

Таблиця 1 Загальний опис класів та їх призначення.

Клас	Призначення
<b>BackupManager</b>	Відповідає за основні операції резервного копіювання: створення та збереження резервних копій, відновлення даних, перевірку цілісності та архівацію файлів.
<b>Scheduler</b>	Клас для планування та автоматизації резервного копіювання. Керує розкладом запуску, часом та періодичністю виконання задач копіювання.
<b>Cleaner</b>	Відповідає за очищення дискового простору шляхом автоматичного видалення застарілих копій відповідно до заданих умов або кількості збережених резервів.
<b>UIManager</b>	Клас, що забезпечує взаємодію користувача з програмою. Містить графічний інтерфейс, обробку запитів, введення налаштувань та перегляд звітів/журналів.
<b>LogManager</b>	Веде журнал дій, фіксує помилки та результати операцій. Також відповідає за створення текстових звітів після кожної сесії резервного копіювання.
<b>Encryptor</b>	Клас для шифрування та розшифрування файлів перед їх збереженням у резервних копіях або при відновленні з них. Він дозволяє користувачам захистити свої дані за допомогою пароля.

1. Діаграма класів: Діаграма класів показує структуру програми, визначаючи основні класи, такі як BackupManager, Sheduler, Cleaner, UIManeger, LogManager та Encryptor, а також їх методи та взаємозв'язки. Ця діаграма дозволяє зрозуміти, як компоненти взаємодіють між собою і які функції вони виконують у межах програми.

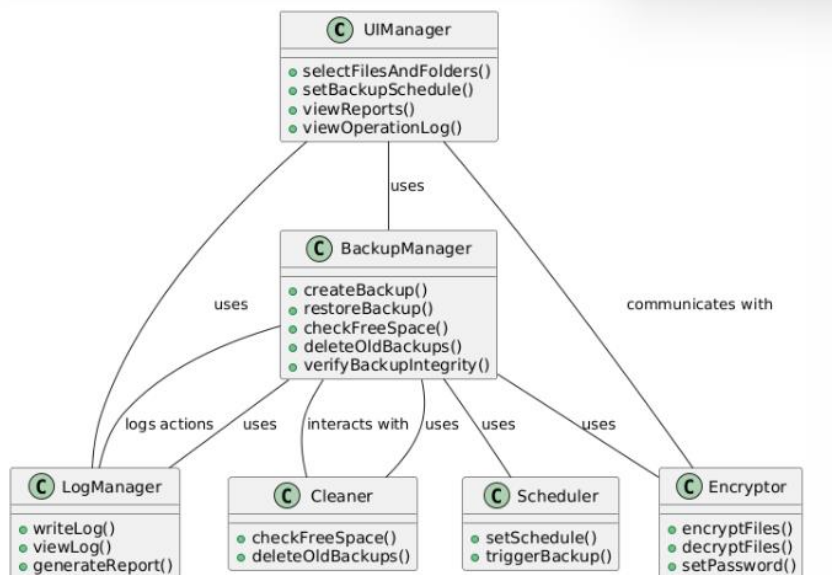


Рис. 1. Діаграма класів.

## 2. Структурна схема.

Структурна схема програми відображає загальний потік роботи додатку, показуючи, як користувачі вибирають файли та папки для резервного копіювання, налаштовують автоматичне копіювання та взаємодіють з іншими функціями, такими як шифрування та відновлення.

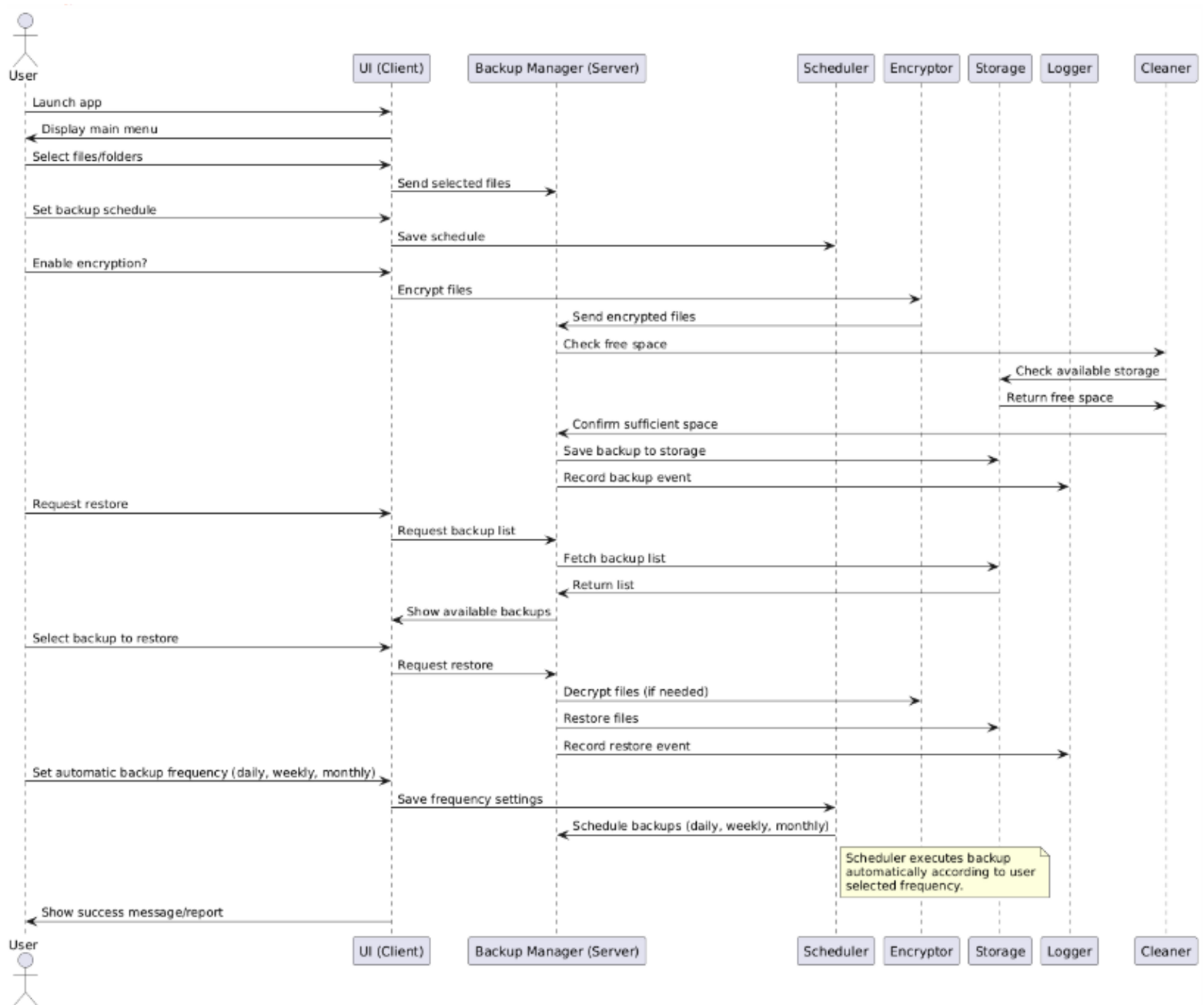


Рис. 2. Структурна схема програми.

- **User:** Користувач, який взаємодіє з програмою через **UIManager**.
- **UIManager:** Контролює весь процес взаємодії з користувачем (вибір файлів, налаштування розкладу, перегляд звітів).
- **BackupManager:** Відповідає за резервне копіювання, відновлення та взаємодію з іншими компонентами (зберігання резервних копій, шифрування, перевірка місця на диску).

- **Encryptor**: Використовується для шифрування та розшифрування файлів.
- **Cleaner**: Перевіряє вільне місце на диску і видаляє старі копії, коли потрібно.
- **LogManager**: Веде журнал операцій та створює звіти.
- **Scheduler**: Автоматизує процес резервного копіювання за розкладом.
- **BackupStorage**: Місце збереження резервних копій.

### Взаємодія:

- Користувач через **UIManager** ініціює процес резервного копіювання або відновлення.
- **BackupManager** забезпечує основний процес резервного копіювання, взаємодіє з **BackupStorage**, шифрує файли через **Encryptor** і перевіряє вільне місце через **Cleaner**.
- **LogManager** записує усі дії в журнал і надає звіти користувачеві.
- **Scheduler** автоматизує процес копіювання за розкладом.

### 3. Діаграма послідовності для резервного копіювання.

Діаграма послідовності описує покрокову взаємодію об'єктів програми в конкретних сценаріях, таких як резервне копіювання, шифрування файлів або відновлення з резервної копії, і вказує, коли і які функції викликаються в реальному часі.

Як користувач взаємодіє з системою резервного копіювання:

1. **Запуск програми** – користувач відкриває застосунок, інтерфейс відображає головне меню.
2. **Створення резервної копії**:
  - Користувач обирає файли та (за потреби) шифрує їх за допомогою пароля.
  - Система перевіряє наявність вільного місця.
  - Якщо місця достатньо – копія зберігається та фіксується в журналі.
  - Якщо ні – виводиться попередження.
3. **Налаштування автоматичного копіювання** – користувач обирає періодичність (день, тиждень, місяць), планувальник зберігає розклад.
4. **Відновлення файлів**:
  - Користувач переглядає список резервних копій.
  - Обирає потрібну, вводить пароль для розшифрування (якщо потрібно).
  - Система виконує відновлення та реєструє дію.

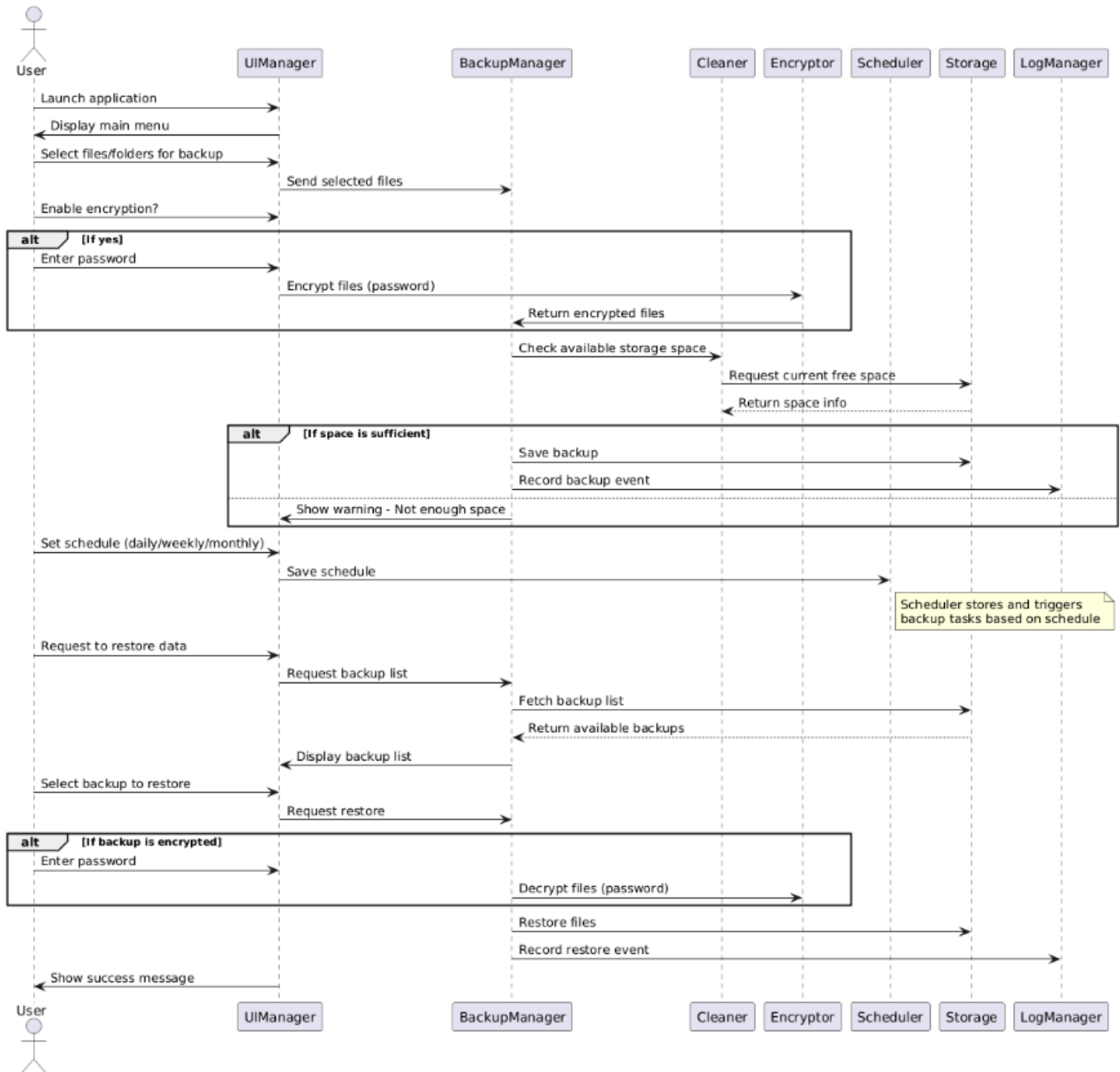


Рис. 3. Діаграма послідовності для резервного копіювання

#### 4. Діаграма варіантів застосування.

Діаграма варіантів застосування ілюструє основні сценарії використання програми користувачем, такі як вибір файлів для резервного копіювання, налаштування автоматичного копіювання, шифрування та відновлення файлів, що допомагає чітко визначити вимоги до програмного забезпечення.

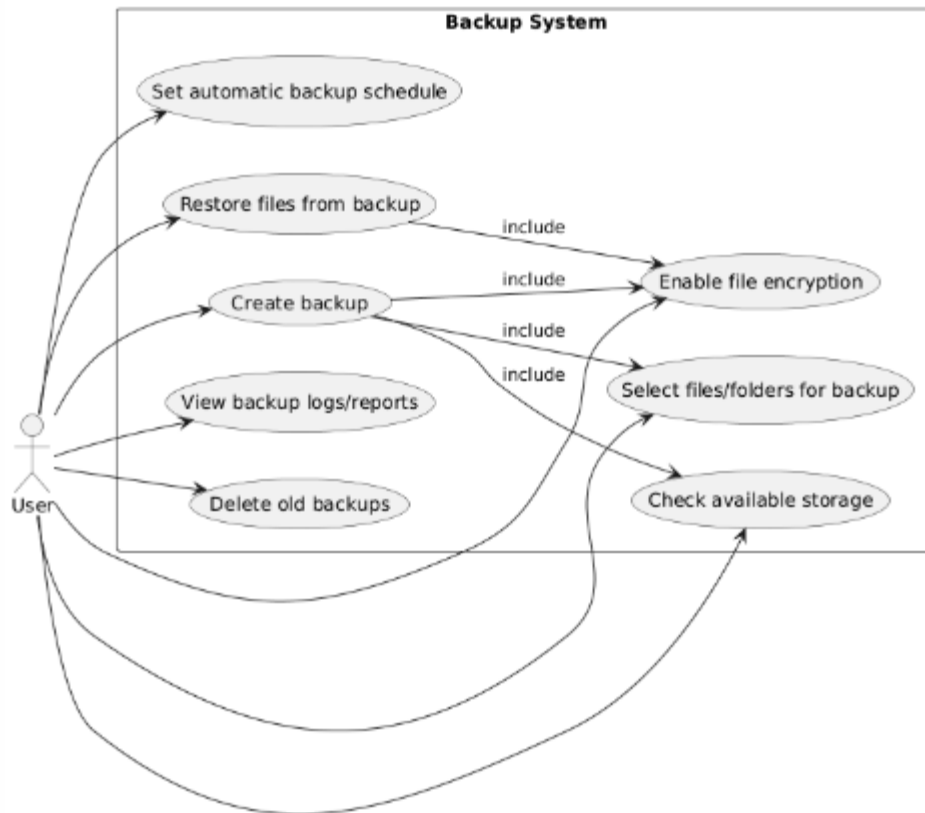


Рис. 4. Діаграма варіантів застосування.

## 5. Діаграма компонентів програми.

Діаграма компонентів показує фізичну структуру системи, зокрема як окремі модулі та компоненти програми взаємодіють між собою. Вона відображає основні частини програми, такі як модулі для резервного копіювання, шифрування, відновлення файлів та планування задач, а також їх взаємозв'язки. Ця діаграма дозволяє зрозуміти, як компоненти програми взаємодіють з зовнішніми системами (наприклад, файловою системою чи архіватором), а також які ресурси використовуються для виконання основних операцій. Вона корисна для визначення меж компонентів та аналізу можливих проблем при інтеграції різних частин програми.

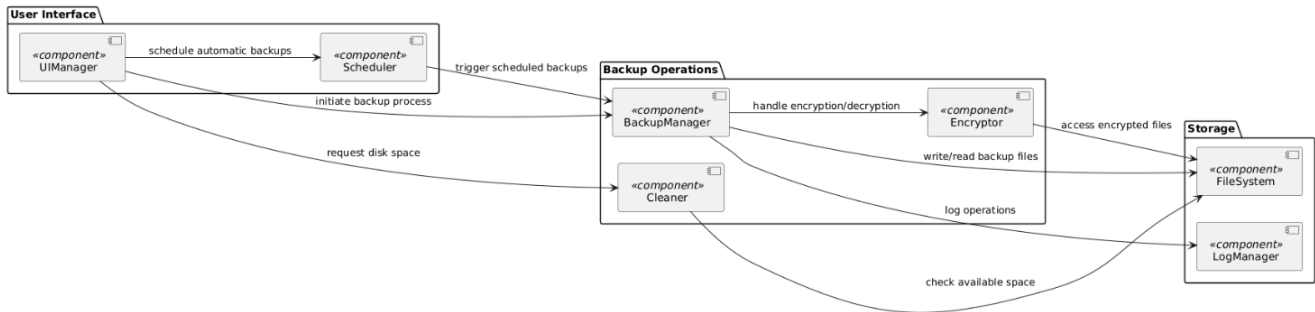


Рис. 5. Діаграма компонентів програми.

### 1. Сховище (Storage)

- **FileSystem:** Цей компонент керує фактичним зберіганням та отриманням резервних копій файлів. Він відповідає за запис і зчитування даних резервного копіювання, включаючи зашифровані файли.
- **LogManager:** Цей компонент веде журнал всіх дій, що виконуються системою, таких як резервне копіювання, відновлення та помилки, що забезпечує правильний моніторинг та відслідковування діяльності системи.

### 2. Операції резервного копіювання (Backup Operations)

- **BackupManager:** Це основний компонент процесу резервного копіювання, який координує всі операції, пов'язані зі створенням і відновленням резервних копій. Він взаємодіє з компонентами, такими як FileSystem, Encryptor та LogManager.
- **Cleaner:** Відповідає за керування доступним місцем на диску. Перевіряє, чи є достатньо вільного місця для нових резервних копій, і в разі нестачі місця може вивести попередження.
- **Encryptor:** Цей компонент відповідає за шифрування та розшифрування файлів. Він гарантує, що дані резервного копіювання будуть збережені в захищеному вигляді шляхом їх шифрування перед збереженням та розшифрування після відновлення.

### 3. Користувацький інтерфейс (User Interface)

- **UIManager:** Забезпечує взаємодію користувача з системою. Дає змогу користувачам вибирати файли для резервного копіювання, налаштовувати розклад, конфігурувати шифрування та керувати резервними копіями. Взаємодіє з усіма компонентами на серверній стороні для виконання дій користувача.

- **Scheduler**: Цей компонент відповідає за автоматизацію процесу резервного копіювання відповідно до розкладу, визначеного користувачем (щодня, щотижня, щомісяця). Він запускає **BackupManager** в зазначений час.

### Взаємодія між компонентами

- **UIManager** взаємодіє з **BackupManager**, **Cleaner** та **Scheduler** для запуску і планування операцій резервного копіювання.
- **BackupManager** взаємодіє з **Encryptor** для шифрування/розшифрування файлів, з **FileSystem** для обробки даних резервного копіювання та з **LogManager** для реєстрації всіх операцій.
- **Cleaner** перевіряє доступне місце на диску, запитуючи **FileSystem**, щоб переконатися, що для нових резервних копій є достатньо місця.
- **Scheduler** автоматично запускає резервне копіювання, викликаючи **BackupManager** відповідно до заданого розкладу.
- **Encryptor** шифрує та розшифровує файли під час процесу резервного копіювання та відновлення.