

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Наукові праці
Донецького національного технічного
університету

**Серія: “Інформатика, кібернетика
та обчислювальна техніка”**

Всеукраїнський науковий збірник

Заснований у травні 1996 року

Виходить 2 рази на рік

№ 2(35) ' 2022 - №1(36) ' 2023

Луцьк – 2023

Публікується згідно з рішенням Вченої ради ДВНЗ «Донецький національний технічний університет» (протокол № 2 від 23.02.2023).

Збірник містить наукові статті співробітників ДонНТУ та інших навчальних і наукових закладів України, які є науковими партнерами ДонНТУ. Публікації висвітлюють результати наукових досліджень і розробок в таких напрямках, як інформатика, чисельні методи, паралельні обчислення, програмування, розробка засобів обчислювальної техніки, дослідження комп'ютерних мереж, машинна графіка і обробка зображень, математичне моделювання в різних галузях. Матеріали збірника призначені для наукових співробітників, викладачів, інженерно-технічних працівників, аспірантів та студентів.

Засновник та видавець – Донецький національний технічний університет (ДонНТУ)

РЕДАКЦІЙНА КОЛЕГІЯ:

Д-р техн. наук, проф. Є.О. Башков (головний редактор); канд. техн. наук, доц. Н.О. Маслова (заступник головного редактора); член-кореспондент НАН України, д-р техн. наук, проф. В.П. Боюн; д-р техн. наук, проф. О.О. Баркалов; д-р техн. наук, проф. О.А. Дмитрієва, д-р техн. наук, проф. О.В. Вовна; д-р техн. наук, проф. А.А. Зорі; д-р техн. наук, проф. С.Д. Погорілий; д-р техн. наук, проф. О.Н. Романюк; д-р техн. наук, проф. В.А. Святний; д-р техн. наук, проф. Тянев Димитр Стоянов; д-р техн. наук, проф. Г.Г. Швачич; канд. техн. наук, доц. І.Я. Зеленцова; канд. техн. наук, доц. С.М. Вороной; канд. техн. наук, доц. І.А. Назарова (відп. секретар випуску).

Адреса редакції:

Юридична адреса: 85300, Україна, Донецька область, м. Покровськ, пл. Шибанкова, 2, ДВНЗ «ДонНТУ», 3.312.

Фактична адреса: 43018, Україна, Волинська область, м. Луцьк, вул. Потебні, 56, ДВНЗ «ДонНТУ».

E-mail: yevhen.bashkov@donmtu.edu.ua

Збірник зареєстровано в Державному комітеті інформаційної політики, телебачення та радіомовлення України. Свідоцтво: серія КВ, №7374 від 03.06.2003.

Збірник включено до переліку наукових фахових видань України, в яких можуть публікуватися результати дисертаційних робіт на здобуття наукових ступенів доктора наук, кандидата наук та ступеня доктора філософії за спеціальностями 121 Інженерія програмного забезпечення, 122 Комп'ютерні науки, 123 Комп'ютерна інженерія (наказ Міністерства освіти і науки України №409 від 17 березня 2020 р.)

Збірник "Наукові праці ДонНТУ, серія "Інформатика, кібернетика та обчислювальна техніка" за наказом № 409 МОНУ від 17. 03.2020 отримав категорію Б.

ЗМІСТ

О.А. Дмитрієва, Є.О. Бабенко	4
Проектування моделі розпізнавання текстових даних з застосуванням обчислювального інтелекту	
Н.О. Маслова, А.А. Гусєва	11
Застосування мережі Stellar при управлінні криптоактивами	
Є.В. Павловський, І.В. Ярош	17
Аналіз методів представлення деревовидних структур у реляційній моделі даних	
В.І. Голінько, О.В. Голінько	22
Методи та алгоритми контролю стану газодифузійного фільтра аналізаторів метану	
Т.В. Алтухова	30
Математичне обґрунтування впливу діагностичних ознак на технічний стан міжфазної ізоляції електродвигунів	
Ю.С. Антонов, О.В. Мазурук	36
Особливості розробки підсистеми обрання наукового керівника	
Є.К. Завальнюк, О.Н. Романюк, А.В. Снігур, Р.П. Шевчук	45
Розробка фізично коректної моделі відбиття світла на основі розрахунку кута між векторами	
Г.А. Єгошина, С.М. Вороной, О.І. Полікаровських, Р.О. Гохман	51
Аналіз відповідності інтерфейсів Progressive Web Apps та веб-застосувань засобами трансформеру DistilBERT	
Є.О. Башков, Т.В. Алтухова, Є.О. Єжова	61
Розробка методу аутентифікації користувача на основі клавіатурного почерку	
І.А. Назарова, С.В. Голуб	69
Топологічні аспекти моделювання паралельних обчислень при розв'язанні жорстких задач Коші у кластерних системах	
І.В. Ярош, Є.В. Павловський, І.А. Назарова	79
Математичне моделювання процесу прогнозування витрат часу на розв'язання типової задачі з розробки програмного забезпечення	
О. В. Самощенко, Б. С. Гусєв, <u>В. В. Лапко</u>	85
Математичний опис операції ділення цілих чисел	
Ю.В. Прокоп, О.Г. Трофименко, О.В. Шевченко, Д.І. Новіков	97
Візуалізація як засіб покращення ефективності засвоєння курсу алгоритмів та структур даних	

А.А. Гусева, магістр,
Н.О. Маслова, к.т.н., доцент
Донецький національний технічний університет, м. Луцьк, Україна
natalia.maslova@donntu.edu.ua
anna.husieva.kita@donntu.edu.ua

Застосування мережі Stellar при управлінні криптоактивами

Під час активного розвитку технології Blockchain, та росту популярності криптовалют, у користувачів з'явилась потреба в ефективному управлінні своїми активами. В роботі представлено процес створення системи контролю криптоактивів із мінімальним набором функцій, зрозумілим інтерфейсом та досягненням основної мети користувачів – систематизацією та сприянням ефективному росту вкладених інвестицій. Однією з популярних платформ для проведення операцій з блокчейном є мережа Stellar (Stellar Network). Завдяки гнучкості мережі та її особливостям, створено кросплатформну систему, зв'язок із якою виконується засобами Telegram, а основний функціонал розроблено з застосуванням Horizon API та SDK на мові програмування Python

Ключові слова: Stellar Network, блокчейн, програма, Python, система, мережа, транзакція, криптоактиви

DOI: 10.31474/1996-1588-2023-1-36-11-16

Вступ

Сьогодні «криптовалюта» користується великим попитом серед людей, що обумовлено багатьма факторами. Основними з цих факторів є: незалежність від коливань цифрового валютного ринку, децентралізація та надійність грошових транзакцій. Через це люди інвестують свої фінанси у криптоактиви та шукають способи їх примноження. Для роботи з перспективними криптоактивами вони користуються блокчейн-мережами. Однією з таких блокчейн-мереж є Stellar, що має певні відмінності від аналогів.

Метою роботи є: розкриття переваг Stellar Network при роботі з криптоактивами; опис системи, що призначена задовольнити основні потреби користувачів; наочна демонстрація переваг та ефективності роботи системи в порівнянні з неавтоматизованою роботою користувача.

Опис особливостей Stellar Network

Stellar Network – це мережа, що була розроблена для інноваційних внесків та конкуренції в міжнародних транзакціях. Основним нововведенням мережі є безпечний механізм транзакцій через ненадійних посередників із використанням SCP (Secure Copy) — протоколу для безпечного копіювання файлів між локальним та віддаленим хостом або між двома віддаленими хостами [1].

Платіжні мережі мають низьку швидкість транзакцій у порівнянні з іншими типами

розподільних систем. Ведучі блокчейни Bitcoin та Ethereum, підтверджують до 15 транзакцій/сек, що менше, ніж у Stellar. Крім того, цим системам потрібно забагато часу (від кількох хвилин до години), щоб підтвердити транзакцію безпечним способом, тому що «докази роботи» вимагають очікування, поки не буде добуто принаймні кілька блоків. Набір SWIFT без блокчейна в піковий день виконувався в середньому всього 420 транзакцій/сек [2]. В свою чергу Stellar Network було порівняно з 5-секундним цільовим інтервалом реєстру. Результати показали значно нижчі затримки даної границі, навіть якщо кілька нереалізованих оптимізацій все ще знаходяться в процесі обробки.

На даний момент застосовується 132 повних активних вузлів, з них 52 беруть участь у консенсусі, підписуючи повідомлення про голосування. Рис.1 демонструє мережу з лінією між двома вузлами, якщо один з'являється в сегментах кворуму іншого (бліда лінія). Темною лінією показано двонаправлену залежність [3]. У центрі знаходиться кластер із 26 «валідаторів першого рівня», якими керують відомі регулятори в галузі криптовалютної індустрії, такі, як Wirex, SDF, Blockdaemon, SatoshiPay, Public Node, LOBSTR, COINVEST і Franklin Templeton, та інші. Зрізи кворуму мають важливу роль при початковому навантаженні мережі.

SCP дозволяє Stellar атомарно здійснювати незворотні транзакції між довільними учасниками, які не знають один про одного та не довіряють один одному. Це, у свою чергу, гарантує новим користувачам доступ до тих же ринків, що й

усталеним користувачам, забезпечує отримання найкращих доступних обмінних курсів навіть від ненадійних «маркет-мейкерів» і значно скорочує затримку платежів.

Криптовалюта, яка працює в мережі Stellar Stellar має аббревіатуру XLM (xlm).

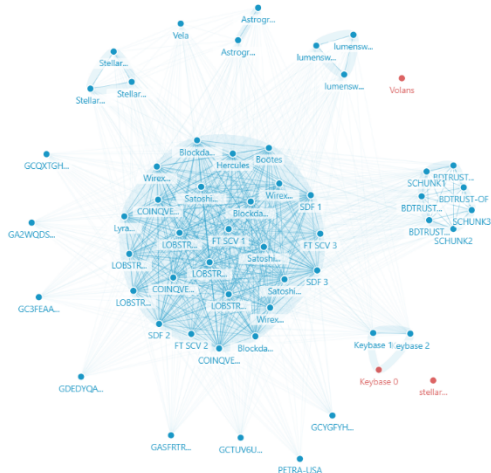


Рисунок 1 – Карта зрізів кворумів

Враховуючи особливості Stellar Network, виникають вагомі підстави для розробки системи, що автоматизує основні процеси при роботі користувача із мережею.

Завдання системи та засоби їх реалізації

Як було вже зазначено, основною метою користувачів є інвестування свої фінансових активів для подальшого збільшення свого прибутку за рахунок руху ринку криптовалют. Для цього користувачу необхідно постійно знаходитись у пошуку вигідних криптоактивів та не забувати про керування вже існуючих у його «криптопортфелі».

Для керування своїми угодами, користувач повинен уважно виконувати певний перелік однотипних дій, на які можуть впливати різні фактори. Аби зменшити їх вплив, до розроблюваної системи було поставлено такі основні завдання:

- автоматичне редагування угод користувача шляхом зміни їх параметрів на більш вигідні, відповідно до руху ринку відповідного криптоактиву та заданих параметрів користувача (об'єм закупівлі, ціна та інші);
- автоматичний пошук вигідних угод криптоактивів за заданими параметрами користувача (об'єм закупівлі, ціна та інші).

Окрім вищенаведених задач передбачити можливість:

- переказу коштів;
- отримання балансу гаманця;
- отримання інформації про стан гаманця (що було отримано, продано і т.п.).

Для реалізації поставлених задач було використано:

- засоби Stellar Network (Horizon API) для зв'язку із мережею [4];
- мову програмування Python – як оптимальний комплект для розробки програмного забезпечення для роботи із Horizon API [5];
- додаток Telegram – для комфортного та швидкого з'єднання користувача та системи [6].

Основні аспекти створення системи

На основі поставлених задач та можливостей обраних засобів розробки, було створено автоматизовану систему StellarJobber.

Для здійснення основної мети, систему було спроектовано таким чином, аби з нею міг працювати навіть користувач з мінімальними навичками застосування програмних засобів.

Проектування системи було орієнтовано на зрозумілу та чітку послідовність дій й набір основних функцій, що задовольняють поставленим задачам. На рисунку 2 представлено діаграму прецедентів, головним актором якої виступає користувач.

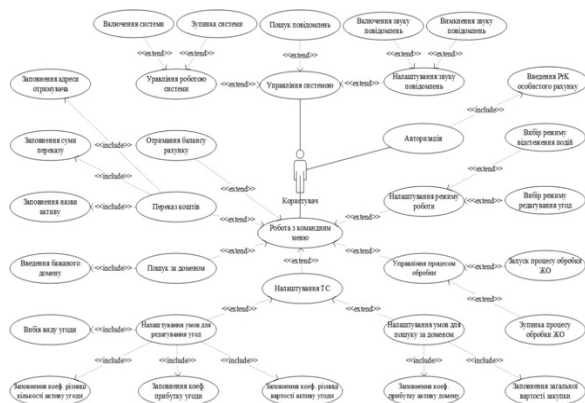


Рисунок 2 – Діаграма прецедентів

Для актора доступний набір із трьох основних функцій (авторизація, управління системою, робота з командним меню), які в свою чергу розширюються та включають у себе інший набір функцій. Найбільш функціональним є набір при роботі із командним меню. До нього входять такі функції: переказ коштів, налаштування торгівельної стратегії (ТС), отримання балансу рахунку, пошук за доменом, управління процесом роботи, налаштування режиму роботи.

Оскільки зв'язок користувача з системою передбачає реалізацію через Telegram, було спроектовано діаграму поведінки системи (станів) для відображення змін її стану при роботі з різними функціями. Дану діаграму відображено на рис. 3.

Система була реалізована за допомогою 10 основних класів:

- MainMenu (головне меню програми та зв'язок із іншими класами);
- GetBalance (отримання інформації про баланс рахунку користувача);
- ChangeTS (зміна параметрів торгівельної стратегії);
- Authorization (авторизація користувача);
- DomainSearch (пошук нових криптоактивів за наданим доменом);
- TSDomainSearch (редагування параметрів торгівельної стратегії пошуку нових криптоактивів);
- TSEditOrders (редагування параметрів торгівельної стратегії для керування угодами користувача);
- PaymentProcess (відправлення коштів);
- ModeProgram (перемикає режими роботи системи);
- Processing (відповідає за стан обробки даних програмою).

вигляді короткої інформації (про налаштування системи) та групою кнопок можливих функцій. Для вибору потрібної функції необхідно натиснути на відповідну кнопку з її назвою. На рис. 5 показано результат вибору функції «Особистий гаманець». Повернутися до головного меню можна за допомогою команди /menu або вбудованої до усіх функцій кнопки «До головного меню» (рис. 5).

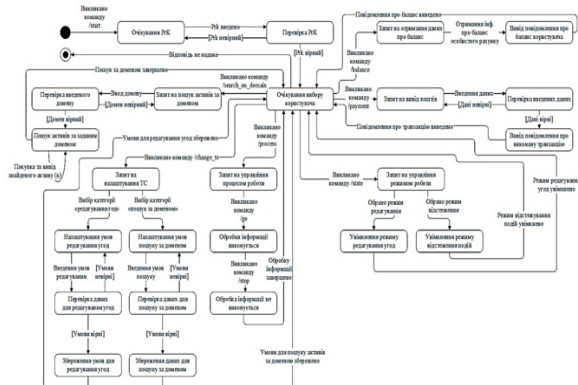


Рисунок 3 – Діаграма станів

Інтерфейс користувача реалізований таким чином, аби мінімізувати зайві дії, помилкові запити та відобразити основну інформацію та функції. На рис. 4 показано процес роботи користувача із програмою.

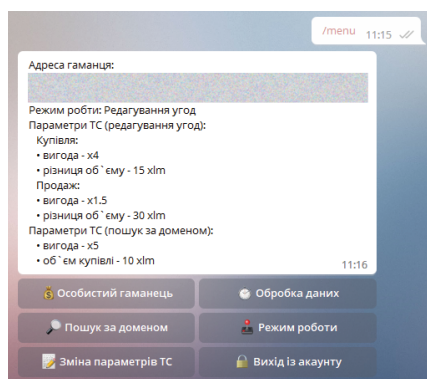


Рисунок 4 – Інтерфейс користувача
(команда /menu)

Користувач надсилає команду (у даному випадку команду – /menu) та отримує відповідь у

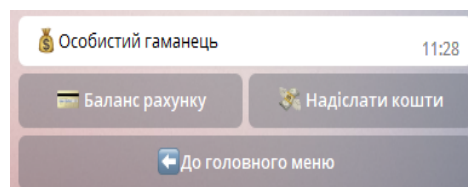


Рисунок 5 – Інтерфейс користувача
(особистий гаманець)

Під час роботи системи користувач буде отримувати постійні повідомлення про результати обробки. На рис. 6 показано приклад повідомлення, яке користувач отримає при зміні параметрів угоди на продаж згідно користувацьких умов.

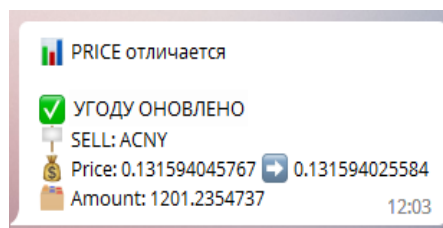


Рисунок 6 – Повідомлення про зміну параметрів угоди

В результаті система StellarJobber виконує усі поставлені для неї задачі, спрощує управління угодами та криптоактивами користувача, звітує про виконання своїх дій та надає можливість редагування параметрів, що мінімізує можливість виникнення непередбачуваних подій (таких як зменшення вигоди, втрата коштів та інші).

Аналіз ефективності роботи системи

По завершенню розробки основних функцій програми та підключенню її до Telegram, було необхідно провести експерименти, аби дослідити ефективність роботи користувача з програмою.

До експерименту було долучено групу користувачів. Для достовірності було забезпечено такі загальні умови для кожного користувача:

- баланс рахунку кожного користувача налічував 300 xlm;
- на початку роботи слід виконати закупку криптоактивів на ≈ 200 xlm;
- на закупку одного активу витратити від 5 до 15 xlm;
- виконувати свій порядок дій 31 день;
- фіксувати баланс рахунку кожен день.

Користувачам системи були поставлені умови:

– виконати закупку монет за допомогою функції програми;

– після закупки монет, на протязі місяця, кожний ранок/день включати програму та виключати вечері.

Для користувачів без системи були поставлені умови:

– виконувати закупку монет власноруч;

– на протязі місяця, відстежувати та редагувати угоди не менше 3 разів на день.

На рис. 7 представлено результати місячного експерименту щодо ефективності прибутку користувачів у вигляді графіку.



Рисунок 7 – Результат експерименту ефективності прибутку

Перша група користувачів (без системи) збільшила свій прибуток в 1.7 рази. Друга група – в 3 рази. На графіку (рис. 7) показано, що прибуток першої групи користувачів менший, ніж у другої групи користувачів в 2.5 рази.

Як результат – використання системи сприяє більш ефективному зростанню прибутку користувачів.

Окрім цього, було виконано дослідження щодо ефективності витрачання часу на виконання функцій системи: редагування угод та пошуку криптоактивів. Для цього було розраховано середній час, який затрачує користувач на виконання однієї задачі:

– редагування угод передбачає процес перегляду угоди та, при необхідності, зміну параметрів;

– пошук криптоактивів передбачає процес перегляду та налаштування параметрів для створення угоди.

У табл. 1 представлено отримані дані щодо витраченого на виконання функції редагування угод часу:

- 1) T_1 – час редагування вручну;
- 2) T_2 – час редагування системи.

На рис. 8 наведено порівняльний графік, за яким видно, що ручна обробка угод потребує значно більшого часу, ніж обробка системою.

Таблиця 1 – Порівняння часу, витраченого на редагування угод

Кількість угод	T_1 (с)	T_2 (с)
1	0,35	0,12
10	3,50	1,21
20	7,00	2,43
40	14,00	4,85
60	21,00	7,28
100	35,00	12,13

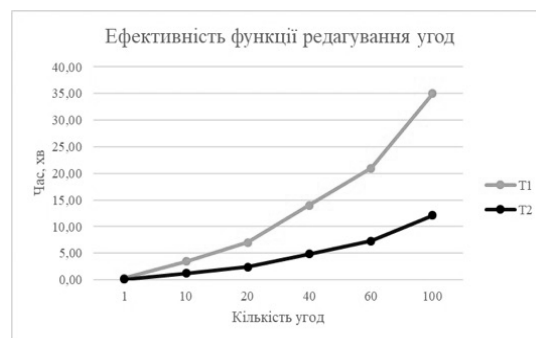


Рисунок 8 – Результат експерименту з функцією редагування угод

У табл. 2 представлено отримані дані щодо затраченого часу на виконання функції пошуку вигідних криптоактивів.

Таблиця 2 – Порівняльні дані затраченого часу на пошук криптоактивів

Кількість криптоактивів	T_1 (с)	T_2 (с)
1	0,52	0,19
10	5,17	1,86
20	10,33	3,71
40	20,67	7,43
60	31,00	11,14
100	51,67	18,57

На рис. 9 показано графік, за яким видно, що ручна обробка угод потребує значно більшого часу, ніж обробка системою.

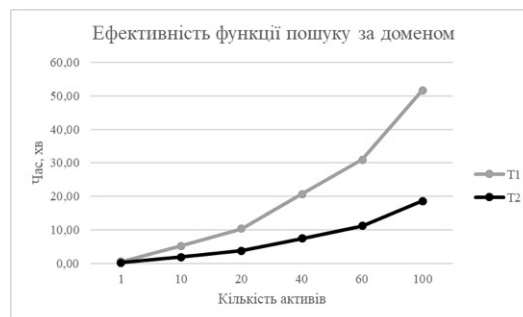


Рисунок 9 – Результат експерименту над функцією пошуку вигідних криптоактивів

Висновки

В статті розглянута мережа Stellar Network, проаналізовано її особливості та переваги.

Сформульована мета роботи та основні вимоги до програмної системи. Обґрунтовано обрання засобів для її реалізації.

Описано процес створення та роботи системи.

Проведено ряд експериментів (контроль прибутку, час редагування угод, пошук

криптоактивів) й отримано дані, котрі демонструють переваги застосування розробленої системи над не автоматизованим способом керування активами.

Експерименти показали можливість майже двократного зростання прибутку, що може бути досягнуто при застосуванні системи. Показано наявність переваг при проведенні експерименту редагування угод, а також суттєве зменшення часу пошуку вигідних криптоактивів.

Список літератури

1. Fast and secure global payments with Stellar. URL: <https://www.stellar.org/papers/fast-and-secure-global-payments-with-stellar>
2. SWIFT in Figures July 2019. URL: <https://www.swift.com/aboutus/swift-fin-traffic-figures/monthly-figures>
3. Stellar network visibility. URL: <https://stellarbeat.io>
4. Horizon API Reference. URL: <https://developers.stellar.org/api/introduction/>
5. Stellar Python SDK. URL: <https://stellar-sdk.readthedocs.io/en/stable/index.html#stellar-python-sdk>
6. Гусєва А.А. Маслова Н.О. Засоби для створення автоматизованого управління криптовалютним портфелем на основі технології Blockchain з застосуванням Stellar Network. Матеріали III Міжнародної студентської наукової конференції «Теоретичне та практичне застосування результатів сучасної науки» (м. Біла Церква, 07.10.2022) С. 177-178.

References

1. "Fast and secure global payments with Stellar", available at: <https://www.stellar.org/papers/fast-and-secure-global-payments-with-stellar>
2. "SWIFT in Figures July (2019)", available at: <https://www.swift.com/aboutus/swift-fin-traffic-figures/monthly-figures>
3. "Stellar network visibility", available at: <https://stellarbeat.io>
4. "Horizon API Reference", available at: <https://developers.stellar.org/api/introduction/>
5. "Stellar Python SDK", available at: <https://stellar-sdk.readthedocs.io/en/stable/index.html#stellar-python-sdk>
6. Husyeva, A.A., Maslova, N.O. (2022), Tools for creating automated cryptocurrency portfolio management based on Blockchain technology using the Stellar Network [Zasoby dlya stvorenniya avtomatyzovanoho upravlinnya kryptovalyutnym portfelem na osnovi tekhnolohiyi Blockchain z zastosuvannyam Stellar Network], *Materials of the III International Student Scientific Conference "Theoretical and Practical Application of the Results of Modern Science" (Bila Tserkva, October 7, 2022)*, pp. 177-178.

Надійшла до редакції 16.11.2022

A. HUSIEVA, N. MASLOVA

Donetsk National Technical University, Luts'k, Ukraine

anna.husieva.kita@donntu.edu.ua

nataliia.maslova@donntu.edu.ua

APPLICATION OF THE STELLAR NETWORK IN THE MANAGEMENT OF CRYPTO ASSETS

During the active development of Blockchain technology and the growing popularity of cryptocurrencies, users have a need for effective management of their assets. The work presents the process of creating a cryptoasset control system with a minimal set of functions, a clear interface and achieving the main goal of users - systematization and promotion of the effective growth of invested investments. One of the popular platforms for conducting blockchain operations is the Stellar network (Stellar Network). Thanks to the flexibility of the network and its features, a cross-platform system has been created, the communication with which is carried out by means of Telegram, and the main functionality has been developed using the Horizon API and SDK in the Python programming language

Keywords: *Stellar Network, Blockchain, Program, Python, System, Network, Transaction, Crypto Assets*

Є.В. Павловський, ст. викладач,
І.В. Ярош, старший викладач
Донецький національний технічний університет, м. Луцьк, Україна
yevhen.pavlovskyi@donntu.edu.ua,
iryna.yarosh@donntu.edu.ua

Аналіз методів представлення деревовидних структур у реляційній моделі даних

В статті розглянуто та проаналізовано найпопулярніші методи представлення деревовидних структур у реляційній моделі даних. Типовий приклад таких даних – категорії, яких може бути досить багато з великим рівнем вкладеності одна в одну. До розглянутих методів належить список суміжностей (Adjacency List) та вкладені множини (Nested Sets). Були отримані кількісні показники часу вибірки даних, які представлені в базі даних розглянутими методами. На основі цих показників проведено аналіз доцільності представлення даних розглянутими методами деревовидних структур, у залежності від характеристик вкладеності та обсягів даних. Виявлено переваги та недоліки розглянутих методів представлення деревовидних структур у реляційній моделі даних.

Ключові слова: список суміжностей (Adjacency List), вкладені множини (Nested Sets), деревовидні структури, рекурсивна вибірка, реляційна модель даних

DOI: 10.31474/1996-1588-2023-1-36-17-21

Вступ

Як відомо, дані повинні зберігатися в такому вигляді, щоб їх вибірка відбувалась з мінімальними затратами часу та задіяних ресурсів. У більшості випадків для зберігання та обробки даних зручно та доцільно використовувати реляційну модель, яка знайшла своє застосування в реляційних базах даних. Перевагами такого представлення даних є той факт, що сутності представляються у вигляді таблиць, пов'язаних між собою різноманітними зв'язками: «один-до-одного», «один-до-багатьох» та «багато-до-багатьох» [1].

Таким чином, маємо єдину систему, пов'язаних між собою сутностей, які представляють деяку предметну область. Кожна сутність має набір полів, які мають певні типи даних: числовий, текстовий, дата/час та інші. Крім того у таблиць є ключове поле, яке однозначно ідентифікує запис в таблиці. Саме це поле відіграє важливу роль при побудові зв'язків між таблицями. Завдяки таким зв'язкам забезпечується приналежність записів однієї таблиці до іншої. При правильному проектуванні такої системи, вона повинна відповідати третій нормальній формі [2].

На сьогоднішній день досить актуальною задачею є представлення деревовидних структур у реляційній моделі даних. Типовий приклад таких даних – категорії, яких може бути досить багато з великим рівнем вкладеності одна в одну.

В даній статті розглянемо та проаналізуємо найпопулярніші методи представлення деревовидних даних, таких як списки суміжності (Adjacency List) та вкладені множини (Nested Sets),

виявимо переваги та недоліки даних методів. Проведемо аналіз отриманих кількісних показників часу вибірки даних, які представлені розглянутими методами.

Огляд предметної області

Щоб більш наочно продемонструвати представлення таких даних, наведемо фрагмент реляційної бази даних «Блог», а саме сутність «Користувач», яка відповідає таблиці «users» та сутність «Повідомлення», яка відповідає таблиці «posts». Користувач може створювати безліч повідомлень, а повідомлення, в свою чергу, може належати тільки одному користувачу. Таким чином маємо зв'язок між цими таблицями «один-до-багатьох» (рис. 1).

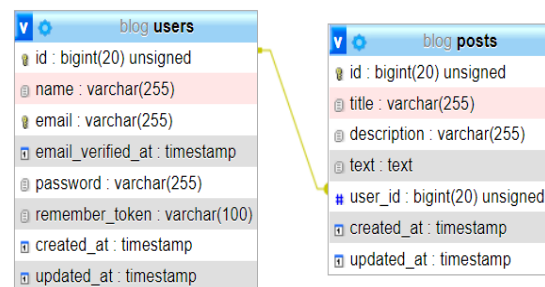


Рисунок 1 – Таблиці поєднані зв'язком
«один-до-багатьох»

Щоб зробити вибірку повідомлень, які належать користувачу з id=1, потрібно зробити sql-запит, який наведений на рисунку 2.