**D. Y. Patil College of College of Engineering and Technology, Kolhapur**

**Department of Computer Science & Engineering**

**Class: SY-A Subject: AOOC**

**Experiment no: 15**

**Group No: G8 Mini Project**

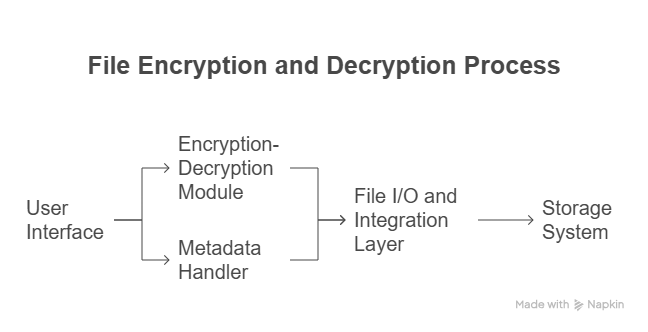
**Title of Mini-Project:** File encryption and decryption tool

**Problem Statement:**

In digital environments, managing and securing files effectively is a growing concern—especially when dealing with confidential, academic, or business-related data. Additionally, users often face challenges understanding the context or purpose of stored files without opening them. There is a need for a system that not only encrypts files for security but also allows users to associate metadata (such as descriptions or tags) with them for better file understanding, categorization, and retrieval.

**Introduction:**

With the exponential growth in digital data, file security and organization have become vital. Traditional file systems rely heavily on file names and locations, which are often insufficient to convey the file's purpose. Over time, users may forget what a file contains or why it was saved. This mini-project introduces a Java-based tool that enables users to encrypt and decrypt files securely using AES encryption. Additionally, the system supports metadata tagging, allowing users to add descriptions to files for easier identification. This approach is particularly useful in academic, research, and business settings where both security and clarity are essential.

**System Architecture:**

**Module description or working of system:**

**1. File Upload & Input Module:**

* Allows users to upload files via a clean and simple Java Swing GUI.
* Prompts users to enter a description or tags for the uploaded file before encryption.
* Ensures that both the file and its metadata are collected at the time of upload.

**2. Metadata Handling Module:**

* Accepts the user-provided description or tags and binds it to the file.
* Stores metadata in a structured .meta file (JSON or plain text) associated with the encrypted file.
* In future upgrades, this module could support embedding metadata directly into file headers for supported formats.

**3. Encryption-Decryption Module:**

* Uses AES (Advanced Encryption Standard) to securely encrypt uploaded files.
* Generates and stores a unique .key file for every encryption operation.
* During decryption, uses the provided key to restore the original file safely.
* Verifies that decryption was successful and shows appropriate status messages.

**4. File Management & Listing Module:**

* Maintains a record of encrypted files along with their metadata.
* Can be extended to display a list of files with options to sort, search, or filter using the descriptions or tags.
* Ensures files, keys, and metadata are grouped logically.

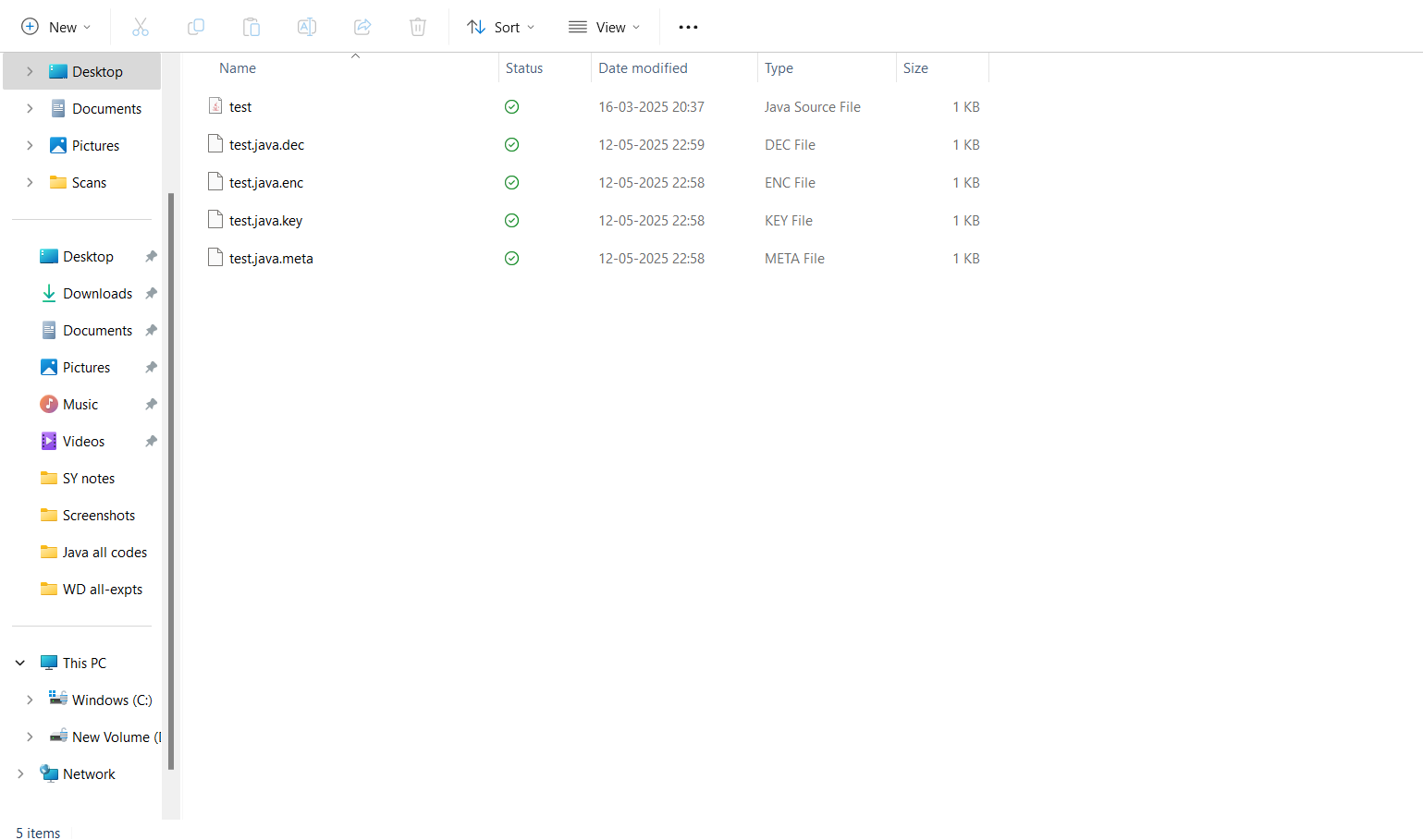
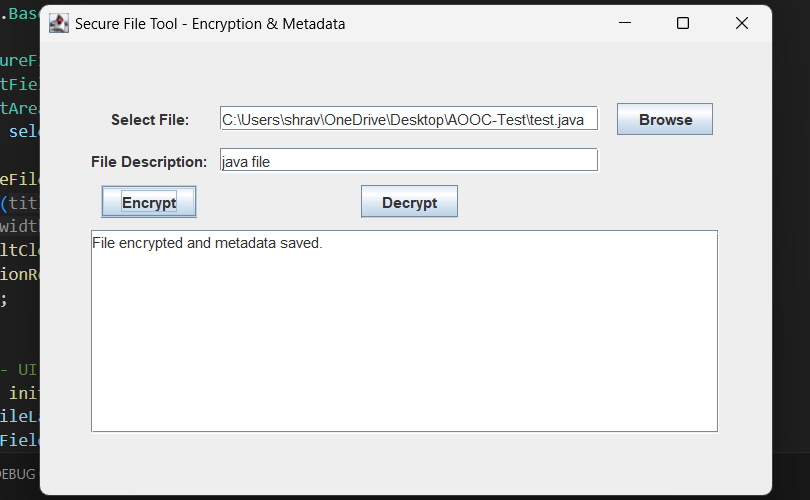
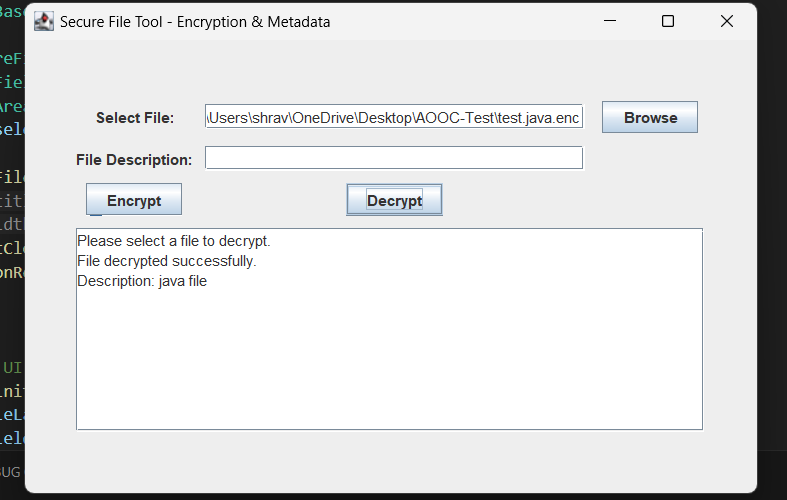
**5. Storage Module:**

* Stores encrypted files (.enc), key files (.key), and metadata (.meta) on local disk in an organized format.
* Ensures data integrity by matching metadata and encrypted files via naming conventions or unique IDs.

**6. Retrieval Module:**

* Allows users to retrieve files based on keywords, tags, or description text.
* Supports decrypting the file once the correct key and description are matched.

**Screenshots:**

****

**Group Members:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Unique id** | **Roll No** | **Name of Student** | **Sign** |
| EN23232092 | 26 | Kshitija Anil Jiwaje |  |
| EN23177271 | 32 | Shravani Mahesh Mali |  |
| EN23232043 | 46 | Rasika Bapuso Bendugale |  |
| EN23146494 | 50 | Sandhya Sardar Chougale |  |