<https://probitcoin.com.ua/crypto-staking.html>

9. Разбираемся в принципах работы стейкинга. URL:

https://incrypted.com/steyking-na-binance/?utm_source=rss&utm_medium=rss&utm_campaign=steyking-na-binance

10.Що таке DeFi-стейкінг Binance і як користуватись. URL:

<https://www.binance.com/uk-UA/support/faq/%D1%89%D0%BE-%D1%82%D0%B0%D0%BA%D0%B5-defi-%D1%81%D1%82%D0%B5%D0%B9%D0%BA%D1%96%D0%BD%D0%B3-binance-%D1%96-%D1%8F%D0%BA-%D0%BD%D0%B8%D0%BC-%D0%BA%D0%BE%D1%80%D0%B8%D1%81%D1%82%D1%83%D0%B2%D0%B0%D1%82%D0%B8%D1%81%D1%8F-9823d6ddae0149f6aaefc5a762c8c69e>

11.Що таке реальна прибутковість у DeFi? URL:

<https://academy.binance.com/uk/articles/what-is-real-yield-in-defi>

12.Стейкинг: что это, как работает и как заработать - Блог Binance.

URL:

<https://www.binance.com/ru/blog/all/%D1%81%D1%82%D0%B5%D0%B9%D0%BA%D0%B8%D0%BD%D0%B3-%D1%87%D1%82%D0%BE-%D1%8D%D1%82%D0%BE-%D0%BA%D0%B0%D0%BA-%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%B0%D0%B5%D1%82-%D0%B8-%D0%BA%D0%B0%D0%BA-%D0%B7%D0%B0%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%B0%D1%82%D1%8C-->

[%D0%B1%D0%BB%D0%BE%D0%B3-binance-421499824684903007](https://binance.com/421499824684903007)

13. Staking rewards. <https://www.stakingrewards.com/>
14. Посібник PHP. URL: <https://www.php.net/manual/ru/index.php>
15. Symfony – Вікіпедія. URL: <https://uk.wikipedia.org/wiki/Symfony>
16. Довідковий посібник MySQL. URL: <http://www.mysql.ru/docs/man/>
17. Сучасний підручник JavaScript. URL: <https://learn.javascript.ru>
18. «Трирівневий MVC» у веб-програмуванні. URL:
<http://naukam.triada.in.ua/index.php/konferentsiji/45-p-yatnadtsyata-vseukrajinska-praktichno-piznavalna-internet-konferentsiya/304-tririvnevij-mvc-u-veb-programuvanni>
19. Етапи створення веб-сайтів. URL:
<https://sites.google.com/site/siteingschool/etapi-stvorennja-veb-sajtiv>
20. Як розгорнути додаток на Symfony. URL:
<https://symfony.com.ua/doc/current/deployment.html>

ДОДАТКИ

Додаток А

КОДЕКС АКАДЕМІЧНОЇ ДОБРОЧЕСНОСТІ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ ХЕРСОНСЬКОГО ДЕРЖАВНОГО УНІВЕРСИТЕТУ

Я, Чобулда Данило Олександрович, учасник освітнього процесу Херсонського державного університету, **УСВІДОМЛЮЮ**, що академічна доброчесність – це фундаментальна етична цінність усієї академічної спільноти світу.

ЗАЯВЛЯЮ, що у своїй освітній і науковій діяльності **ЗОБОВ'ЯЗУЮСЯ**:

– дотримуватися:

- вимог законодавства України та внутрішніх нормативних документів університету, зокрема Статуту Університету;
- принципів та правил академічної доброчесності;
- нульової толерантності до академічного плагіату;
- моральних норм та правил етичної поведінки;
- толерантного ставлення до інших;
- дотримуватися високого рівня культури спілкування;

– надавати згоду на:

- безпосередню перевірку курсових, кваліфікаційних робіт тощо на ознаки наявності академічного плагіату за допомогою спеціалізованих програмних продуктів;
- оброблення, збереження й розміщення кваліфікаційних робіт у відкритому доступі в інституційному репозитарії;
- використання робіт для перевірки на ознаки наявності академічного плагіату в інших роботах виключно з метою виявлення можливих ознак академічного плагіату;

- самостійно виконувати навчальні завдання, завдання поточного й підсумкового контролю результатів навчання;
- надавати достовірну інформацію щодо результатів власної навчальної (наукової, творчої) діяльності, використаних методик досліджень та джерел інформації;
- не використовувати результати досліджень інших авторів без використання покликань на їхню роботу;
- своєю діяльністю сприяти збереженню та примноженню традицій університету, формуванню його позитивного іміджу;
- не чинити правопорушень і не сприяти їхньому скоєнню іншими особами;
- підтримувати атмосферу довіри, взаємної відповідальності та співпраці в освітньому середовищі;
- поважати честь, гідність та особисту недоторканність особи, незважаючи на її стать, вік, матеріальний стан, соціальне становище, расову належність, релігійні й політичні переконання;
- не дискримінувати людей на підставі академічного статусу, а також за національною, расовою, статевою чи іншою належністю;
- відповідально ставитися до своїх обов'язків, вчасно та сумлінно виконувати необхідні навчальні та науково-дослідницькі завдання;
- запобігати виникненню у своїй діяльності конфлікту інтересів, зокрема не використовувати службових і родинних зв'язків з метою отримання нечесної переваги в навчальній, науковій і трудовій діяльності;
- не брати участі у будь-якій діяльності, пов'язаній із обманом, нечесністю, списуванням, фабрикацією;
- не підроблювати документи;
- не поширювати неправдиву та компрометуючу інформацію про інших здобувачів вищої освіти, викладачів і співробітників;

- не отримувати і не пропонувати винагород за несправедливе отримання будь-яких переваг або здійснення впливу на зміну отриманої академічної оцінки;
- не залякувати й не проявляти агресії та насильства проти інших, сексуальні домагання;
- не завдавати шкоди матеріальним цінностям, матеріально-технічній базі університету та особистій власності інших студентів та/або працівників;
- не використовувати без дозволу ректорату (деканату) символіки університету в заходах, не пов'язаних з діяльністю університету;
- не здійснювати і не заохочувати будь-яких спроб, спрямованих на те, щоб за допомогою нечесних і негідних методів досягати власних корисних цілей;
- не завдавати загрози власному здоров'ю або безпеці іншим студентам та/або працівникам.

УСВІДОМЛЮЮ, що відповідно до чинного законодавства у разі недотримання Кодексу академічної доброчесності буду нести академічну та/або інші види відповідальності й проти мене можуть бути застосовані заходи дисциплінарного характеру за порушення принципів академічної доброчесності.

12.09.2019 р.

(дата)



(підпис)

Чобулда Данило

(ім'я, прізвище)