

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Спеціальність: 123 «Комп'ютерна інженерія»

Освітня програма: «Комп'ютерна графіка і Web-дизайн»

Група: 4КГ-05

Дипломний проект

здобувача освіти денної форми навчання

КГ.05.03.000.ДП

*Гаджисєва Аміра
Матін-огли*

м. Одеса
2022 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

Спеціальність: **123 «Комп'ютерна інженерія»**

Освітня програма: **«Комп'ютерна графіка і Web-дизайн»**

Група: **4КГ-05**

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проекту (роботи) на тему:

Розробка чат-боту месенджера Telegram для виконання операцій з цифровою валютою (реалізація візуального дизайну).

Проектний матеріал складається з пояснювальної записки на 6 сторінках та графічного (презентаційного) матеріалу на 2 аркушах (слайдах).

Дипломник _____ (Гаджиев А.М.)

Керівник _____ (Кривченко Ю.В.)

Консультанти:

з економічної частини _____ (Копайгородська Т.Г.)

з охорони праці _____ (Чорновол Н.І.)

з дотримання вимог ЄСКД _____ (Петрашова В.І.)

старший консультант _____ (Скорнякова О.В.)

До захисту допущений

Голова циклової комісії _____ (Скорнякова О.В.)

Завідувач відділення _____ (Суліма Ю.Ю.)

Захист « » _____ 2022 р. Протокол ДКК № _____

Оцінка ДКК _____

Секретар ДКК _____

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНАХТ»

Відділення комп'ютерних систем Комісія КТ та Ш
Спеціальність 123 «Комп'ютерна інженерія»
Освітня програма «Комп'ютерна графіка і Web-дизайн»

ЗАТВЕРДЖУЮ:

Заст. дир. з НВР _____

Беркань І.В.

“ _____ ” _____ 2022 р.

ЗАВДАННЯ

на дипломний проект (роботу)

Здобувачеві (здобувачці) освіти Гаджисев Амір Матін огли
(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Розробка чат-боту месенджера Telegram для виконання операцій з цифровою валютою (реалізація інтерфейсу)

затверджена наказом по коледжу від “ _____ ” _____ 2022 р. № _____

2. Термін здачі закінченого проекту (роботи) _____

3. Вихідні данні до проекту (роботи) _____

1. Використання номеру гаманця криптовалюти Ефір;

2. Використання технології блокчейн;

3. Мова програмування Python, інтегроване середовище розробки PyCharm;

4. Реалізувати програмний модуль розробленої системи: відстеження транзакцій, балансу гаманця мережі криптовалюти Ethereum, курсу криптовалют біржі, зміна мови інтерфейсу.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які необхідно розробити)
Опис предметної області. Загальні відомості про криптовалюту. Переваги криптовалют.
Огляд мережі криптовалюти Ethereum. Види взаємодії із мережею криптовалюти Ethereum.
Розробка алгоритму обробки даних. Розробка інтерфейсу програми. Проектування структур даних програми. Проектування структури модулів програми. Тестування розробленої системи. Економічна частина. Охорона праці

5. Перелік графічного (презентаційного) матеріалу (з точним зазначенням обов'язкових креслень, кількості слайдів)
Вартість криптовалюти у часі; Схема взаємодії користувача та розробленого проекту;
Порівняння аналогів; Алгоритм прийому нових блоків; Алгоритм обробки блока; Алгоритм реєстрації адреси; Загальна структура проекту; Структура користувацького інтерфейсу;
Склад користувацький інтерфейсу; Інструкція для користувача інтерфейсу; Меню боту;
Зміна мови інтерфейсу; Найцінніші гаманці Ефіру; Баланс гаманця; Транзакція у боті;
Перевірка роботи курсу криптовалют.

6. Консультанти по проекту (роботі), із зазначенням розділів проекту, що їх стосується

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
1. Технологічний розділ	Кривченко Ю.В.		
2. Екон. частина	Копайгородська Т.Г.		
3. Охорона праці	Чорновол Н.І.		
Нормоконтроль	Петрашова В.І.		

7. Дата видачі завдання _____

Керівник *Кривченко Ю.В.* _____
(підпис)

Завдання прийняв до виконання _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/р	Назва етапів дипломного проекту (роботи)	Термін виконання етапів дипломного проекту (роботи)	Відмітка про виконання
1.	Вступ. Постановка мети та задач проектування	17.05.2022	
2.	Аналіз предметної галузі	18.05.2022	
3.	Огляд функціональних елементів для реалізації моделі	19.05.2022	
4.	Основні вимоги до сервісу для взаємодії з біржею	20.05.2022	
5.	Вибір технічних та програмних засобів розробки	23.05.2022	
6.	Вибір технологій взаємодії з мережею криптовалюти	24.05.2022	
7.	Опис способів реалізації інтерфейсу користувача	26.05.2022	
8.	Опис алгоритму прийому блоків даних	30.05.2022	
9.	Опис алгоритму обробки даних	31.05.2022	
10.	Опис алгоритму роботи інтерфейсу	1.06.2022	
11.	Розробка інтерфейсу програми	3.06.2022	
12.	Тестування розробленої системи	6.06.2022	
13.	Аналіз результатів, підготовка презентації	8.06.2022	
14.	Розробка питань з економіки та охорони праці	10.06.2022	
15.	Підготовка проекту до захисту	14.06.2022	

Дипломник _____
(підпис)

Керівник _____
(підпис)

ЗМІСТ

Вступ.....	7
1 Технологічний розділ.....	8
1.1 Формування функціоналу та аналіз технологій.....	8
1.2 Вибір технологій та інструментів для реалізації моделі.....	11
1.2.1 Технологія Ethereum.....	13
1.2.2 Криптовалюта Ефір.....	13
1.3 Визначення задач проектування.....	14
1.3.1 Мінімізація порогу входу.....	14
1.3.2 Створення простий, інтуїтивно зрозумілий для звичайної людини інтерфейс.....	15
1.3.3 Широкий спектр можливостей у плані функціоналу програми.....	16
1.3.4 Короткий огляд альтернативних мереж.....	16
1.4 Основні етапи проектування.....	18
1.4.1 Вибір технологій взаємодії з мережею криптовалюти.....	18
1.4.2 Огляд способів реалізації інтерфейсу користувача.....	19
1.4.3 Вибір мови програмування та середовища розробки.....	21
1.5 Розробка алгоритму обробки даних.....	22
1.5.1 Прийом блоків даних.....	24
1.5.2 Алгоритм обробки блоків інформації.....	26
1.6 Розробка інтерфейсу програми.....	28
1.6.1 Проектування структур даних програми.....	29
1.6.2 Проектування структури модулів програми.....	30
1.7 Тестування розробленої системи.....	32
1.7.1 Інструкція для користувача інтерфейсу.....	32
1.7.2 Зміна мови інтерфейсу.....	34
1.7.3 Перегляд балансу гаманця.....	35
1.7.4 Перегляд останньої транзакції.....	37
1.7.5 Перегляд ціни популярних криптовалют.....	39

					КГ 05. 03 000. 00 ДП ПЗ	Арх.
Зл	Арх.	№ докум	Проект	Дата		5

2	Економічна частина.....	40
2.1	Резюме.....	40
2.2	Визначення трудомісткості розробки програмного забезпечення....	40
2.3	Розрахунок ціни програмного продукту.....	44
3.	Охорона праці.....	46
3.1	Вступ.....	46
3.2	Аналіз та безпека умов праці працівника на робочому місці.....	46
3.3	Розробка заходів з охорони праці.....	47
3.3.1	Вимоги до приміщення.....	47
3.3.2	Мікроклімат робочої зони програміста.....	48
3.3.3	Освітлення робочого місця.....	48
3.3.4	Вплив шуму на програміста.....	49
3.3.5	Виробничі випромінювання.....	50
3.3.6	Електробезпека. Статична електрика.....	50
3.3.7	Організація робочого місця.....	51
3.4	Пожежна безпека.....	52
	Висновки.....	54
	Перелік використаних джерел.....	55
	ДОДАТОК А. Фрагмент листингу програмного модуля обробки транзакцій, балансу гаманця та хендлерів.....	56

Ethereum - це не Біткойн-форк, і він не працює на тому ж Блокчейні, що і Біткойн. Це абсолютно новий Блокчейн з різними технічними складовими, але такий же за ідеєю, як і будь-який інший Блокчейн.



Він заснований на тимчасовій мережі, але замість надання користувачам можливості використовувати низку визначених операцій (наприклад, криптовалютних транзакцій), як це робить Біткойн, протокол Ethereum дозволяє користувачам виконувати практично будь-який код, який вони хочуть.

Подібно протоколу Біткойнів, вузли в протоколі Ethereum жертвують своєю обчислювальною потужністю, щоб підтримувати і оновлювати Блокчейн. Тим не менш, вони також працюють на віртуальній машині Ethereum (EVM).

EVM - це «суперкомп'ютер», який складається з сукупної обчислювальної потужності вузлів в мережі. Цей децентралізований віртуальний суперкомп'ютер використовується для запуску інтелектуальних контрактів (код), представлених користувачами. Потім цей код зберігається в Блокчейні Ethereum і залишається ятам без будь-яких змін (незмінних) назавжди.

1.2 Вибір функціональних елементів для реалізації моделі

Розроблювана система є посередником між цільовим користувачем та мережею криптовалют. З метою забезпечення роботи кінцевого автоматів у

					КГ 05. 03 000. 00 ДП ПЗ	Апр
Зл	Апр	№ докум	Проває	Дата		

процес і взаємодії з інтерфейсом програми також має бути застосовано модуль для зберігання користувацьких даних. Розроблювана система буде складатися з трьох функціональних елементів:

1. Інтерфейс користувача;
2. Інтерфейс локальної бази даних;
3. Інтерфейс роботи з мережею криптовалют.

Взаємодія елементів та інтерфейсів розроблюваної системи представлена на рисунку 1.1.

Взаємодія користувача та сервісу, починається з привітання та пропозиції ввести адресу мережі криптовалют для відстежування з боку програми.

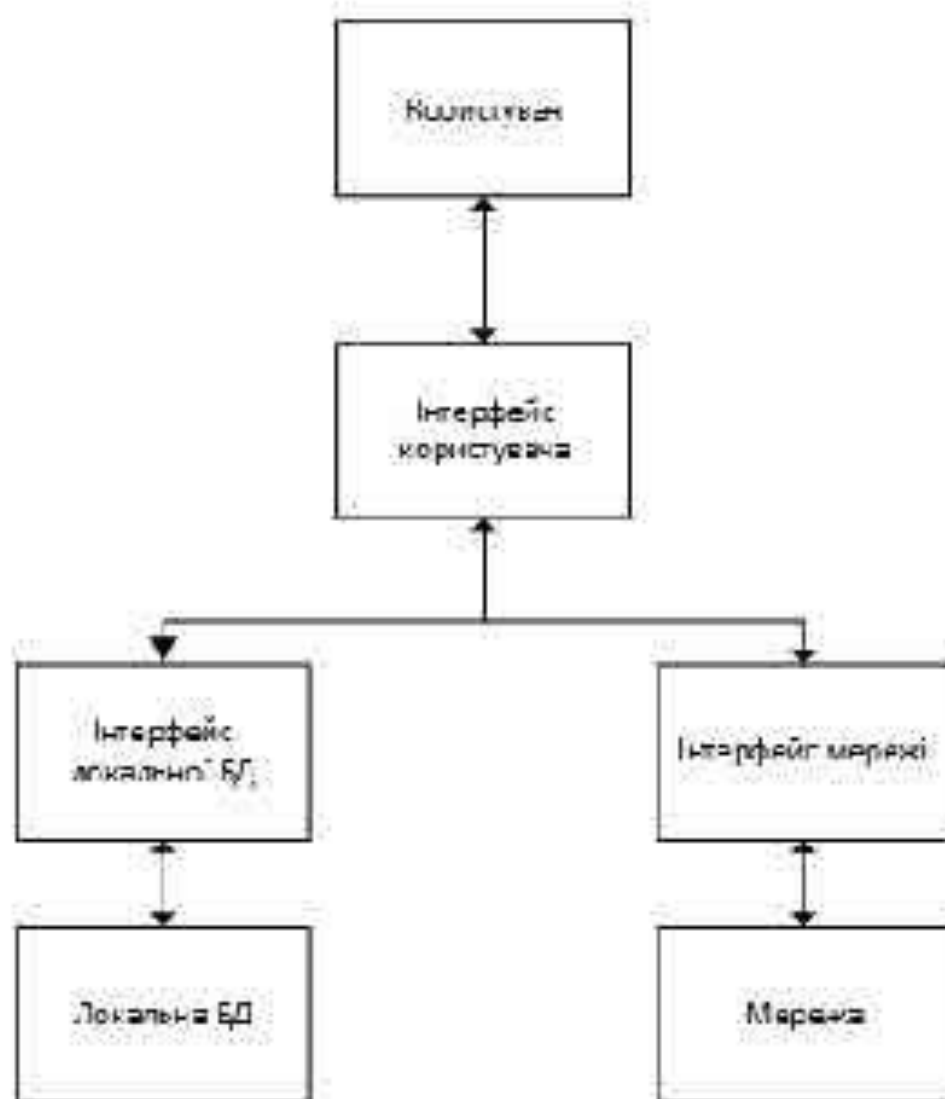


Рисунок 1.1. Взаємодія елементів та інтерфейсів розроблюваної системи

Зл	Арх	№ докум	Проект	Дата

КГ 05. 03 000. 00 ДП ПЗ

Арх

12

сторона. Згідно з etherscan.org, його слід розглядати як паливо для додатків у децентралізованій мережі Ethereum.

Це абстрактний спосіб сформулювати функцію Ефіру, і конкретний приклад може допомогти прояснити ситуацію. Скажімо, у мережі Ethereum є програма, яка дозволяє створювати, змінювати та видаляти прості нотатки. Щоб виконати будь-яке з цих завдань, програмі потрібна потужність обробки через мережу.

Щоб покрити вартість цієї потужності, вам, ймовірно, доведеться сплачувати граничну комісію щоразу, коли ви захочете внести будь-які зміни у свої наявні нотатки. Ефір – це токен, за допомогою якого ви здійснюєте цей платіж. У певному сенсі це цифрове масло, оскільки воно дозволяє мережі обробляти внесені вами зміни. Як вид палива, має сенс, що комісія за транзакцію Ether буде різною залежно від того, скільки палива потрібно для послуги.

Кожна конкретна дія в мережі Ethereum або в децентралізованому додатку вимагає різної обчислювальної потужності та часу. Чим більше потрібна потужність і час, тим вище ефірна плата за виконання дії. Таким чином, ефір відрізняється від цифрової валюти, як от біткойн.

1.3 Визначення задач проектування

1.3.1 Мінімізація порогу входу

Мінімізувати поріг входу нових користувачів для використання програми

Поріг входження це мінімальна кількість теоретичних знань в предметній області, щоб розпочати практичну частину. Тобто, задача стоїть у тому, щоб людина, що навіть ніколи не користувалася ботами за не знайома з подібним функціоналом одразу змогла отримати усю потрібну їй інформацію без звернення за допомогою до документацій, інструкцій тощо.

Говорити по-простому, це сходинка, з якої зручно і комфортно починати якусь підприємство справа. Припустимо, людина хоче інвестувати певну суму у що-небудь ліквідне, що принесе швидкий дохід. (моча в нерухоість). Провісник якось розвідує, він розуміє, що за теперішніх 2,5 тисяч доларів за квадратний метр - виплутуватися рано, але спостерігається тенденція до падіння (безробіття, відтік

					КГ 05. 03 000. 00 ДП ПЗ	Ара
За	Ара	№ докум	Проває	Дата		14

ціни. Коли BTC досягає верхньої або нижньої позначки, бот надсилає вам повідомлення.

RocketCoin

Цей бот також про статистику, але з трохи розширеним функціоналом. У нього є команди:

`/status` - Тут можна переглянути статистику з основних монет:

`top5` — просто ціна перших п'яти монет за рівнем капіталізації

`/global` - загальна інформація про капіталізацію, домінування BTC, обсяг торгів і т.д. Так, це все можна подивитися в Coinmarketcap, але особисто я весь робочий час перебуваю в Telegram і зручно, коли все перебуває під рукою.

Simple Crypto Broker Bot

Цей дуже простий бот дозволяє вам заводити, виводити та оперативно обмінювати монети за актуальним курсом, надає докладну інформацію про угоди. Дозволяє здійснювати як довгі так і короткі угоди, відстежує ліквідність. Також є демо майданчик.

`/long` - покупка, довга позиція, дозволяє заробити під час зростання курсу

`/sell` - продаж, коротка позиція, дозволяє заробити на зниженні курсу

TrackxBot

Цей бот дозволяє вам відслідковувати транзакції по гаманцю. Завдяки цьому боту ви без проблем дізнаєтеся першочергово те, що вам зайшли гроші чи отримали токени по ICO отримали токени по AirDrop і т.д. А ще можна додавати кілька гаманців і давати їм людські імена для того, щоб не згадувати набір цифр. Бот використовує лише відкриті ключі та дозволяє відстежувати BTC, BCH, ETH, ERC20 та інші крипто валюти.

Crypto Exchange Listing

Передбачає листинг токенів на біржах до офіційного оголошення.

Tokenstats Announcements

Повідомляє вас про нові токени за допомогою особистих повідомлень, відправлених ботом. Ще один простий, але надзвичайно корисний інструмент, який виглядає безцінним для інвесторів ICO. Введіть ROI, а потім аббревіатуру

										Арс
										17
Зл	Арс	№ докум	Проває	Дата	КГ 05. 03 000. 00 ДП ПЗ					

токена, і вам буде показано повернення, здійснене в ETH, BTC і USD. Не всі боти Telegram вимагають ручного налаштування та конфігурації за допомогою ряду команд: інші просто вимагають, щоб ви приєдналися до каналу та слідкуйте за ними. Tokenstats Announcements сповіщає вас про нові токени, в той час як Crypto Exchange Listing «прагне точно передбачити розміщення нового токена на біржі до офіційного оголошення». Його рекорд, який базується на депозитах токенів на відомі біржові гаманці, напрочуд хороших.

Порівнянок функціональних аналогів

Характеристика	Etherscan	Ethereum Token Explorer Bot
Тип інтерфейсу	Веб-сайт	Чат-бот
Спосіб відображення даних	Відображення інформації з локальної бази даних з використанням технології зменшення часу для звернення до сховища та очікування на відповідь	Відображення інформації з локальної бази через пряме звернення
Можливість фільтрації	Так	Обов'язкова фільтрація

1.4 Основні етапи проектування

1.4.1 Вибір типу логіки взаємодії з мережею криптовалюти

Для взаємодії з потрібними мені даними криптовалютного ринку я планую використовувати біржу під назвою CoinGecko. А саме використання його публічних API. Такий підхід не потребує від розробника-користувача високопродуктивної комп'ютерної системи або великого об'єму пам'яті. Уся інформація отримується лише за певним запитом до дистриб'ютора. Відповідно, даний підхід з апаратної частини потребує лише стабільного зв'язку з мережею Інтернет.

CoinGecko є одним із сайтів для розміщення живих цін на криптовалюту. Окрім цього, CoinGecko вітає нових користувачів криптовалют у спільноті

навчальними матеріалами, підрахунками прибутку та інтерактивними розіграшами. Коли вперше потрапляє на CoinGecko, біткойн за замовчуванням стає провідною криптовалютою з найбільшою ринковою капіталізацією та мережею. Проте є тисячі монет, які можна знайти. Без правильних інструментів для повернення пошуку перегляд проєктів за проєктом може бути виснажливим. Ось чому CoinGecko класифікує усі різні характеристики токенів, щоб користувачі могли бачити безліч різних точок даних та інформації, що стосується крипторинсів.

Для подання інформації щодо стану віртуальних рахунків та транзакцій планується використання бібліотеки Etherscan.

Мінімальний, але повний API Python для Etherscan.io .

Etherscan є провідною платформою для дослідження блокчейну, пошуку, API та аналізу для Ethereum, децентралізованої платформи смарт-контрактів.

Для забезпечення рівноправного доступу до даних блокчейну ми розробили API для розробників Etherscan, щоб надати розробникам прямий доступ до даних та служб провідника блоків Etherscan за допомогою запитів GET/POST.

API Etherscan надається як соціальна робота та без гарантії, тому використовуйте те, що вам потрібно, і не більше.

Для цього ви можете знайти відповідну кінцеву точку з нашого цифрового спектру API, до яких ви можете просто запитати, щоб надати дані, залишаючись повністю зосередженими на створенні своїх програм.

Etherscan пропонує 2 типи планів API, набір безкоштовних кінцевих точок спільноти та API PRO, який складається з додаткових отриманих даних блокчейну.

Надаються всі стандартні та професійні кінцеві точки. Також підтримуються тестові мережі Kovan, Rinkeby і Ropsten.

1.4.2 Огляд способів реалізації інтерфейсу користувача

Користувальницький інтерфейс (UI) — це функції пристроїв або програми, які дозволяють користувачеві взаємодіяти з ним. Це полегшує вашій цільовій аудиторії взаємодію з нею.

Очевидним способом є створення власного мобільного додатка. Перевагою

										Ара
Зл	Ара	№ докум	Провак	Дата	КГ 05. 03 000. 00 ДП ПЗ					19

самостійно визначає свої наміри і ця інформація допомагає перенавчати модель машинного навчання, щоб покращити точність бота. У цьому випадку, залишається можливість помилки, якщо користувачі обирають варіанти, що направляють бота у непрапорний бік.

Обробку даних, що приймає бот можна розділити на два етапи:

1. Прийом повідомлення
2. Обробка інформації та відповідь

1.5.1 Прийом блоків даних

Оскільки перший етап полягає у взаємодії користувача з мережею, то він має виконуватися з мінімальним навантаженням для користувача. Це означає, що алгоритм його роботи треба створити як можна більш лінійним та не внаслідок цього до ресурсів, аби підпрограма могла заздалегідь отримати належну і інформацію.

					КГ 05. 03 000. 00 ДП ПЗ	Арх.
Зл	Арх.	№ докум	Підпис	Дата		34



Рисунок 1.3 Алгоритм прийому блоків даних

Зл	Ар	№ розр	Підпис	Дата

КГ 05. 03 000. 00 ДП ПЗ

Ар

25

Як видно з рисунку 1.3, алгоритм має лише одне розгалуження, що відповідає за цикл, у якому виконується основний код програми. Такий підхід є досить продуктивним, оскільки виконання кожної з дій циклу не потребує великих обчислювальних та часових затрат.

1.5.2 Алгоритм обробки блоків інформації

Другий етап, на відміну від першого є вже досить залежним від ресурсів, і потребує витрат як часових, так і обчислювальних.

Головним чинником на час виконання підпрограми впливає особливість використаного API. У моєму випадку це CoinGecko або Etherscan. При запиті сервіс відправляє відповідь згідно з заданими користувачем параметрами.

Аби мінімізувати кількість запитів до сервісу, а також зменшити навантаження на обчислювальну машину, можна встановити штучну часову затримку. Для цього буде використовуватися асинхронний підхід до написання програми. Це означає, що даний "історич" є актуальним виключно для конкретного блоку і ніяк не впливає на роботу програми в цілому.

Також, є не обхідність у отриманні внутрішніх транзакцій. І поставляють я вони окремо через певні свої особливості, а це у свою чергу означає додаткові звернення до API. Тому це збільшує витрати у часі.

Одна з вищезазначених особливостей це є походження цих транзакцій. Вони породжені зовнішніми, тому що бувають такі ситуації, що в блоці немає жодного грошового переказу, запит щодо внутрішніх транзакцій не завжди є доцільним.

Після об'єднання в один список усіх транзакцій, залишається перевірити всі адреси, чи відстежується вона хоча б одним користувачем програмного продукту. Якщо користувача знайдено – наділати йому повідомлення.

					КГ 05. 03 000. 00 ДП ПЗ	Арх.
Зл	Арх.	№ докум	Проває	Дата		№

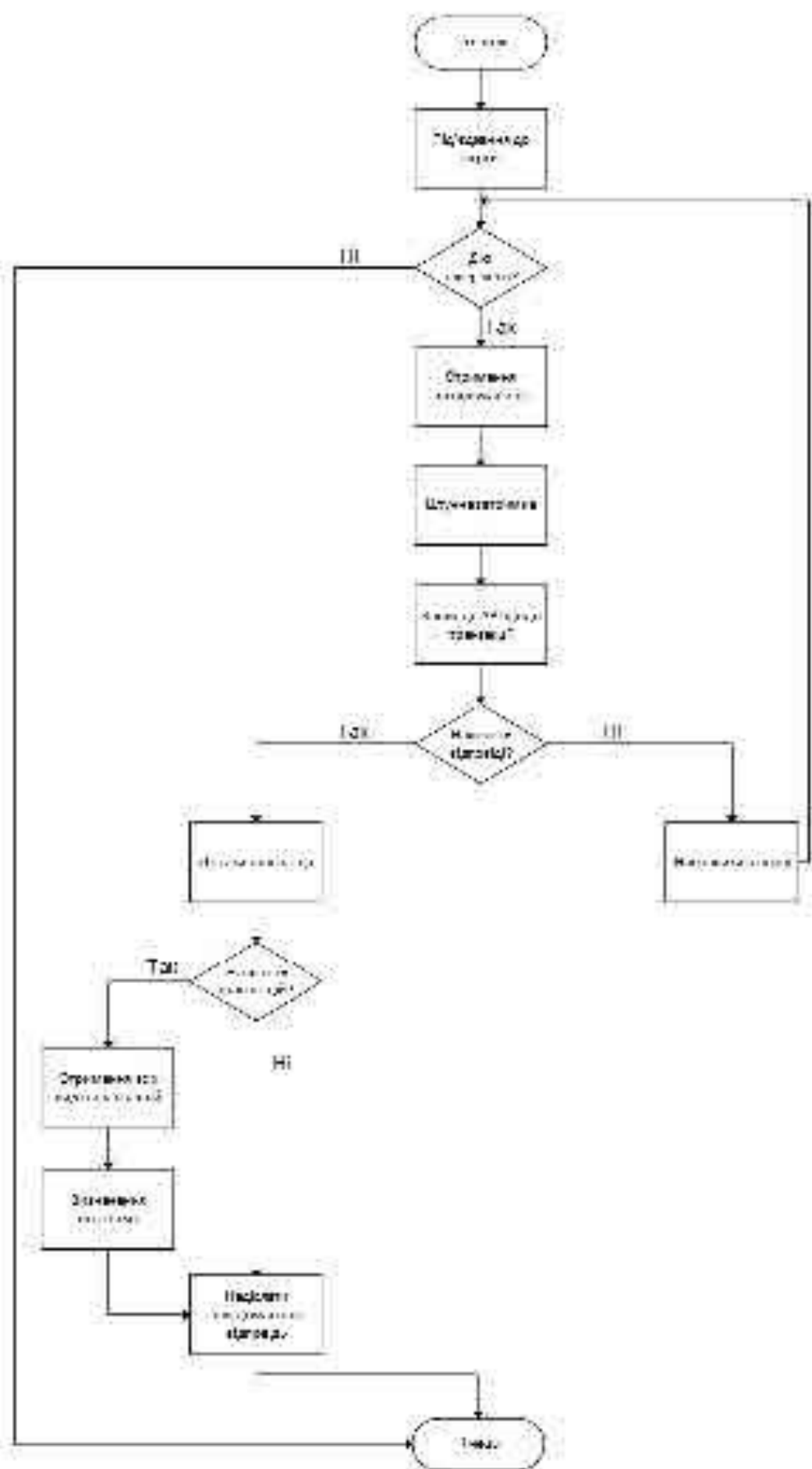


Рисунок 1.4 Алгоритм обробки

З.п.	А.р.	№ докум.	Підпис	Дата

КГ 05. 03 000. 00 ДП ПЗ

А.р.

27

1.6 Розробка інтерфейсу програми

Інтерфейс програми збудований на технології чат-ботів у месенджері Telegram. Цей підхід характеризується як подібність до діалогу між реальною людиною/користувачем та програмним додатком за чітко визначеним алгоритмом, або, простіше кажучи, сценарієм. Фактично, цей сценарій є повноцінним автоматом, оскільки він має власні умови та правила, за якими відбуваються переходи у інтерфейсі. Також використовується внутрішня пам'ять для збереження проміжних даних.

Інтерфейс програми має бути інтуїтивно зрозумілий користувачу для вдалої взаємодії з користувачем, у випадку цього проекту для взаємодії зрозумілий сценарій реєстрації адреси цільової мережі криптовалют. Під час діалогу додаток має отримати власне адресу для відстежування для відображення інформації, що необхідно в процесі експлуатації.

Даний алгоритм зображено на рисунку 1.5.

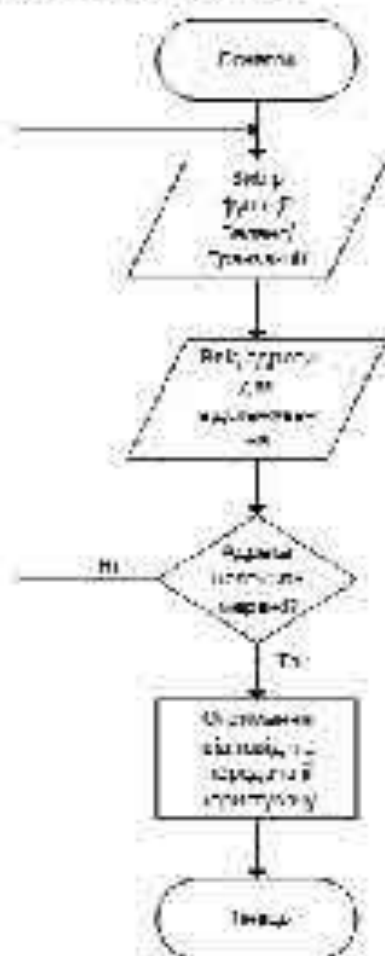


Рисунок 1.5 Алгоритм роботи інтерфейсу

Зл	Арх	№ докум	Проект	Дата

КГ 05. 03 000. 00 ДП ПЗ

Арх

22

1.6.1 Проектування структур даних програми

Інтерфейс програми для повноцінної взаємодії з користувачем потребує сховища даних з принципом зберігання ключа та значення. Так як у якості сховища використовується Redis, до якого потрапляють об'єкти json у строковому представленні, доцільно вказати типи даних до конвертації. Данні наведені у таблиці 1.1.

Таблиця 1.1

Назва	Тип	Призначення
language	str	Зміна мови інтерфейсу
wallet	str	Адреса гаманця
coin	str	Вибір криптовалюти

Інтерфейс може бути на трьох мовах: українській, англійській та російській. Обраний варіант впливає на подальшу мову взаємодії з користувачем, якщо є така можливість.

Адреса гаманця, назва валюти є проміжними даними, тому не займають пам'яті довгий час.

Поле історії транзакцій необхідно для мінімізації звернень програми до API сторонніх сервісів. Це не лише допомагає уникнути блокування, але й пришвидшує роботу інтерфейсу. Щойно користувач повертає меню перегляду історії транзакцій – дані видаляються.

Основні дані програми

Основні дані – це дані, що так чи інакше підлягають довготривалому утриманню. Данні наведені у таблиці 1.2

Таблиця 1.2

Назва	Тип	Призначення
user_id	bigint	Унікальний ідентифікатор користувача
language	varchar	Мова інтерфейсу користувача



Рисунок 1.7 Файлы проекта

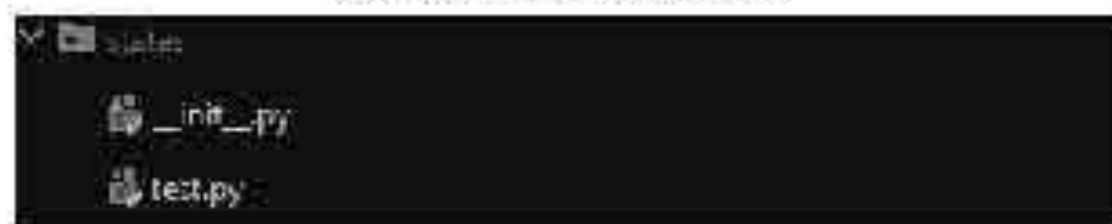


Рисунок 1.8 Файлы статус

Зл	А р	№ о з	П р о	Д а

КГ 05. 03 000. 00 ДП ПЗ

1.7 Тестування розробленої системи

1.7.1 Інструкція для користувача інтерфейсу

Як і у більшості випадків, необхідно надіслати команду “/start”. Це стандартна команда для всіх віртуальних «спікерозмовників» у месенджері Telegram. Отримавши вище зазначену команду, бот відповідає вітальним повідомленням.

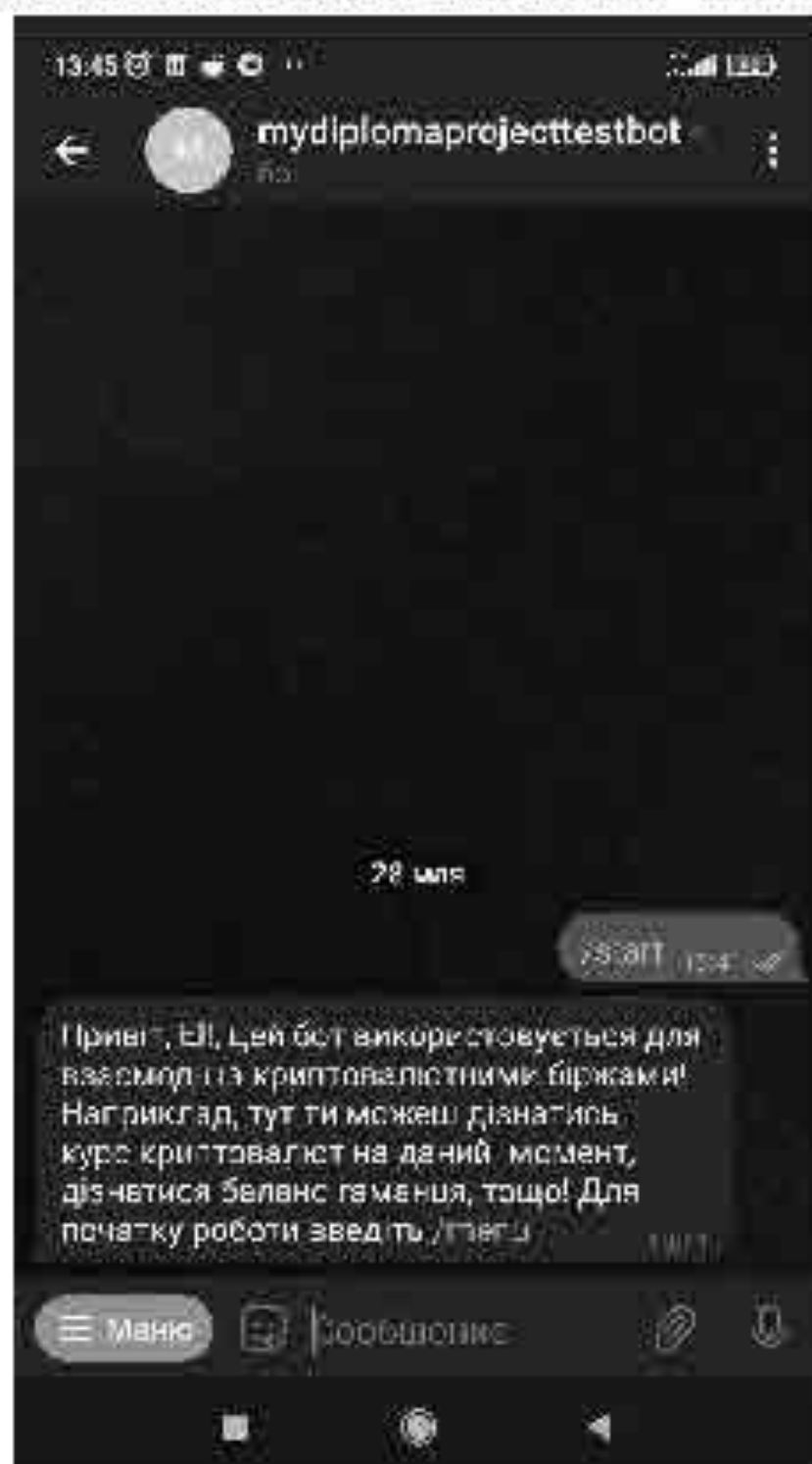


Рисунок 1.9 Привітання

					КГ 05. 03 000. 00 ДП ПЗ	Ар.
Зл	Ар.	№ докум	Проває	Дата		32

Для початку роботи натискаємо на підсвічену команду *Меню*, та отримуємо у своє розпорядженні меню боту



Рисунок 1.10 Меню

1.7.3 Перегляд балансу гаманця

Щоб переглянути баланс будь-якого гаманця, що вас цікавить, потрібно натиснути на кнопку «Дізнатися баланс гаманця», після чого бот попросить вас вказати унікальний номер гаманця, після чого відправить ці дані до API, а отримавши відповідь – передасть її вам у потрібному та зручному форматі.

Щоб перевірити правильність роботи бота я буду використовувати відкриту базу гаманців та їх стану Etherscan.

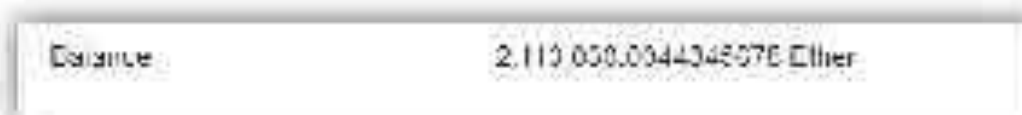
Тому спочатку ми повинні обрати гаманець, що нас цікавить



№	Адреса	Мітка	Баланс	Розмір	Статус
1	0x1234567890123456789012345678901234567890	0x1234567890123456789012345678901234567890	0.000000000000000000	0.000000000000000000	0.000000000000000000
2	0x9876543210987654321098765432109876543210	0x9876543210987654321098765432109876543210	0.000000000000000000	0.000000000000000000	0.000000000000000000
3	0xABCDEF0123456789012345678901234567890123	0xABCDEF0123456789012345678901234567890123	0.000000000000000000	0.000000000000000000	0.000000000000000000
4	0x11	0x11	0.000000000000000000	0.000000000000000000	0.000000000000000000
5	0x22	0x22	0.000000000000000000	0.000000000000000000	0.000000000000000000
6	0x33	0x33	0.000000000000000000	0.000000000000000000	0.000000000000000000
7	0x44	0x44	0.000000000000000000	0.000000000000000000	0.000000000000000000
8	0x55	0x55	0.000000000000000000	0.000000000000000000	0.000000000000000000

Рисунок 1.12 Найцінніші гаманці Ефіру

Дивимося баланс



Баланс	2.113 000.0044345076 Ether
--------	----------------------------

Рисунок 1.13 Баланс гаманця

Перевіримо у бота



Рисунок 1.14 Баланс гаманця

І як бачимо, всі данні збігаються, тільки у боті число є більш округленим.

						КГ 05. 03 000. 00 ДП ПЗ	Ар
Зл	Ар	№ рахуку	Пішов	Дата			Зл

1.7.4 Перегляд останньої транзакції

Щоб переглянути транзакції будь-якого гаманця, що вас цікавить, потрібно натиснути на кнопку «Перевірити останню Ethereum транзакцію», після чого бот попросить вас вказати унікальний номер гаманця, після чого він знову відправить ці дані до API, а отримавши відповідь – передасть її вам.

Перевірити роботу будемо у цього же гаманця за допомогою Etherscan.



Рисунок 1.15 Транзакція Etherscan

Тепер переір'яємо роботу боту:

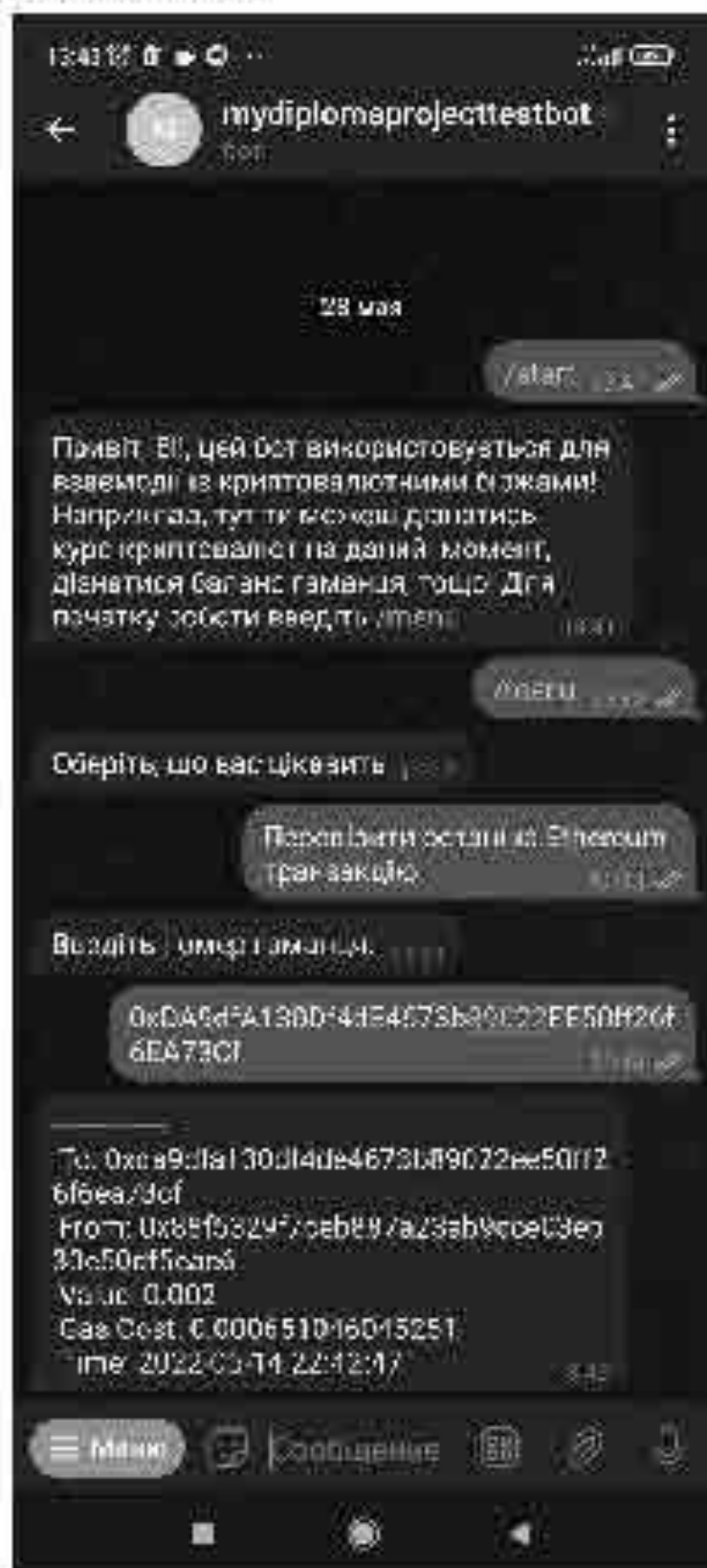


Рисунок 1.16 Транзакція у боті

Як бачимо, усі дані цілком збігаються. Відрізняється лише час тому, що я використовуву часовий пояс UTC +3.

					КГ 05. 03 000. 00 ДП ПЗ	Ар.
Зл	Ар.	№ раху	Пішов	Дата		28

1.7.5 Перегляд ціни популярних криптовалют

Щоб переглянути ціну на найпопулярніші криптовалюти потрібно натиснути на кнопку «Дізнатися курс криптовалют», після чого ви отримаєте ціле меню з вибором, яка саме криптовалюта вас цікавить. Після вибору ви отримаєте ціну в доларах на даний момент.



Рисунок 1.17 Перевірка роботи курсу криптовалют

Зл	Агр	№ докум	Проває	Дата

КГ 05. 03 000. 00 ДП ПЗ

2 ЕКОНОМІЧНА ЧАСТИНА

2.1 Резюме

В даному дипломному проекті розроблені програмні засоби для розробки системи сповіщень про нові події мережі цільової криптовалюти.

Ефективність кожного програмного продукту визначається його якістю та ефективністю процесу розробки. Якість ПП визначається наступними складовими: з точки зору користувача; з позиції використання ресурсів; виконання вимог до програмного забезпечення.

Оцінка якості програмного продукту з точки зору користувача визначається необхідним на стадії функціонування розміром оперативної пам'яті ЕОТ, витратами машинного часу, пропускнув спроможністю каналів передачі даних. Оцінка якості програмного продукту включає визначення трудомісткості і вартості його створення.

2.2 Визначення трудомісткості розробки програмного забезпечення

Тривалість розробки програмного продукту залежить від його обсягу, трудомісткості розробки, кваліфікації виконавців, а також планових термінів, визначених умовами ринку. Методом структурної аналогії по відповідних категоріях аналогів програмного забезпечення визначається обсяг програмних засобів, у тис. знах умовних машинних команд програми аналога

Таблиця 2.1 Каталог аналогів

Найменування ПП	Обсяг функції ПП – V_n , усл. машинних команд.
1. ПП агрегатованих розрахунків	1300 – 8600
2. ПП загальної математики і ПП імітаційного моделювання	7800 – 8800
3. ПП організації обчислювального процесу	13000 – 10200

У таблиці 2.1 представлені аналоги програмного забезпечення, функції яких, у більшому або меншому ступені, виконує розроблений програмний продукт. Для нашого варіанта виділено сірим кольором.

Вибравши аналог ПП, що містить V_0 в умовних машинних командах, трудомісткість визначати на основі табл. 2.2

Таблиця 2.2

Обсяг ПП, тис умов машинних команд	Норма часу, люд/год
1.00	229
2.00	244
3.00	262

На підставі отриманого значення, по довіднику, визначається укрупнена норма часу на розробку аналога програмного забезпечення (коректується поправочним коефіцієнтом враховуючої умови розробки ПП, тобто в умовах комп'ютера, $K_k=0,7+0,8$): $T^* = 229 \times 0,8 = 183,2$ (люд/год).

Трудомісткість програмного продукту визначається по кожному етапу розробки окремо на підставі трудомісткості аналога з урахуванням складності розробки, ступеня новизни і ступеня використання в розробці стандартних модулів на підставі формул:

$$T_m = T^* \rho \times L_1 \times K_n \quad (2.1)$$

$$T_m = T^* \rho \times L_2 \times K_n \quad (2.2)$$

$$T_m = T^* \rho \times L_1 \times K_n \times K_r \quad (2.3)$$

Для розрахунку необхідні наступні коефіцієнти:

L_1 – питома вага іго етапу розробки (див. табл. 2.2.);

K_n – поправочний коефіцієнт, що враховує ступінь новизни (див. табл. 2.3.);

K_r – поправочний коефіцієнт, що враховує ступінь використання в розробці типових програм (див. табл. 2.4.).

Таблиця 2.2 Значення питомих коефіцієнтів трудоміткості стадії в загальній трудоміткості робіт ПП

Код стадії	Ступінь новизни		
	А	Б	В
ТЗ (L ₁)	0,15	0,12	0,12
ТП (L ₂)	0,16	0,15	0,11
РП (L ₃)	0,55	0,58	0,61

Для цього варіанта виділено сірим кольором.

Таблиця 2.3 Значення поправочного коефіцієнта, що враховує ступінь новизни

Код ступеня новизни	Ступінь новизни	Значення K _н
А	Принципово нові ПО	1,75 - 1,2
Б	ПО - розвиток визначеного параметричного ряду	1,0 - 0,8
В	ПО мавчині аналог	0,7

Для цього варіанта виділено сірим кольором.

Таблиця 2.4 Значення коефіцієнта ступеня використання в розробці типових програм

Ступінь охоплення реалізованих функцій розроблявального ПО типовими програмами, %	Значення K _р
60 і вище	0,6
40-60	0,7
20-40	0,8
До 20	0,9

Для цього варіанта виділено сірим кольором.

Тепер розраховуємо трудоміткість по кожному етапу окремо:

Трудоміткість технічного завдання

$$T_{\text{т}} = T_{\text{а}} * L_1 * K_{\text{а}} = 160,3 * 0,12 * 0,8 = 15,38 \text{ (люд/годик)} \quad (2.1)$$

Трудоємність розробки технічного проекту

$$T_{\text{тп}} = T_{\text{а}} * L_2 * K_{\text{а}} = 160,3 * 0,11 * 0,8 = 14,11 \text{ (люд/годик)} \quad (2.2)$$

Трудоємність розробки робочого проекту

$$T_{\text{рп}} = T_{\text{а}} * L_3 * K_{\text{а}} * K_{\text{р}} = 160,3 * 0,61 * 0,8 * 0,8 = 62,58 \text{ (люд/годик)} \quad (2.3)$$

Для подальших розрахунків визначили кількість пагера, витраченого на кожен етап: технічне завдання $N_{\text{тз}} = 2$ (стр), розробка ТП $N_{\text{тп}} = 20$ (стр), розробка робочого проекту $N_{\text{рп}} = 18$ (стр), пояснювальна записка відповідно $N_{\text{оз}} = 29$ (стр). Розрахунок зведений у таблицю 2.5.

Таблиця 2.5 Розрахунок трудоємності ПП

Найменування етапів	Розрахунок, годик		
	2	3	4
1.ТЗ	$T_{\text{тз}} = 15,39$	$T_{\text{м}} = 0,7 * N_{\text{тз}} = 0,7 * 2 = 1,4$	$T_{\text{м}} = 0,15 * N_{\text{тз}} = 0,15 * 2 = 0,30$
2.Розробка ТП	$T_{\text{тп}} = 14,12$	$T_{\text{м}} = 0,7 * N_{\text{тп}} = 0,7 * 20 = 14,0$	$T_{\text{м}} = 0,15 * N_{\text{тп}} = 0,15 * 20 = 3,0$
3.Розробка РП	$T_{\text{рп}} = 54,76$	$T_{\text{м}} = 0,7 * N_{\text{рп}} = 0,7 * 18 = 12,5$	$T_{\text{м}} = 0,15 * N_{\text{рп}} = 0,15 * 18 = 2,7$
4.Розробка ПЗ	$T_{\text{оз}} = 1,5 * N_{\text{оз}} = 1,5 * 29 = 43,5$	$T_{\text{м}} = 0,7 * N_{\text{оз}} = 0,7 * 29 = 20,3$	$T_{\text{м}} = 0,15 * N_{\text{оз}} = 0,15 * 29 = 4,4$
Усього, в т.ч.:	$\Sigma T = 186,3$		
- на розробку	$\Sigma T_{\text{р}} = 127,7$		
- контроль керівника		$\Sigma T_{\text{м}} = 48,2$	
- нормоконтроль			$\Sigma T_{\text{м}} = 10,4$

З.о.	А.р.	№ розр.	Підпис	Дата

КГ 05. 03 000. 00 ДП ПЗ

А.р.

43

2.3 Розрахунок ціни програного продукту

У цьому розділі для визначення ціни розраховуємо основну заробітну плату виконавців, матеріальні витрати, вартість машинно – години і витрати на розробку ПО. Розрахунок основної заробітної плати виконавців приведений у таблиці 4.6. Відповідно до статті 8 «Закону про Державний бюджет України на 2022» встановлено мінімальну заробітну плату умисляному розмірі з 1 січня 2022 року - 6500 гривень; мінімальну погодинну тарифну ставку – 39.26 грн.

Таблиця 2.6 Розрахунок основної заробітної плати виконавців

Найменування робіт	Трудомісткість в робіт, години	Погодинна тарифна ставка, грн.	Розрахунок, грн
1. Розробка ПП	128	40,00	5120,00
2. Контроль керівника	48	60,00	2880,00
3. Нормоконт-роль	10	60,00	600,00
Усього	-	-	8600,00

Зробимо розрахунок матеріальних витрат на розробку ПП. Розрахунок зведемо в таблицю 2.7

Таблиця 2.7 Розрахунок матеріальних витрат на розробку ПО

Найменування матеріальних витрат	Тип, модель	Кількість	Ціна одиниці, грн.	Вартість, грн.
Папір	Лист А4	70	2,00	140,00
Разом	-	-	-	
Транспортно – заготівельні витрати (10%)				14,00
Усього				154,00

На підставі отриманих даних по окремих статтях витрат складена калькуляція планової собівартості в цілому ПП за формою, приведеною в таблиці 2.8.

Таблиця 2.8 Розрахунок статей витрат планової собівартості

Стаття витрат	Значення, грн.	Формула розрахунку
1. Матеріали	154,00	V_m (див. табл. 4.7)
2. Основна заробітна плата	8600,00	Z_o (див. табл. 4.6)
3. Додаткова заробітна плата	1290,00	$Z_d = 0,15 \times Z_o = 0,15 * 8600,00$
4. Відрахування до єдиного фонду соціального внеску	2175,80	$V_{с.в.} = 0,22 \times (Z_o + Z_d) = 0,22 * (8600,00 + 1290,00)$
5. Накладні витрати	2580,00	$V_{н.к.} = 0,3 \times Z_o = 0,3 * 8600,00$
6. Пошта собівартість	14799,80	$C_{sum} = V_m + Z_o + Z_d + V_{с.в.} + V_{н.к.} =$

Розмір прибутку, що включається в ціну, визначаємо по наступній формулі:

$$П = (C_o * P) / 100 = (14799,80 * 10) / 100 = 1479,98 \text{ грн} \quad (2.4)$$

Де P – плановий рівень рентабельності (10-15%).

Оптова ціна (кошторисна вартість) визначається по формулі:

$$C_o = C_{sum} + П = 14799,80 + 1479,98 = 16279,78 \text{ грн} \quad (2.5)$$

Податок на додану вартість визначаємо по наступній формулі:

$$ПДВ = 0,2 * C_o = 0,2 * 16279,78 = 3255,96 \text{ грн} \quad (2.6)$$

Виходячи з отриманих даних, ціна реалізації розробленого програмного продукту на основі наступної формули, становитиме:

$$C_p = C_o + ПДВ = 16279,78 + 3255,96 = 19535,74 \text{ грн} \quad (2.7)$$

3 ОХОРОНА ПРАЦІ

3.1 Вступ

Питання охорони праці людині потребує рішення на всіх стадіях трудового процесу незалежно від виду професійної діяльності. Забезпечення безпеки і здорових умов праці в більшості залежить від правильної оцінки небезпечних, шкідливих виробничих факторів.

Одноразові по складності зміни в організмі людини можуть бути викликані різними причинами. Це може залежати від виробничого середовища, надмірного фізичного і розумового навантаження, нервово-емоційна напруги, а також різних сполучень цих причин.

Умови праці на робочому місці програміста-розробника мають відповідати вимогам безпеки праці, у відповідності з нормативно-правовим актом з охорони праці (НПАОП 0.00-7.15-18) «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями», а також НПАОП 0.00-1.28-10 "Правила охорони праці під час експлуатації електронно-обчислювальних машин", затверджені наказом Державного комітету України з промислової безпеки, охорони праці та гірничого нагляду від 26.03.2010 №65.

У даному розділі дипломного проекту розглядається питання охорони праці програміста на стадії розробки нової системи сповіщень про нові події мережі цільової криптовалюти, яка взаємодіє з користувачем за допомогою власного інтерфейсу.

Лабораторія, у якій працює програміст знаходиться на підприємстві ТОВ «2С».

3.2 Аналіз та безпека умов праці працівника на робочому місці

Аналіз умов праці доводить, що у приміщенні лабораторії на програміста можуть мати негативний вплив наступні фізичні та психофізіологічні фактори:

- підвищена або знижена вологість повітря;

					КГ 05. 03 000. 00 ДП ПЗ	Апр
Зл	Апр	№ розр	Проває	Дата		46

- підвищена або знижена температура повітря робочої зони;
- недостатня освітленість робочого місця;
- підвищений рівень шуму на робочому місці;
- підвищений рівень електромагнітних випромінювань;
- підвищена іонізація повітря;
- фізичні перевантаження (одноманітна поза викликає статичну втому);
- нервово-психічні перевантаження (розумова перенапрута, перенапрута аналізаторів);

Рекомендації, щодо усунення шкідливих факторів або максимальне зниження їх впливу на працівника наведеного у роботі:

- Для зменшення втоми очей рекомендується використання правильного освітлення – гарне природне освітлення, у темний час доби лампа повинна освітлювати тільки кімнату, але не екран монітора – це допоможе уникнути відблисків, що ускладнюють роботу.

- До рекомендації стосовно зменшення навантаження нервової системи відносяться прогулянки на свіжому повітрі, спілкування з людьми у реальному світі та необхідні перерви при роботі з комп'ютерною технікою.

- Для зменшення шкідливого впливу незручної пози та довгого знаходження у сидільному положенні, необхідний правильний підбір робочої меблі. Вона збереже правильне положення тіла під час роботи за комп'ютером і зменшить навантаження на м'язи.

3.3 Розробка заходів з охорони праці

3.3.1 Вимоги до приміщення

Робоче приміщення програміста, має загальну площу 20 м², висоту стелі 3 м. У приміщенні знаходиться 7 робочих місць з ПК. Слід відзначити, що площа одного робочого місця оператора ПК не повинна бути меншою за 6м², а об'єм не менший за 20м³ [1], тобто площі та об'єму даного приміщення не вистачає для

					КГ 05. 03 000. 00 ДП ПЗ	А пр.
Зл	А пр.	№ розр.	Підпис	Дата		47

розташування 7 робочих місць операторів ПК. Забороняється встановлювати комп'ютери в приміщеннях, розташованих у підвалах будівель. Приміщення укомплектоване системами центрального опалення та кондиціювання повітря. Також в приміщенні присутні аптечки першої домедичної допомоги.

3.2.2 Мікроклімат робочої зони програміста

Оскільки робота за комп'ютером характеризується малими фізичними навантаженнями, то цей вид діяльності належить до категорії легких робіт за критерієм енерговитрат організму Іа, Іб, тому повинні дотримуватися наступні вимоги згідно ДСН 3.3.6.042-99:

Норми мікроклімату для приміщень з ВДТ ЕОМ та ПЕМ

Пора року	Категорія робіт	Температура повітря, С, не більше	Відносна вологість повітря %	Швидкість руху повітря, м/с
Холодна	Легка- Іа	22-24	40-60	0,1
	Легка- Іб	21-23	40-60	0,1
Тепла	Легка- Іа	23-25	40-60	0,1
	Легка- Іб	22-24	40-60	0,1

Ці вимоги виконуються.

3.2.3 Освітлення робочого місця

В приміщенні, де використовується праця на комп'ютері, наявні елементи природного та штучного освітлення.

Нормованим параметром природного освітлення згідно ДБН В.2.5-28 – 2006 є коефіцієнт природного освітлення (КПО). КПО встановлюється в залежності від розряду виконуваних зорових робіт. Робота програміста відноситься до робіт середньої точності (IV розряд зорових робіт, мінімальний розмір об'єкту розрізнення складає 0,5-1 дмм), для яких при використанні бокового освітлення КПО=1,5%.

Для освітлення приміщення, у якому працює програміст, використовується змішане освітлення, тобто сполучення природного й штучного освітлення. Для

						КГ 05. 03 000. 00 ДП ПЗ	А пр.
Зл	А пр.	№ розр.	Проває	Дата			«

загального освітлення приміщення, де перебуває робоче місце програміста, використовуються газорозрядні лампи типу ЛД.

Робоче місце, обладнане ПК повинно бути розташоване так, щоб уникнути попадання в очі прямого сонячного світла.

Штучне освітлення здійснюється системою загального рівномірного освітлення. Застосування світильників без розсіювачів та екрануючих стжок забороняється. Рівень освітленості на робочому столі в зоні розташування документів має бути в межах 300–500 лк. Вимоги виконані.

3.2.4 Вплив шуму на програміста

Як було показано вище, в лабораторії знаходяться сім робочих місць з ПК, кожне з яких устатковане монітором, вінчестером в системному блоці, трьома вентиляторами системи охолодження ПК та клавіатурою. Крім того поряд працює периферійна техніка. Таким чином у приміщенні мають місце шуми механічного і аеродинамічного походження, широкосмугові із аперіодичним підсиленням при роботі принтерів. Орієнтовні еквівалентні рівні звукового тиску джерел шуму, що діють на програміста на його робочому місці, представлені в таблиці 1.

Таблиця 1 – Рівні звукового тиску від різних джерел

Джерело шуму	Рівень шуму, дБА
Жорсткий диск	45
Вентилятор	45
Принтер матричний	55
Сканер	50

Допустимий еквівалентний рівень шуму для робочого місця програміста складає 50 дБА

3.2.5 Виробництво випромінювання

Допустимі значення параметрів неіонізуючих електромагнітних випромінювань від монітору комп'ютера представлені в таблиці 2.

Таблиця 2 – Допустимі значення параметрів неіонізуючих електромагнітних випромінювань

Найменування параметра	Допустимі значення
Напруженість електричної складової електромагнітного поля на відстані 50 см від поверхні відеомонітора	10 В/м
Напруженість магнітної складової електромагнітного поля на відстані 50 см від поверхні відеомонітора	0,3 А/м
Напруженість електростатичного поля не повинна перевищувати: для дорослих користувачів для дітей дошкільних установ і що вчать середнім спеціальних і вищих навчальних закладів	20кВ/м 15кВ/м

Нормованим параметром не використаного рентгенівського випромінювання виступає потужність експозиційної дози. На відстані 5 см від поверхні екрану монітору її рівень не повинен перевищувати 100 мкР/год. Максимальний рівень рентгенівського випромінювання на робочому місці програміста зазвичай не перевищує 20 мкР/год.

На відстані 5-10 см від екрана і корпусу монітора різні напруженості можуть досягати 140 В/м по електричній складовій, що значно перевищує допустимі значення.

3.2.6 Електробезпека. Статична електрика

Приміщення лабораторії за безпекою ураження електричним струмом можна віднести до I класу, тобто це приміщення без підвищеної небезпеки (сухе,

					КГ 05. 03 000. 00 ДП ПЗ	А пр.
Зл	А пр.	№ докум	Проває	Дата		30

без пилу, з нормальною температурою повітря, вольованими підлогами і малим числом заземлених приладів).

На робочому місці програміста з всього устаткування металевим є лише корпус системного блоку комп'ютера, але тут використовуються системні блоки, що відповідають стандартам фірми ІВМ, у яких крім робочої воляції передбачений елемент для заземлення і провід з жилом, що заземлює, для приєднання до джерела живлення.

На протязі роботи на корпусі комп'ютера накопичується статична електрика. На відстані 5-10 см від екрана напруженість електростатичного поля складає 60-280 кВ/м, тобто в 10 разів перевищує норму 20 кВ/м.

3.2.7 Організація робочого місця

Кожне робоче місце обладнане робочим столом площею 1,2 м², стільцем та персональним комп'ютером, що складається з монітора, системного блоку, клавіатури та миші.

Обладнання і організація робочого місця з ПК мають забезпечувати відповідність конструкцій всіх елементів робочого місця та їх взаємного розташування ергономічним вимогам з урахуванням характеру і особливостей трудової діяльності (ГОСТ 12.2.032-78, ГОСТ 22.269-76, ГОСТ 21.889-76). Конструкція робочого місця й взаємне розташування всіх його елементів (сидіння, органи керування, засоби відображення інформації) відповідають антропометричним, фізіологічним і психологічним вимогам, а також характеру роботи. Конструкція робочих меблів повинна забезпечувати можливість індивідуального регулювання відповідно росту працівника для підтримки зручної пози. Робочий стіл повинен бути пофарбований матовою фарбою. Дисплей розташований так, що його верхній край перебуває на рівні очей на відстані близько 70 см, що укладається в у притупленій рамці від 60 до 90 см. Частота мерехтіння екрана $f_{мер} = 100$ Гц, що відповідає умові $f_{мер} > 70$ Гц.

Робоче місце розташоване перпендикулярно віконним прорізам, це зроблено з тією метою, щоб виключити пряму й відбиту мерехтливість екрана від вікон і

					КГ 05. 03 000. 00 ДП ПЗ	А пр.
Зл	А пр.	№ докум.	Проект	Дата		51

приладів штучного освітлення, якими є лампи накаливання. Обладнання і організація робочого місця з ВДТ мають забезпечувати відповідність конструкції всіх елементів робочого місця та їх взаємного розташування, ергономічним вимогам з урахуванням характеру і особливостей трудової діяльності (ГОСТ 12.2.032.-78, ГОСТ 22.269.-76, ГОСТ 21.889-76).

3.4 Пожежна безпека

Ступінь вогнестійкості будинків приймається в залежності від їхнього призначення, категорії по вибухопожежній і пожежній небезпеці, по поверховості, площі поверху в межах пожежного відсіку згідно НАПБ Б.03.002-2007.

Будинок, у якому знаходиться лабораторія по пожежній небезпеці будівельних конструкцій відноситься до категорії К1 (малопожежобезпечні), оскільки тут присутні займисті (книжки, документи, меблі, оргтехніка і т.д.) і легкогорючі речовини (сейфи, різне устаткування і т.д.), що при взаємодії з вогнем можуть горіти без вибуху.

По конструктивних характеристиках будинок можна віднести до будинків з несучими і огорожувачими конструкціями із природних або штучних кам'яних матеріалів, бетону або залізобетону, де для перекриттів допускається використання дерев'яних конструкцій, захищених штукатуркою або важкогорючими листовими, а також плитними матеріалами. Отже, ступінь вогнестійкості будинку можна визначити як третю (III).

Причинами виникнення пожежі можуть бути:

- несправності електропроводки, розеток і вимикачів які можуть привести до короткого замикання або пробов ізоляції;
- використання ушкоджених (не справних) електроприладів;
- використання в приміщенні електронагрівальних приладів з відкритими нагрівальними елементами;
- виникнення пожежі внаслідок влучення блискавки в будинок;

					КГ 05. 03 000. 00 ДП ПЗ	Ар.
Зл	Ар.	№ розр.	Підпис	Дата		32

- загорання будівель внаслідок зовнішніх впливів;
- неакуратне поводження з вогнем і недотриманням ір пожежної безпеки.

У випадку виникнення пожежі необхідно відключити електроживлення, викликати по телефоні пожежну команду, евакуювати людей із приміщення відповідно до плану евакуації, приведеному на рисунку 1 і приступити до ліквідації пожежі вогнегасниками. При наявності невеликого вогнища полум'я, можна скористатися підручними засобами з метою припинення доступу по вітря до об'єкта загорання.

Для гасіння пожеж на робочому місці використовують вуглекислотні та порошкові вогнегасники.

- Вуглекислотні вогнегасники випускаються ж ручні (ВВК-5).
- Порошкові вогнегасники ВП-2, ВП-5, ВП-10 та інші.

Наявність переносних засобів пожежогасіння і вогнегасників, їхня кількість і зміст відповідає вимогам ГОСТ 124.009-75 і ISO 3941-77. У приміщенні виконуються всі вимоги по пожежній безпеці відповідно до вимог НАПБ А.0.001-95 "Правила пожежної безпеки в Україні".

У приміщенні також має ся план евакуації на випадок виникнення пожежі.

Забезпечення пожежної безпеки – це один із важливих напрямків щодо охорони життя та здоров'я людей, національного багатства і навколишнього середовища.

					КГ 05. 03 000. 00 ДП ПЗ	А р.
Зл	А р.	№ оапу	Прова	Дата		53

ВИСНОВКИ

Основов дипломного проекту була розробка чат-бота для взаємодії з криптовалютною біржею та криптовалютою Ethereum. У цій роботі було розглянуто і проаналізовано переваги та недоліки, а саме, ботів в месенджері Telegram, їх спосіб створення та взаємодія з ними, з криптовалютою та біржею. А саме реалізація алгоритму обробки даних користувача при взаємодії з гаманцем цифрової валюти.

Однією з головних особливостей даної програми є швидка та проста для рядового користувача взаємодія як з криптовалютною біржею CoinGecko так і з криптовалютою Ethereum, для цього були використані спеціалізовані API, такі як Etherscan API та CoinGecko API.

За допомогою вищезазначених засобів було реалізовано бота, що заощадить ваш час при роботі з вищезазначеними ресурсами, основними функціями якого є перегляд стану криптовалютного рахунку, транзакцій або перевірка ціни на найпопулярніші на даний момент криптовалюти.

У дипломному проекті було проведено тестування розробленої системи обробки транзакцій, балансу гаманця мережі криптовалюти Ethereum, курсу криптовалют біржі CoinGecko. Результати оцінювання чітко проаналізовано. Даний розділ надає інструкцію експлуатації для користувача програмного продукту, описує можливості та основні способи взаємодії з наданим інтерфейсом.

					КГ 05. 03 000. 00 ДП ПЗ	А р.
Зл	А р.	№ оц.ру	Підпис	Дата		СІ

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Чат-бот - что это такое, как работает, виды, преимущества и недостатки, как создать [Электронный ресурс] – Режим доступа: <http://kak-bog.ru/chat-bot-cto-eto-takoe-kak-rabotaet-vidy-preimushchestva>
2. Чат-боты: обзор и состояние технологий в отрасли m [Электронный ресурс] – Режим доступа: https://webguru.pro/blog/chatbots_development/
3. Чат-боты: обзор и состояние технологий в отрасли [Электронный ресурс] – Режим доступа: <http://hpx.net/archives/425>
4. Чат-бот для бизнеса [Электронный ресурс] – Режим доступа: <https://www.integrity.com.ua/activities/chatbots-for-business/>
5. Exploring chatbots for news [Электронный ресурс] – Режим доступа: <https://medium.com/ios-digital/exploring-chatbots-for-news-7f12d226c53d>
6. AI and Chatbots in Education [Электронный ресурс] – Режим доступа: <https://chatbotmagazine.com/ai-and-chatbots-in-education-whatdoes-the-futurehold-9772f5c13960>
7. Advantages of Chatbot Integration into Entertainment Industry [Электронный ресурс] – Режим доступа: <https://chatbotslife.com/advantages-of-chatbot-integration-into-entertainment-industry-f62bfd3c003>
8. Чат-бот : зачем создавать, кому использовать [Электронный ресурс] – Режим доступа: <https://blog.click.ru/chat-bot-na-sajte-zachem-sozdavat-komu-ispolzovat-i-kak-nastroit/>
9. 50 лучших чат-ботов для Telegram [Электронный ресурс] – Режим доступа: <https://ip.me/2016/04/50-popular-telegram-bots/python-telegram-bot> [Электронный ресурс] – Режим доступа: <https://github.com/python-telegram-bot/python-telegram-bot>
10. Луцц М. Программирование на Python, том II, 4-е издание. – Пер. сангл. – СПб.: Символ-Плюс, 2011. – 992 с.
11. How Assistant work & Introduction to Dialogflow [Электронный ресурс] – Режим доступа: <https://codeburst.io/how-assistant-work-introduction-to-dialogflow-319a72ba2db>

						КГ 05. 03 000. 00 ДП ПЗ	А р.
Зл	А р.	№ оц.ру	Проект	Дата			35

12. Introduction to the Google Assistant Library [Электронный ресурс] – Режим доступа: <https://developers.google.com/assistant/sdk/guides/library/python>
13. Messaging, simplified [Электронный ресурс] - Режим доступа: https://www.nexmo.com/products/sms/build?utm_source=google_search&utm_medium=paid&utm_campaign=EMEA_EURO_EN_SMS_General_B&utm_term
14. SpeechRecognition
15. Enhance Your Presence on Search [Электронный ресурс] - Режим доступа: <https://developers.google.com/search/>
16. Telegram APIs [Электронный ресурс]-Режим доступа: <https://core.telegram.org/>
17. pyTelegramBotAPI [Электронный ресурс]-Режим доступа: <https://github.com/eternnoir/pyTelegramBotAPI>
18. Пишем telegram-бота на python с помощью библиотеки telebot [Электронный ресурс] - Режим доступа: <https://habr.com/ru/post/448310>

					КГ 05. 03 000. 00 ДП ПЗ	А р.
Э.л	А р.	И з м е н е н и я	П р о в е с	Д а т а		36

ДОДАТОК А. Фрагмент листингу програмного модулю обробки транзакцій, балансу гаманця та хендлерів у Телеграм-боті на мові Python.

```
transactions.py
import logging

from aiogram import types
from aiogram.dispatcher import FSMContext
from aiogram.dispatcher.filters import Command, Text
from aiogram.types import CallbackQuery
from bitcoinlib.wallets import Wallet
from pycoingecko import CoinGeckoAPI
from requests import get
from matplotlib import pyplot as plt
import asyncio
import data
from states import BalanceCheck, TransCheck

cg = CoinGeckoAPI()
API_KEY = '91NEEET9HB38KMHAP46M8314WU39ZNYX15'
# https://api.etherscan.io/api
BASE_URL = "https://api.etherscan.io/api"
ETHER_VALUE = 10 ** 18

from keyboards.default import menu
from keyboards.default.menu import menu_eng, menu_ru
from keyboards.inline.callback_datas import choice_callback
from keyboards.inline.choice_buttons import choice
from loader import dp, bot
from datetime import datetime

to=""
from_addr=""
value=0
gas=0
time=""

@dp.message_handler(Text(equals=["Перевірити останню Ethereum транзакцію"]))
async def get_balance(message: types.Message):
    await message.answer(f'Введіть номер гаманця: ')
    await TransCheck.askForTrans1.set()

@dp.message_handler(Text(equals=["Check last Ethereum transaction"]))
async def get_balance(message: types.Message):
    await message.answer(f'Введіть номер гаманця: ')
    await TransCheck.askForTrans1.set()

@dp.message_handler(Text(equals=["Последняя Ethereum транзакция"]))
async def get_balance(message: types.Message):
    await message.answer(f'Введіть номер гаманця: ')

```

```
await TransCheck.askForTrans1.set()
```

```
@dp.message_handler(state=TransCheck.askForTrans1)
```

```
async def answer_q2(message: types.Message, state: FSMContext):
```

```
    answer = message.text
```

```
    await state.update_data(address=answer)
```

```
    def make_api_url(module, action, address, **kwargs):
```

```
        url = BASE_URL +
```

```
        f"?module={module}&action={action}&address={address}&apikey={API_KEY}"
```

```
        for key, value in kwargs.items():
```

```
            url += f"&{key}={value}"
```

```
    return url
```

```
    def get_transactions(address):
```

```
        transactions_url = make_api_url('account', 'txlist', address, startblock=0,  
endblock=99999999, page=1,
```

```
            offset=10000,
```

```
            sort='asc')
```

```
        response = get(transactions_url)
```

```
        data = response.json()['result']
```

```
        for tx in data:
```

```
            global to
```

```
            to = tx['to']
```

```
            from_addr = tx['from']
```

```
            value = int(tx['value']) / ETHER_VALUE
```

```
            gas = int(tx['gasUsed']) * int(tx['gasPrice']) / ETHER_VALUE
```

```
            time = datetime.fromtimestamp(int(tx['timeStamp']))
```

```
            asyncio.create_task(bot.send_message(message.from_user.id, f"-----\n To: {to}\n From: {from_addr}\n Value: {value}\n Gas Cost: {gas}\n Time: {time}", ))
```

```
        address = answer
```

```
        get_transactions(address)
```

```
    await state.finish()
```

```

menu.py
import logging

from aiogram import types
from aiogram.dispatcher import FSMContext
from aiogram.dispatcher.filters import Command, Text
from aiogram.types import CallbackQuery
from bitcoinlib.wallets import Wallet
from pycoingecko import CoinGeckoAPI
from requests import get
from matplotlib import pyplot as plt

import data
from states import BalanceCheck, TransCheck

cg = CoinGeckoAPI()
API_KEY = "91NEEET9HB38K00HAP46M831400U39ZNYX15"
# https://api.etherscan.io/api
BASE_URL = "https://api.etherscan.io/api"
ETHER_VALUE = 10 ** 18

from keyboards.default import menu
from keyboards.default.menu import menu_eng, menu_ru
from keyboards.inline.callback_datas import choice_callback
from keyboards.inline.choice_buttons import choice
from loader import dp, bot
from datetime import datetime

@dp.message_handler(Command("menu"))
async def show_menu(message: types.Message):
    await message.answer("Оберіть, що вас цікавить", reply_markup=menu)

@dp.message_handler(Text(equals=["Дізнатися баланс гаманця"]))
async def get_balance(message: types.Message):
    await message.answer(f"Введіть номер гаманця: ")

```

```
await BalanceCheck.askForBalance1_set()
```

```
@dp.message_handler(Text(equals=["Ethereum Wallet Balance"]))
```

```
async def get_balance(message: types.Message):
```

```
    await message.answer(f'Insert wallet number: ')
```

```
    await BalanceCheck.askForBalance1_set()
```

```
@dp.message_handler(Text(equals=["Узнать баланс Ethereum кошелька"]))
```

```
async def get_balance(message: types.Message):
```

```
    await message.answer(f'Введите номер кошелька: ')
```

```
    await BalanceCheck.askForBalance1_set()
```

```
@dp.message_handler(state=BalanceCheck.askForBalance1)
```

```
async def answer_q1(message: types.Message, state: FSMContext):
```

```
    answer = message.text
```

```
    await state.update_data(address=answer)
```

```
def make_api_url(module, action, address, **kwargs):
```

```
    url = BASE_URL +
```

```
f"?module={module}&action={action}&address={address}&apikey={API_KEY}"
```

```
    for key, value in kwargs.items():
```

```
        url += f"&{key}={value}"
```

```
    return url
```

```
def get_account_balance(address):
```

```
    get_balance_url = make_api_url('account', 'balance', address, tag='latest', x='2')
```

```
    response = get(get_balance_url)
```

```
    data = response.json()
```

```
    value = int(data['result']) / ETHER_VALUE
```

```
    return value
```

```
eth = get_account_balance(answer)
await message.answer(f'Balance: {eth} ETH')
await state.finish()
```

```
@dp.message_handler(Text(equals=["ЗМІНИТИ МОВУ 🇺🇸🇸🇰"]))
async def show_menu(message: types.Message):
    await message.answer("Choose what do you want to do", reply_markup=menu_eng)
```

```
@dp.message_handler(Text(equals=["Change Language 🇺🇸🇸🇰"]))
async def show_menu(message: types.Message):
    await message.answer("Выберите, что вас интересует", reply_markup=menu_ru)
```

```
@dp.message_handler(Text(equals=["ОМЕНИТЬ ЯЗЫК 🇺🇸🇸🇰"]))
async def show_menu(message: types.Message):
    await message.answer("Оберіть, що вас цікавить", reply_markup=menu)
```

```
@dp.message_handler(Text(equals=["Дізнатися курс криптовалюти"]))
async def get_course(message: types.Message):
    await message.answer(text="Курси на даний момент", reply_markup=choice)
```

```
@dp.message_handler(Text(equals=["Узнать курс криптовалюты"]))
async def get_course(message: types.Message):
    await message.answer(text="Курсы на данный момент.", reply_markup=choice)
```

```
@dp.message_handler(Text(equals=["Cryptocurrency rates"]))
async def get_course(message: types.Message):
    await message.answer(text="Current rates:", reply_markup=choice)
```

```
@dp.callback_query_handler(text='ethereum')
async def ethereum(call: CallbackQuery):
    price = cg.get_price(ids='ethereum', vs_currencies='usd')
    await call.answer(cache_time=60)
    await call.message.answer(f'Ethereum: 1 ETH = ${price[ethereum][usd]:.2f}')
```

```
@dp.callback_query_handler(text='bitcoin')
async def bitcoin(call: CallbackQuery):
    price = cg.get_price(ids='bitcoin', vs_currencies='usd')
    await call.answer(cache_time=60)
    await call.message.answer(f'Bitcoin: 1 BTC = ${price[bitcoin][usd]:.2f}')
```

```
@dp.callback_query_handler(text='litecoin')
async def bitcoin(call: CallbackQuery):
    price = cg.get_price(ids='litecoin', vs_currencies='usd')
    await call.answer(cache_time=60)
    await call.message.answer(f'Litecoin: 1 LTC = ${price[litecoin][usd]:.2f}')
```

```
@dp.callback_query_handler(text='tether')
async def bitcoin(call: CallbackQuery):
    price = cg.get_price(ids='tether', vs_currencies='usd')
    await call.answer(cache_time=60)
    await call.message.answer(f'Tether: 1 USDT = ${price[tether][usd]:.2f}')
```

```
@dp.callback_query_handler(text='binancecoin')
async def bitcoin(call: CallbackQuery):
    price = cg.get_price(ids='binancecoin', vs_currencies='usd')
    await call.answer(cache_time=60)
    await call.message.answer(f'BNB: 1 BNB = ${price[binancecoin][usd]:.2f}')
```

```
@dp.callback_query_handler(text='solana')
```

```
async def bitcoin(call: CallbackQuery):
    price = cg.get_price(ids='solana', vs_currencies='usd')
    await call.answer(cache_time=60)
    await call.message.answer(f'Solana: 1 SOL = ${price["solana"]["usd"]:.2f}')


```

```
@dp.callback_query_handler(text='cancel')
async def cancel(call: CallbackQuery):
    await call.message.edit_reply_markup()


```