Зміст

[User Story 1: Ініціалізація та запуск програми 2](#_Toc196759950)

[User Story 2: Моніторинг використання CPU 3](#_Toc196759951)

[User Story 3: Відображення використання CPU у GUI 4](#_Toc196759952)

[User Story 4: Моніторинг використання RAM 5](#_Toc196759953)

[User Story 5: Відображення використання RAM у GUI 6](#_Toc196759954)

[User Story 6: Моніторинг мережевої активності 7](#_Toc196759955)

[User Story 7: Відображення мережевої активності у GUI 8](#_Toc196759956)

[User Story 8: Управління процесами (завершення процесу) 9](#_Toc196759957)

[User Story 9: Логування даних та сповіщень 10](#_Toc196759958)

[User Story 10: Експорт даних у файл 11](#_Toc196759959)

# User Story 1: Ініціалізація та запуск програми

Опис:  
Як користувач, я хочу, щоб програма ініціалізувала основні компоненти та запускала графічний інтерфейс, щоб я міг почати моніторинг системи.

Критерії приймання:

* Програма автоматично налаштовує всі необхідні компоненти для моніторингу системи під час запуску.
* Відкривається головне вікно з вкладками для перегляду даних про CPU, RAM, мережу та процеси.
* Починається фоновий збір даних про стан системи.
* Програма коректно завершує роботу при закритті вікна, не залишаючи активних процесів.
* Перевірка: Програма запускається, вікно відображається, закриття програми відбувається без помилок.

Оцінка часу:

* Написання коду: 3 години
* Відлагодження: 2 години
* Тестування: 1 година
* Оновлення документації: 1 година
* Загальний час: 7 годин

# User Story 2: Моніторинг використання CPU

Опис:  
Як користувач, я хочу, щоб програма моніторила використання CPU в реальному часі, щоб я міг бачити, як завантажений мій процесор.

Критерії приймання:

* Програма кожні 3 секунди збирає дані про загальне використання CPU та використання кожного ядра.
* Зібрані дані зберігаються для подальшого відображення та аналізу.
* Перевірка: Дані оновлюються кожні 3 секунди і містять значення в діапазоні 0–100%.

Оцінка часу:

* Написання коду: 2 години
* Відлагодження: 1 година
* Тестування: 1 година
* Оновлення документації: 1 година
* Загальний час: 5 годин

# User Story 3: Відображення використання CPU у GUI

Опис:  
Як користувач, я хочу бачити використання CPU у графічному інтерфейсі, щоб візуально оцінити завантаження процесора.

Критерії приймання:

* У графічному інтерфейсі відображається текстова інформація про загальне використання CPU (наприклад, "Загальне використання CPU: 45.2%").
* Графік показує використання CPU для кожного ядра за останні 60 секунд.
* Якщо використання CPU перевищує заданий поріг, користувач отримує сповіщення.
* Перевірка: Текст і графік оновлюються кожні 3 секунди, сповіщення з’являється при перевищенні порогу.

Оцінка часу:

* Написання коду: 3 години
* Відлагодження: 2 години
* Тестування: 1 година
* Оновлення документації: 1 година
* Загальний час: 7 годин

# User Story 4: Моніторинг використання RAM

Опис:  
Як користувач, я хочу, щоб програма моніторила використання оперативної пам’яті, щоб я знав, скільки пам’яті використовується.

Критерії приймання:

* Програма кожні 3 секунди збирає дані про відсоток використаної оперативної пам’яті.
* Зібрані дані зберігаються для подальшого відображення та аналізу.
* Перевірка: Дані оновлюються кожні 3 секунди і містять значення в діапазоні 0–100%.

Оцінка часу:

* Написання коду: 2 години WIRE
* Відлагодження: 1 година
* Тестування: 1 година
* Оновлення документації: 1 година
* Загальний час: 5 годин

# User Story 5: Відображення використання RAM у GUI

Опис:  
Як користувач, я хочу бачити використання RAM у графічному інтерфейсі, щоб оцінити доступну пам’ять.

Критерії приймання:

* У графічному інтерфейсі відображається текстова інформація про відсоток використаної RAM та обсяг (наприклад, "Використання RAM: 65.3% (6.2/8.0 GB)").
* Графік показує використання RAM за останні 60 секунд.
* Якщо використання RAM перевищує заданий поріг, користувач отримує сповіщення.
* Перевірка: Текст і графік оновлюються кожні 3 секунди, сповіщення з’являється при перевищенні порогу.

Оцінка часу:

* Написання коду: 3 години
* Відлагодження: 2 години
* Тестування: 1 година
* Оновлення документації: 1 година
* Загальний час: 7 годин

# User Story 6: Моніторинг мережевої активності

Опис:  
Як користувач, я хочу, щоб програма моніторила мережеву активність, щоб я знав, скільки даних передається.

Критерії приймання:

* Програма кожні 3 секунди збирає дані про швидкість завантаження та відправлення даних через мережу.
* Зібрані дані зберігаються для подальшого відображення та аналізу.
* Перевірка: Дані оновлюються кожні 3 секунди і містять коректні значення швидкості (у Mbps).

Оцінка часу:

* Написання коду: 2 години
* Відлагодження: 1 година
* Тестування: 1 година
* Оновлення документації: 1 година
* Загальний час: 5 годин

# User Story 7: Відображення мережевої активності у GUI

Опис:  
Як користувач, я хочу бачити мережеву активність у графічному інтерфейсі, щоб оцінити обсяг передачі даних.

Критерії приймання:

* У графічному інтерфейсі відображається текстова інформація про швидкість завантаження та відправлення (наприклад, "Завантаження: 5.2 Mbps | Відправлення: 2.1 Mbps").
* Графік показує швидкість завантаження та відправлення за останні 60 секунд.
* Якщо швидкість передачі даних перевищує заданий поріг, користувач отримує сповіщення.
* Перевірка: Текст і графік оновлюються кожні 3 секунди, сповіщення з’являється при перевищенні порогу.

Оцінка часу:

* Написання коду: 3 години
* Відлагодження: 2 години
* Тестування: 1 година
* Оновлення документації: 1 година
* Загальний час: 7 годин

# User Story 8: Управління процесами (завершення процесу)

Опис:  
Як користувач, я хочу мати можливість завершувати процеси через інтерфейс, щоб зменшити навантаження на систему.

Критерії приймання:

* У графічному інтерфейсі відображається таблиця активних процесів із інформацією про їхній ідентифікатор (PID), назву, використання пам’яті та CPU.
* Користувач може вибрати процес і завершити його після підтвердження.
* Після завершення процесу таблиця оновлюється, відображаючи актуальний стан.
* Якщо завершення процесу неможливе (наприклад, через брак прав), користувач отримує повідомлення про помилку.
* Перевірка: Процес завершується, таблиця оновлюється, помилки обробляються коректно.

Оцінка часу:

* Написання коду: 3 години
* Відлагодження: 2 години
* Тестування: 1 година
* Оновлення документації: 1 година
* Загальний час: 7 годин

# User Story 9: Логування даних та сповіщень

Опис:  
Як розробник, я хочу, щоб програма логувала дані та сповіщення, щоб я міг діагностувати проблеми.

Критерії приймання:

* Програма записує діагностичні повідомлення та сповіщення у файл журналу.
* Сповіщення зберігаються в окремому списку для перегляду.
* Перевірка: Файл журналу містить коректні записи, список сповіщень оновлюється.

Оцінка часу:

* Написання коду: 2 години
* Відлагодження: 1 година
* Тестування: 1 година
* Оновлення документації: 1 година
* Загальний час: 5 годин

# User Story 10: Експорт даних у файл

Опис:  
Як користувач, я хочу експортувати зібрані дані у файл, щоб зберегти звіт про стан системи.

Критерії приймання:

* Користувач може натиснути кнопку для створення текстового файлу зі звітом про використання CPU, RAM, мережі та сповіщення.
* Файл має унікальну назву з позначкою часу (наприклад, system\_report\_20250506\_1200.txt).
* Перевірка: Файл створюється і містить коректні дані.

Оцінка часу:

* Написання коду: 2 години
* Відлагодження: 1 година
* Тестування: 1 година
* Оновлення документації: 1 година
* Загальний час: 5 годин