Info-Hide: Secure File Hiding System

1. Overview: Info-Hide is a application oversees a hidden file safely. The Java uses OOP and DAO qualifies this system to be based on secure file storage and data storage. Encrypted proprietary text file is stored inside the database itself thereby maintaining confidentiality and access control.

2. Features:

• File Encryption and Hiding: Encrypts text files securely and stores them in MySQL database.
• User Authentication and Authorization: Email verification, OTP, login/signup processes secured.
• Database Management: Resistance against any forms of storage reliability and integrity, because it deletes all references made in the database to the encrypted file.
• Data Access Object (DAO): Disconnected database operations enhance business logic into a highly scalable and maintainable structure.

3. Technologies Used:

• Programming Language: Java
• Database- MySQL
• Security Features: Encrypt, OTP based authentication, secure login/signup
• Architecture- The DAO pattern for interacting with the database in a structured manner

4. Solution Architecture:

• Frontend: Java Swing for user interface (optional CLI operations for flexibility)
• Back End: Core Java with application of OOP principles
• CRUD operations: on MySQL follow the DAO pattern for structured interaction
• Security: AES encryption for files, hashed passwords, and OTP based authentication

5. Implementation Details:

• User Authentication Module:
o User registration with email verification
o Login with encrypted password storage
o OTP-based authentication for added security
• File Encryption & Storage Module:
o Encrypt fixed text file with AES encryption
o File is stored securely in MySQL.
o Corresponding retrieval with decryption on successful authentication-provided
• Implementation DAO Layer:
o UserDAO: Does all operations concerning user records in database
o FileDAO: Handles encrypted file storage and retrieval
o OTPDAO: Handles account OTP verification

6. Database schema:

• User Table: Includes all user credentials and hashed passwords, together with details on email verification.
• File Table: Stores all encrypted file data, user associations, and your files.
• OTP table: Includes records on OTP for authentication.

7. Security Measures:

- AES encryption for file security
- Hashed passwords (e.g. BCrypt)
- OTP-based authentication for secure access

8. Future Enhancements:

- Overhead support for other file formats
- Multi-level access with role-based permissions
- Cloud storage integration

9. Deployment Instructions:

- Install MySQL and develop the database schema.
- Set the Java enviroment to put in required dependencies.
- Run the Java application and interact through CLI or GUI.