ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ

Імені ІГОРЯ СІКОРСЬКОГО»

КАФЕДРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ КОМП’ЮТЕРНИХ СИСТЕМ

КУРСОВА РОБОТА

з дисципліни

Основи програмування

на тему: «Симуляція роботи криптогаманця»

Студента *1* курсу групи *КП-23*

Спеціальність *121 Інженерія програмного забезпечення*.

Сербіна Артема Сергійовича

Керівник старший викладач *Погорелов В.В.*

Національна оцінка

Кількість балів: Оцінка: ECTS

Члени комісії \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_

(підпис) (вчене звання, науковий ступінь, прізвище та ініціали)

(підпис) (вчене звання, науковий ступінь, прізвище та ініціали)

(підпис) (вчене звання, науковий ступінь, прізвище та ініціал

Київ – 2023 рік

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ФПМ НТУУ „КПІ”** | | | | | | |
| (назва вищого закладу освіти) | | | | | | |
| Кафедра | | ***Програмного забезпечення комп’ютерних систем*** | | | | |
| Дисципліна | | ***Основи програмування*** | | | | |
| Галузь знань | | ***12 Інформаційні технології*** | | | | |
| Курс | ***перший*** | | Група | ***КП-23*** | Семестр | ***другий*** |

# ЗАВДАННЯ

**на курсовий проект(роботу) студента**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Сербін Артем Сергійович | | | | | |
| (прізвище, ім’я, по батькові) | | | | | |
| 1. Тема проекту(роботи) | Симуляція роботи криптогаманця | | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
| 2. Строк здачі студентом закінченого проекту(роботи) | | | ***XX.05.20XX р.*** | | |
| 3. Вихідні дані до проекту(роботи) | |  | | | |
|  | | | | | |
|  | | | | | |
|  | | | | | |
| 4. Зміст розрахунково-пояснювальної записки (перелік питань, які підлягають розробці) | | | | |  |
| ***1. Постановка задачі.*** | | | | | |
| ***2. Метод розв’язку задачі*** | | | | | |
| ***3. Загальна блок-схема алгоритму та опис алгоритму*** | | | | | |
| ***4. Опис програмного продукту.*** | | | | | |
| ***5. Результати роботи.*** | | | | | |
| ***6. Висновки.*** | | | | | |
| ***7. Список використаної літератури.*** | | | | | |
|  | | | | | |
| 5. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень) | | | |  | |
| ***1. Загальна блок-схема алгоритму.*** | | | | | |
| ***2. Ілюстрації роботи програми.*** | | | | | |
| 6. Дата видачі завдання |  | | | | |

**ЗМІСТ**

Вступ………………………………………………………………………………4

Розділ 1. Постановка задачі……………………………………………………...5

Підрозділ 1.1: Огляд існуючих підходів та аналіз алгоритмів………….5

Підрозділ 1.2: Обґрунтування вибраного варіанту реалізації та постановка задачі…………………………………………………………..5

Розділ 2. Розробка програмного продукту………………………………………6

Підрозділ 2.1: Метод розв'язку задачі…………………………………….6

Підрозділ 2.2: Алгоритм розв'язку задачі…………………………………7

Розділ 3. Опис розробленого програмного продукту…………………………..8

Підрозділ 3.1: Опис головних структур та змінних програми…………..8

Підрозділ 3.2: Опис головних функцій програми………………………..9

Підрозділ 3.3: Опис інтерфейсу…………………………………………..10

Підрозділ 3.4: Результат роботи програмного продукту………………..13

Висновок………………………………………………………………………….15

**ВСТУП**

Симуляція криптогаманця є цікавою та актуальною задачею в галузі програмування. Вона вимагає розробки алгоритмів, які дозволять моделювати криптографічні процеси та взаємодію з криптографічними гаманцями.

Метою даної курсової роботи є створення, за допомогою Visual Studio Windows Forms та мови програмування C#, графічного додатку, що представляє базову функціональність криптогаманця для легкого старту вивчення тематики криптовалюти загалом.

**РОЗДІЛ 1**

**Постановка задачі**

**Підрозділ 1.1: Огляд існуючих підходів та аналіз алгоритмів**

З кожним днем криптографія та криптовалюта стають все більш популярними. Криптогаманець є інструментом, що дозволяє користувачам зберігати, керувати та використовувати криптовалюту. На сьогодні вже створено сотні криптогаманців, але тих, що відповідають усім умовам шифрування, алгоритмам криптографії та повної правильності в мережах блокчейнів, налічується близько пари десятків. Для симуляції криптогаманця необхідно розробити на основі існуючих криптогаманців алгоритми, які забезпечать створення, збереження, передачу та отримання криптовалюти на віртуальний гаманець.

**Підрозділ 1.2: Обґрунтування вибраного варіанту реалізації та постановка задачі**

У даній курсовій роботі обрано варіант реалізації базового функціоналу криптогаманця для початківців. Мною планується розробити програму, яка дозволить користувачам створювати гаманці для зберігання криптовалюти, поповнювати їх, переказувати кошти між гаманцями, переглядати стан свого балансу і т. д.

Постановка задачі має наступний зміст:

* Розробка інтерфейсу користувача, який дозволить зареєструватися та створити новий унікальний гаманець.
* Забезпечення можливості переказу коштів між різними гаманцями за допомогою введення адреси отримувача та суми переказу.
* Відображення балансу гаманця та історії транзакцій.
* Все це відбувається у демонстраційному режимі, що дозволяє використовувати вигадані кошти та транзакції без реального доступу до криптовалюти.

Для успішної реалізації симуляції криптогаманця необхідні наступні програмні та технічні засоби:

* Середовище програмування: Visual Studio, версія 2019 або новіша.
* Мова програмування: C#.
* Інструментарій для розробки графічного інтерфейсу: Visual Studio Windows Forms.
* Комп'ютер з операційною системою Windows або віртуальна машина з встановленою ОС Windows для виконання та тестування програми.

**РОЗДІЛ 2**

**Розробка програмного продукту**

**Підрозділ 2.1: Метод розв'язку задачі**

Метод розв'язку задачі буде базуватися на створенні графічного інтерфейсу користувача та розробці відповідних функцій для керування коштами та транзакціями.

Перш за все, необхідно створити графічний інтерфейс для користувача з використанням Visual Studio Windows Forms. Інтерфейс має бути зручним та інтуїтивно зрозумілим для початківців, щоб вони могли легко керувати своїми коштами та виконувати транзакції. Він може включати елементи, такі як кнопки для створення нового гаманця, перегляду балансу, виконання переказів та інші функції, які необхідні для роботи з криптовалютою.

Крім графічного інтерфейсу, необхідно розробити функції, які забезпечують роботу з криптовалютою. Ці функції можуть включати: створення нового унікального гаманця, зберігання пароля від гаманця тільки на стороні користувача, забезпечуючи його безпеку, отримання балансу гаманця та відображення його на інтерфейсі, виконання переказу коштів з одного гаманця на інший за допомогою публічного адреса отримувача, зберігання та відображення історії транзакцій користувача. Для зберігання гаманців та транзакцій можуть бути використані колекції, такі як Dictionary.

**Підрозділ 2.2: Алгоритм розв'язку задачі**

Загальний алгоритм роботи програми:

1. Ініціалізація програми: створення графічного інтерфейсу користувача для реєстрації або входу;
2. - При реєстрації:

* Запис даних користувача до колекції;
* Генерація нового гаманця;
* Пункт 3;
* При вході:
* Перевірка даних на їх наявність у колекції;
* Завантаження гаманця;
* Пункт 3;

1. Обробка запитів користувача, таких як переказ коштів або перегляд історії транзакцій;
2. Оновлення та збереження балансу користувача;
3. Оновлення та збереження історії транзакцій;
4. Завершення роботи програми.

Структурна блок-схема узагальненого алгоритму роботи програми:

Start

Program initialization

Registration

Login

Data recording

Data verification

Generation of a new wallet

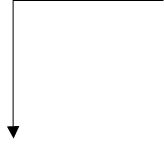
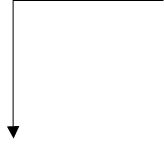
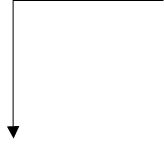
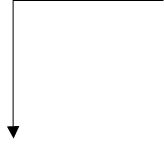
Loading the wallet

Balance update

Processing user requests

Transaction history update

End



**РОЗДІЛ 3**

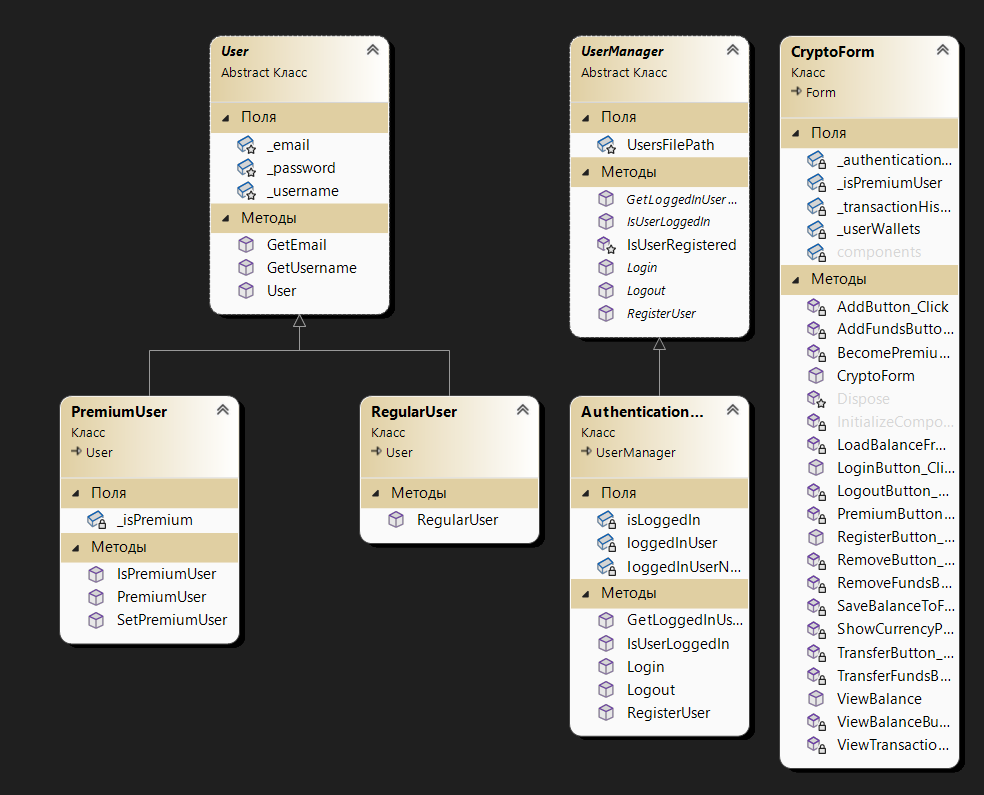
**Опис розробленого програмного продукту**

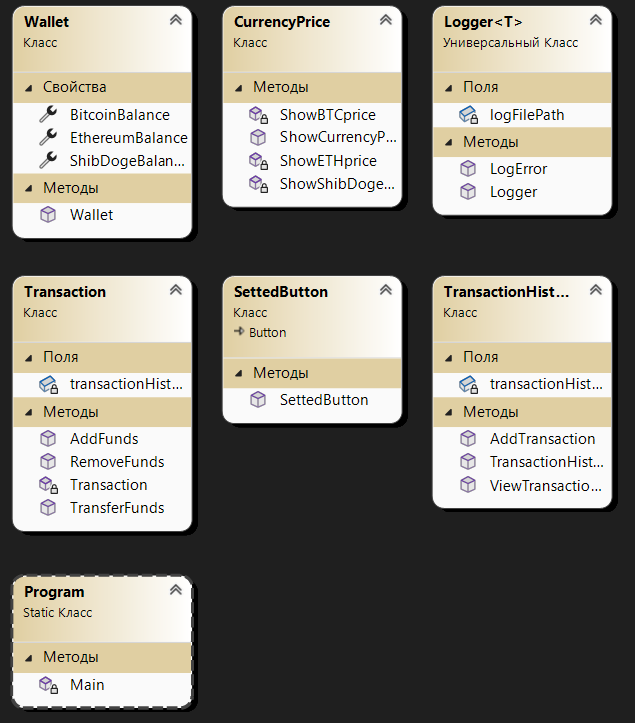
**Підрозділ 3.1: Опис головних структур та змінних програми**

Головними структурами даної програми є:

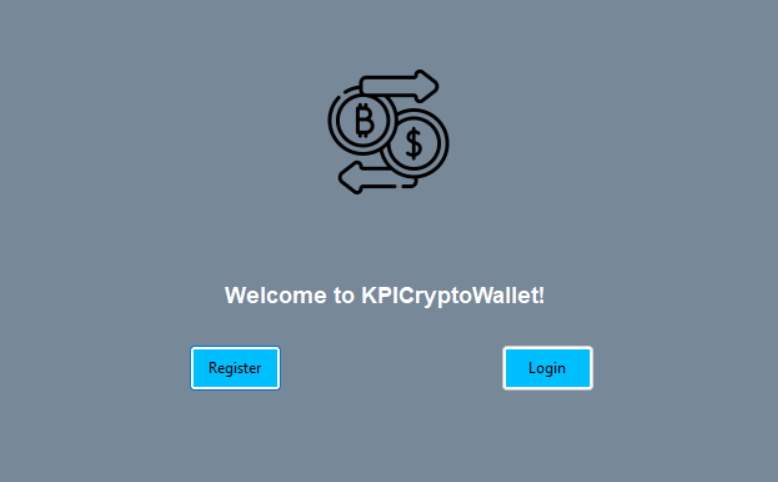
* CryptoForm – клас, що відповідає за створення графічного інтерфейсу користувача;
* AuthenticationManager – клас, що відповідає за реєстрацію користувача, запис його даних до колекції для подальших входів у систему, перевірку даних при спробі входу;
* Transaction – клас, що відповідає за функціональність всіх видів транзакцій, що зроблені у програмі.

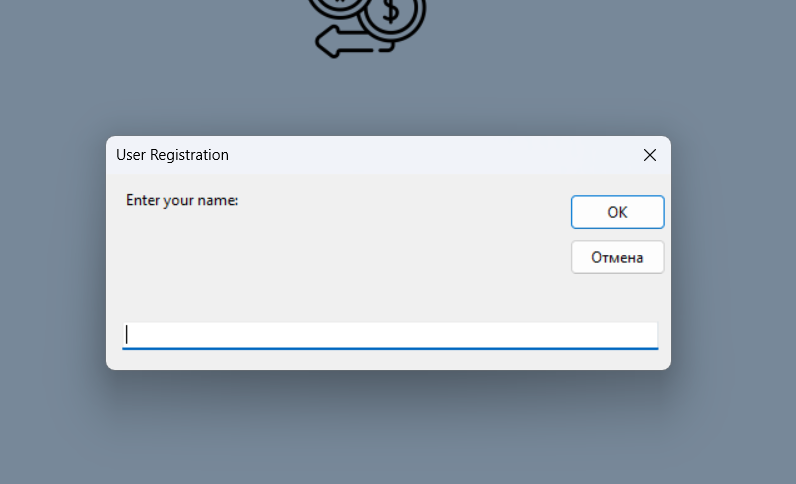
**Підрозділ 3.2: Опис головних функцій програми**

****

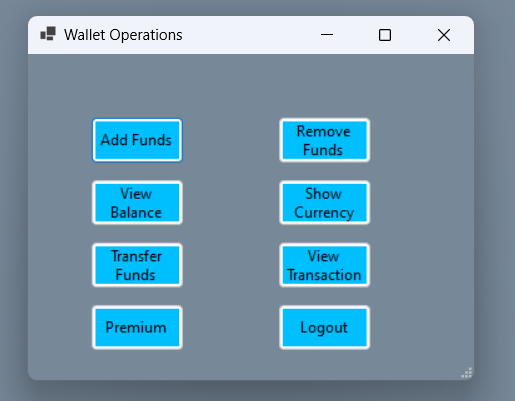
****Схема класів

* RegisterUser() – перевіряє, чи існує користувач з тими ж даними, які були введені у полі реєстрації. Якщо ні, ці дані додаються до колекції для входу у систему;
* LoadBalanceFromFile() – завантажує гаманець користувача, який здійснив вхід;
* AddFunds() – поповнює баланс обраної користувачем криптовалюти на вказану ним суму;
* SaveBalanceToFile() – зберігає баланс користувача після виконання функцій AddFunds(), RemoveFunds(), TransferFunds().
* ViewBalance() – виводить інформацію про баланс користувача;

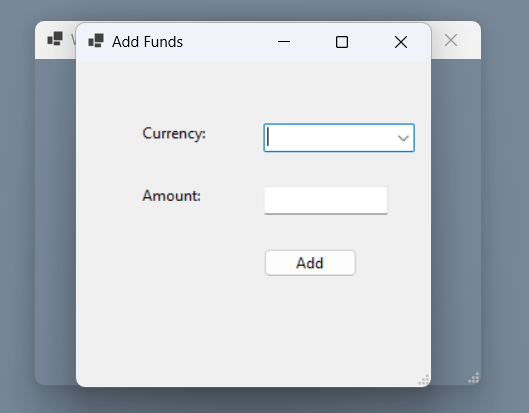
**Підрозділ 3.3: Опис інтерфейсу**

Інтерфейс для реєстрації або входу

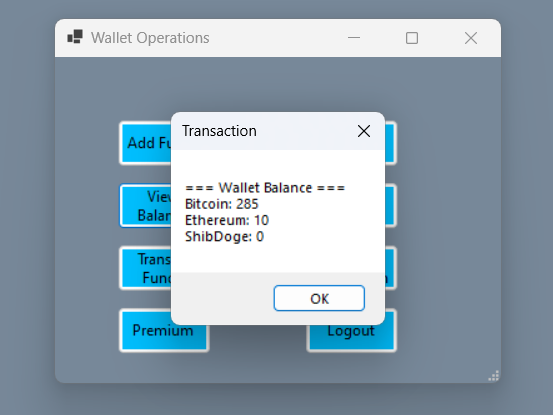
Екран реєстрації



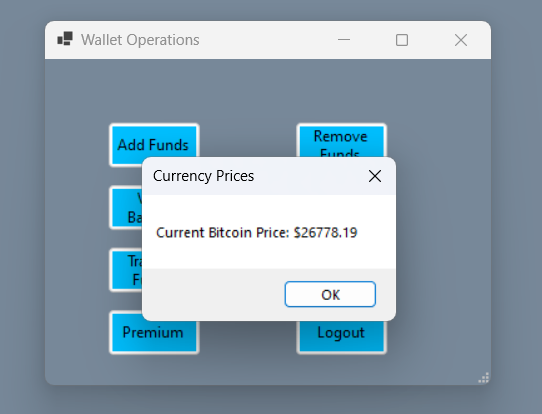
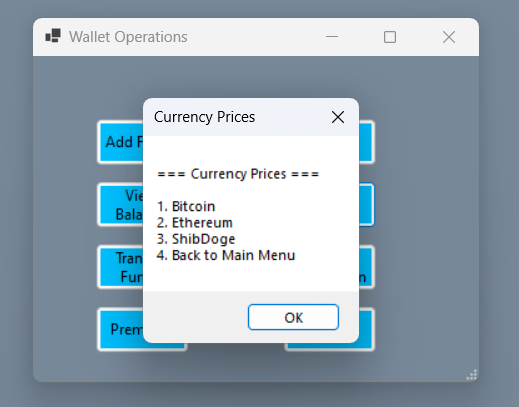
Головний екран для здійснення операцій



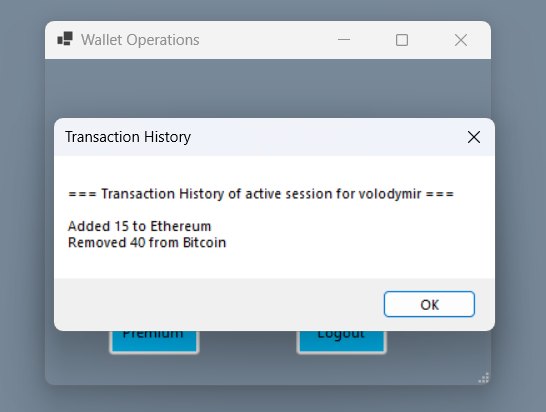
Екран поповнення рахунку

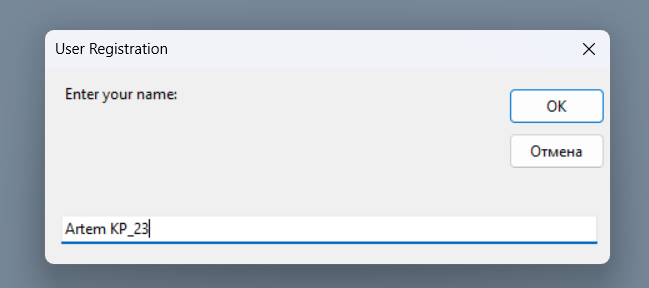


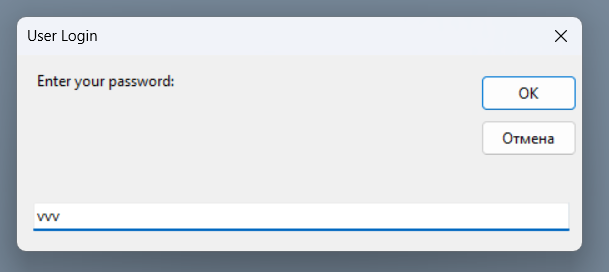
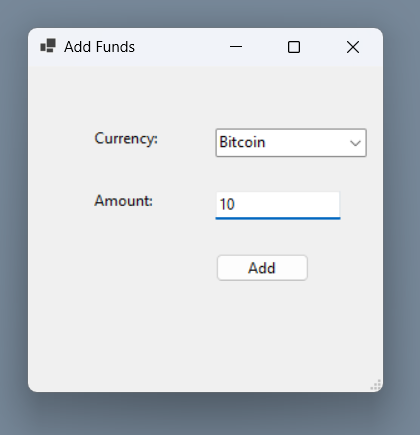
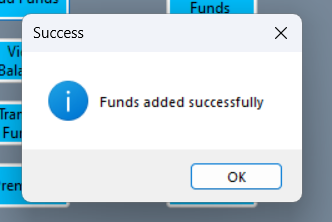
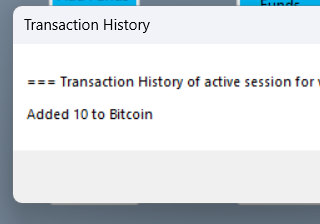
Екран перегляду балансу



Екран перегляду поточних цін на конкретну криптовалюту

 Екран перегляду історії транзакцій

**** **Підрозділ 3.4: Результат роботи програмного продукту**

1. Реєструємось;
2. Входимо до свого аккаунту;
3. Натискаємо у головному меню на кнопку «Add Funds», у цьому меню обираємо потрібну нам криптовалюту для поповнення, вводимо суму та натискаємо «Add»;
4. Повідомлення, що операція пройшла успішно. Натискаємо «ОК»;
5. Як можемо бачити, операція дійсно пройшла.

**ВИСНОВОК**

У ході розробки курсової роботи було створено програмний продукт - криптогаманець, який надає базовий функціонал для керування демо-криптовалютними активами. Робота мала на меті розробку імітаційного середовища для початківців, щоб допомогти їм ознайомитися з основами криптографічних гаманців та вивчити процес здійснення транзакцій у блокчейні.

У першому розділі була поставлена задача та проведений огляд існуючих підходів та алгоритмів, пов'язаних з криптографічними гаманцями. Було розглянуто різні методи зберігання активів та виконання транзакцій. Аналізуючи існуючі рішення, було визначено необхідний функціонал та обрано підхід для подальшої реалізації.

У другому розділі була розроблена програма для управління криптовалютними активами. Метод розв'язку задачі передбачав використання мови програмування C# та інструментарію Visual Studio Windows Forms для розробки графічного інтерфейсу. Було реалізовано генерацію гаманця, відображення балансу, виконання транзакцій та збереження історії. Алгоритм роботи програми був описаний у вигляді блок-схеми, яка показує послідовність дій та взаємозв'язки між функціональними блоками.

В результаті роботи над курсовою роботою було досягнуто наступних результатів:

Розроблено програмний продукт - криптогаманець, який забезпечує базовий функціонал для керування криптовалютними активами.

Використовуючи мову програмування C# та інструментарій Visual Studio Windows Forms, було реалізовано графічний інтерфейс, що спрощує взаємодію з програмою та надає зручний спосіб керування активами.

Криптогаманець забезпечує можливість генерації гаманця, перегляду балансу, виконання транзакцій та збереження історії транзакцій.

У роботі було виявлено певні недоліки, зокрема, обмеження функціональності до базових операцій та відсутність реальної реалізації блокчейну та шифрування. Однак, ці обмеження були зумовлені спрощеними вимогами до програмного продукту та його орієнтацією на початківців.

Отже, розроблений програмний продукт відповідає поставленим завданням та меті проєкту, надаючи базовий функціонал для керування криптовалютними активами. Дальші покращення та розширення можуть зробити криптогаманець ще більш потужним і зручним для користувачів.

Пропозиції для подальшого вдосконалення:

* Додати функціонал шифрування даних для забезпечення безпеки користувачів та захисту конфіденційної інформації.
* Розширити можливості криптогаманця, підтримуючи різні типи криптовалют та протоколи.
* Вдосконалити інтерфейс користувача, забезпечивши більш інтуїтивний та зручний досвід взаємодії з програмою.
* Дослідити можливість інтеграції з реальним блокчейн-мережами, щоб надати реалістичну симуляцію транзакцій та взаємодію з глобальною криптовалютною інфраструктурою.

Загалом, криптогаманець є початковим кроком у вивченні та розумінні принципів криптовалют та криптографії. Із подальшим розвитком та вдосконаленням, він може стати потужним інструментом для керування та обміну криптовалютними активами.