

UNIVERSITY OF MINES AND TECHNOLOGY
TARKWA

FACULTY OF COMPUTING AND MATHEMATICAL SCIENCES
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

PROJECT REPORT ENTITLED
PERSONAL DOCUMENT VAULT

BY

ARTHUR PRINCE TAKYI

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE
AWARD OF THE DEGREE OF BACHELOR OF SCIENCE IN COMPUTER
SCIENCE AND ENGINEERING

PROJECT SUPERVISOR

.....

DR SYLVESTER AKPAH

TARKWA GHANA

SEPTEMBER, 2024

DECLARATION

I, Arthur Prince Takyi, declare that this project work is my own work. It is being submitted for the degree of Bachelor of Science in Computer Science and Engineering in the University of Mines and Technology (UMaT), Tarkwa. It has not been submitted for any degree or examination in any other University..

Signature of Student: _____

_____ day of _____ (year) _____

ABSTRACT

As personal information becomes available on computers, the possibility of data breaches, unauthorized access, and inefficiency in workflow for both individuals and organizations increases. This has almost become a widespread complaint due to the seeming lack of security and oversight for these documents. This project addresses that issue by establishing the Personal Document Vault (PDV), a secure, user-friendly system with its primary focus on maximum security, easy organization, and reliable control. The vault will house confidential documents using several security measures, including Advanced Encryption Standard (AES) 256-bit encryption, Rivest-Shamir-Adleman (RSA) and Two-Factor Authentication (2FA). The system was developed using these advanced encryption techniques, and testing showed a significant reduction in unauthorized access and data breaches while improving document organization and retrieval. To further enhance security and functionality, it is recommended to integrate biometric authentication and AI-driven categorization, with regular updates to encryption protocols to maintain security standards. The PDV has successfully provided users with a secure, efficient solution for managing personal documents. With its robust security features, it ensures the prevention of unauthorized access and data leakage. The system categorizes and tags documents, supports advanced search capabilities, and integrates version control and an audit trail for comprehensive document tracking. Its intuitive interface enhances productivity and workflow efficiency, restoring users' confidence in managing their documents digitally while safeguarding their sensitive information.

DEDICATION

I dedicate this project to my loving and supportive family, especially my father, Mr John Takyi Arthur, and my mother, Mrs Rose Arthur. Your unwavering support and encouragement have been invaluable throughout this journey. This accomplishment is as much yours as it is mine. I am truly grateful for everything you've done to help me reach this milestone. To the entire family, this is for you.

ACKNOWLEDGEMENTS

I attribute my achievements to the Almighty God, who has guided and supported me throughout my journey. A heartfelt acknowledgement goes to Dr Sylvester Akpah, my supervisor, whose unwavering dedication and brilliant insights have been instrumental in making this project a success. I am also very grateful to my friends and family, whose valuable insights and financial and moral support have contributed to the successful completion of this project. A special thanks go to Japhet Kuntu Blankson and Paul Nii Botchwey for their contribution in the completion of this project. I extend my appreciation to all the lecturers at UMaT who have nurtured my academic growth and provided support during my time on campus.

TABLE OF CONTENTS

DECLARATION	ii
ABSTRACT	iii
DEDICATION	iv
ACKNOWLEDGEMENTS	v
TABLE OF CONTENTS	vi
LIST OF FIGURES	viii
LIST OF TABLES	ix
LIST OF ABBREVIATIONS	x
CHAPTER 1 INTRODUCTION	1
1.1 Statement of Problem	1
1.2 Aim and Objective	2
1.3 Tools and Facilities Used	2
1.4 Methods Used	3
1.5 Scope of Work	4
1.6 Work Organization	4
CHAPTER 2 LITERATURE REVIEW	5
2.1 Overview of Literature Review	5
2.2 Digital Document	5
2.3 Security Concerns In Document Management	5
2.4 Securing Digital Document	5
2.4.1 Advanced Encryption Standard 256 Bits	6
2.4.2 Rivest-Adleman-Shamir	6
2.4.3 Comparative Analysis	7
2.4.4 Why Combine AES And RSA	8
2.4.5 Advantages And Disadvantages Of AES 256 Bits And RSA	9

2.4.6	Potential Vulnerabilities and Mitigations	9
2.4.7	Steganography	10
2.4.8	Implementing a Secure Personal Document Vault	12
2.5	Related Works	13
2.6	Proposed Solution	17
CHAPTER 3	SYSTEM DESIGN AND METHODOLOGY	18
3.1	Overview of System Design and Methodology	18
3.2	Software Development Life Cycle	18
3.3	Agile Software Development	18
3.3.1	Plan Stage	19
3.3.2	Design Stage	20
3.4	Development Stage	26
3.4.1	Execute Stage	29
3.4.2	Maintain Stage	30
CHAPTER 4	SYSTEM TESTING AND RESULTS	31
4.1	Overview of System Testing and Results	31
4.2	Personal Document Vault User Interface	31
4.2.1	Authentication	32
4.2.2	Vault	35
4.3	Error Pages	39
CHAPTER 5	CONCLUSION AND RECOMMENDATION	41
5.1	Conclusion	41
5.2	Recommendation and Future Works	41
REFERENCES		43

LIST OF FIGURES

Figure	Title	Page
3.1	Stages of Agile Software Development (Source: Ewin, 2024)	19
3.2	Three Tier Architecture (Source: McCall, 2024)	21
3.3	PDV Use Case Diagram	23
3.4	Document Encryption Process	25
3.5	Document Decryption Process	26
4.1	PDV Landing Page	31
4.2	Login Page	32
4.3	Forgot Password Page	33
4.4	Register Page	33
4.5	Account Pairing Page	34
4.6	Verify Page	34
4.7	Vault Statistics	35
4.8	Activity Tracking Graphs	36
4.9	Activity Tracking Table	36
4.10	User Info	36
4.11	Folders	37
4.12	Inside a Folder	37
4.13	Extract Key	37
4.14	OCR and Editor Tool	38
4.15	MFA Settings	38
4.16	Trash	38
4.17	Page Not Found (404)	39
4.18	Internal Server Error (500)	40
4.19	Forbidden (403)	40
4.20	Bad Request (400)	40

LIST OF TABLES

Table	Title	Page
2.1	AES and RSA Comparison	7
2.2	Advantages and Disadvantages of AES and RSA	9
2.3	Summary of Related Papers	13
3.1	Frontend Technologies Used and Their Applications	27
3.2	Key Libraries Used in the Backend	28

LIST OF ABBREVIATIONS

PDV	Personal Document Vault
TOTP	Time-based One-Time Password
AES	Advanced Encryption Standard
RSA	Rivest-Shamir-Adleman
MFA	Multi-Factor Authentication
2FA	Two-Factor Authentication
HTTPS	HyperText Transfer Protocol Secure
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
SFTP	Secure File Transfer Protocol
PDF	Portable Document Format
PDM	Product Data Management
DES	Data Encryption Standard
SSL	Secure Sockets Layer
TLS	Transport Layer Security
DMS	Document Management System
LSB	Least Significant Bit
DCT	Discrete Cosine Transform
AWS	Amazon Web Services
3DES	Triple Data Encryption Standard
ECC	Elliptic Curve Cryptography
RC6	Rivest Cipher 6
RC2	Rivest Cipher 2
ACLs	Access Control Lists
RAD	Rapid Application Development
EDMS	Electronic Document Management System
UAT	User Acceptance Testing
SDLC	Software Development Life Cycle
UML	Unified Modeling Language
ACID	Atomicity, Consistency, Isolation, Durability

CHAPTER 1

INTRODUCTION

1.1 Statement of Problem

In the contemporary digital landscape, a vast array of documents is generated in one's life due to day-to-day activities across numerous sectors, including healthcare, finance, commerce, education, and personal endeavors (Smith, 2022). These documents often contain sensitive and confidential information crucial for personal, professional, and organizational purposes. However, the prevailing issue lies in the inadequate protection and management of these documents, leading to significant vulnerabilities and inefficiencies in workflow (Johnson *et al.*, 2023).

The prevalent practice of storing sensitive documents in an unsecured manner on smart devices, including laptops, mobile phones, and tablets, or in hardcopy format leaves them susceptible to a myriad of threats (Brown, 2021). These threats range from sophisticated phishing attacks targeting digital documents to unauthorized access by individuals in physical proximity, jeopardizing the confidentiality, integrity, and availability of the information contained within.

Furthermore, the disorganized and scattered nature of these documents intensify the problem. They are often buried within cluttered folders alongside various file types, impeding effective retrieval and increasing the likelihood of accidental deletion or misplacement (Lee and Wang, 2024). Consequently, individuals and organizations experience inefficiencies in their daily workflows, spending valuable time and resources navigating through the chaos to locate critical documents when needed.

To address these challenges comprehensively, there is an urgent need for the development of a robust Personal Document Vault (PDV). This solution prioritizes the implementation of stringent security measures to safeguard documents during transmission and storage. Additionally, it offers intuitive features for categorization, organization, and management, enabling users to efficiently track, access, and utilize their documents while maintaining data integrity and confidentiality (Davis, 2023).

By providing a secure, centralized repository for personal documents, the proposed Personal Document Vault will empower individuals and organizations to mitigate risks associated with unauthorized access and data breaches. Moreover, it will streamline document management processes, enhancing productivity and workflow efficiency in various domains. Thus, the development and adoption of such a solution are imperative to address the pressing challenges posed by the proliferation of sensitive documents in today's digital age.

1.2 Aim and Objective

The project aims to create a secure, efficient solution for managing personal documents, addressing challenges like inadequate protection and workflow in the digital age, while ensuring confidentiality and protection.

The above aim can be achieved through the following objectives:

- i. To implement stringent and multi-layered security measures to secure the documents;
- ii. To provide a centralized repository to house document;
- iii. To enable intuitive document organization and management;
- iv. To provide a user-friendly interface to manage document.

1.3 Tools and Facilities Used

Tools and facilities used are:

- i. Django Web Framework;
- ii. Bootstrap;
- iii. Google Authenticator Mobile App;
- iv. Python Programming Language;
- v. Cryptography Library;
- vi. Sublime Text Editor;
- vii. Windows Command Prompt;
- viii. PostgreSQL;

- ix. PyOTP;
- x. Git and Github;
- xi. Tesseract OCR;
- xii. Laptop; and
- xiii. Internet.

1.4 Methods Used

The methods used are:

- i. Implemented strong encryption method with Advanced Encryption Standard (AES) 256 bits combined with Rivest-Shamir-Adleman (RSA) to secure the documents;
- ii. Managed encryption keys with a QRCode-based image steganography using the Least Significant Bit (LSB) technique to thwart attackers;
- iii. Implemented high authentication methods to verify and validate users using normal authentication and Two Factor Authentication (2FA);
- iv. Employed secured communication protocols such as HTTPS or SFTP when transferring documents over networks to prevent eavesdropping and man-in-the-middle attacks;
- v. Implemented features for categorizing and tagging documents based on metadata such as content type, date, and relevance;
- vi. Implemented a search feature for easy access of document;
- vii. Incorporated version control mechanisms to track document revisions and changes over time; and
- viii. Logged all document-related activities, including access attempts, modifications, and deletions, to establish an audit trail.

1.5 Scope of Work

The scope of my work covers the essential components required to develop a secure, efficient, and user-friendly Personal Document Vault. It includes tasks related to security implementation, document organization and management, centralized repository, user experience and productivity, development methodology, deployment and maintenance, and documentation and training.

1.6 Work Organization

The project is structured as follows: Chapter one is comprised of the problem statement, project objectives, tools and facilities used, methods used, project scope and organisation of work. Chapter two covers a review of relevant literature pertaining to Personal Document Vault. Chapter three elaborates the methodologies used in the software design process and implementation. Chapter four outlines the implementation and results of Personal Document Vault. Chapter five concludes the project and provides the necessary recommendations.

CHAPTER 2

LITERATURE REVIEW

2.1 Overview of Literature Review

This chapter provides a comprehensive exploration of existing research concerning document management systems. It emphasizes an often-overlooked aspect such as management and security of digital documents.

2.2 Digital Document

These days, digital documents are essential to information management in the modern world. Electronic files containing formatted text, images, or other data types can be classified using various automated systems. These systems aim to analyze and categorize different file formats, including office documents, scanned files, and multimedia content (Eken *et al.*, 2019). Spreadsheets, presentations, text documents, and Portable Document Format (PDF) has become a widely adopted format for electronic document exchange, offering robust visual presentation across platforms (Seggern *et al.*, 2019). The transition from physical to digital documents has been a gradual process since the introduction of word processing in the 1960s and they offer advantages such as lower production costs and instant accessibility, they have not completely replaced physical documents (Dimou and Syropoulos, 2021).

2.3 Security Concerns In Document Management

Modern document storage is digital, which raises a number of security issues. Unauthorized access, data breaches, ransomware and malware attacks, and insider threats are common dangers to digital documents (Rao and Nayak, 2014). These dangers have the power to jeopardize the availability, confidentiality, and integrity of saved documents, which could result in serious harm to one's reputation, finances, or personal life.

2.4 Securing Digital Document

A key component in maintaining the privacy of stored documents is encryption. Encryption ensures that even if unauthorized parties obtain access to the stored data, they cannot read

its contents without the decryption key by transforming plain text into cipher text (Stallings, 2017). In secure document management, symmetric encryption which uses a single secret key and asymmetric encryption, which uses a public and private key pair, both play crucial roles.

By utilizing steganography and hybrid cryptography, the proposed Personal Document Vault seeks to address issues arising from various document management systems and offer a more reliable security solution.

2.4.1 Advanced Encryption Standard 256 Bits

The Advanced Encryption Standard (AES) was developed to replace the aging Data Encryption Standard (DES) in the late 1990s (Nađ, 2014). AES was created by Belgian cryptographers Vincent Rijmen and Joan Daemen, and adopted by the U.S. National Institute of Standards and Technology (NIST) in 2001 after a five-year standardization process (Selent, 2010). Unlike DES, which used a 56-bit key for 64-bit data blocks, AES employs 128-bit block sizes with key sizes of 128, 192, or 256 bits (Nađ, 2014). AES has since become the global standard for symmetric encryption. AES operates on 128-bit blocks of data, using key lengths of 128, 192, or 256 bits. The 256-bit key variant, known as AES-256, offers the highest level of security. The AES algorithm employs a series of substitution and permutation operations, organized into rounds, with the number of rounds depending on the key size (Stallings, 2017).

The U.S. government believes that AES-256 offers a security level high enough to safeguard secret data up to the Top Secret level (Committee on National Security Systems, 2015). Its extensive use in a variety of sectors, including healthcare and banking, attests to its dependability and effectiveness. Its superiority in file encryption has been demonstrated. AES has been performance-optimized for both hardware and software implementations. According to Gueron, 2010, specific AES instructions are frequently found in modern CPUs, which greatly accelerates encryption and decryption times.

2.4.2 Rivest-Adleman-Shamir

The RSA cryptosystem, introduced in 1977, is one of the earliest practical public-key systems, utilizing two mathematically designed keys for encryption and decryption (Berlin

and Dhenakaran, 2017). Its foundation is the mathematical issue of factoring big numbers, which is challenging for conventional computers to process computationally. Two keys are produced by the RSA algorithm: a private key for decryption and a public key for encryption. The difficulty of factoring the product of two large prime integers is the foundation for RSA's security (Katz and Lindell, 2014). The main advantage of RSA is that it can distribute keys securely without requiring a pre-shared secret. Its application in protocols such as SSL/TLS attests to its usefulness in establishing secure connections over unreliable channels (Rescorla, 2018).

2.4.3 Comparative Analysis

Comparing AES and RSA, they both have their strength and weakness but when made hybrid, there are several benefits. It resolves the key distribution issue in contrast to pure symmetric encryption.

Table 2.1 AES and RSA Comparison

Algorithm/ Parameter	AES	RSA
Type	Symmetric Block Cipher	Asymmetric Block Cipher
Structure	Feistel Network	Exponentiation Congruence
Key Size (bits)	128, 192, 256	1024
Rounds	10, 12, 14	1
Block Size (bits)	128	Minimum 512
Efficiency	High	Low
Security	Adequately secured	Least secure
Power Consumption	Low	High
Pros	Provide higher security and also the efficiency with large key size	Computationally infeasible to compute private key given public key.
Cons	Difficult to implement in software	Slower process due to difficulty of factorisation

(Source:Lai and Heng, 2022)

2.4.4 Why Combine AES And RSA

- i. Enhanced security with two-factor authentication: The advantages of both RSA and AES are combined in hybrid cryptography. AES is usually used for data encryption due to its speed and efficiency, while RSA is used for secure transmission of the AES key (Barker, 2020). The performance of symmetric encryption and the advantages of asymmetric encryption for key management are both offered by this method.
- ii. Skillful key management: The key distribution issue that symmetric systems have is resolved in hybrid systems by using RSA to enable the safe exchange of AES keys. According to Garg and Carsten, 2016, this method works well in multi-user setups since each user can have their own set of RSA keys while sharing AES keys for particular documents or sessions.

2.4.5 Advantages And Disadvantages Of AES 256 Bits And RSA

Table 2.2 Advantages and Disadvantages of AES and RSA

Category	AES	RSA
Advantages	<ol style="list-style-type: none">1. High speed and efficiency2. Low resource consumption3. Resistant to all known practical attacks4. Supports larger key sizes for increased security5. Flexibility in implementation (hardware and software)	<ol style="list-style-type: none">1. Strong security for key exchange2. Widely trusted and used3. Facilitates digital signatures4. Allows secure data transmission5. Provides authentication and integrity checks
Disadvantages	<ol style="list-style-type: none">1. Key management can be complex2. Vulnerable to brute-force attacks if key size is insufficient3. Requires secure key distribution4. Implementation errors can lead to vulnerabilities5. Limited performance on constrained devices	<ol style="list-style-type: none">1. Slower than symmetric algorithms2. High computational cost3. Key generation can be resource-intensive4. Large key sizes required for high security5. Not suitable for encrypting large data volumes

2.4.6 Potential Vulnerabilities and Mitigations

Hybrid cryptography is vulnerable to assaults notwithstanding its advantages. Side-channel attacks are a serious risk because they take advantage of data that is disclosed during encryption procedures (Kocher and Paul, 2011). Vulnerabilities can also be introduced via implementation flaws, which highlights the significance of utilizing thoroughly examined cryptographic libraries. In the future, post-quantum cryptography research has been sparked

by the possibility that quantum computers could pose a challenge to RSA. To guarantee the long-term security of encrypted data, NIST is now standardizing quantum-resistant algorithms (Alagic and Gorjan, 2020).

2.4.7 Steganography

Steganography is the art and science of hidden data concealment—that is, the art of hiding information within carriers that appear to be harmless. While cryptography makes messages unreadable, steganography hides their very existence, providing an advantage in situations where encryption is prohibited or suspicious (Tyagi *et al.*, 2020). The phrase literally translates to "covered writing" and comes from the Greek terms "steganos" (covered) and "graphein" (writing) (Cheddad *et al.*, 2010).

Steganography's fundamental idea is to insert hidden information using digital media's redundancy without appreciably affecting the carrier's perceived quality. This can be accomplished via a variety of methods with diverse digital media types:

- i. Image steganography: It uses Least Significant Bit (LSB) insertion technique is a popular method for hiding secret information within digital images (Aditya *et al.*, 2024);
- ii. Audio steganography: It is a technique for hiding data within audio files, offering secure communication. Various methods exist, including Least Significant Bit (LSB), echo hiding, spread spectrum, and wavelet coding (Aslantaş and Hanilçi, 2022);
- iii. Text steganography: This technique conceals information inside text documents by using linguistic techniques or layout gimmicks (Agarwal, 2013); and
- iv. Video steganography: Hides data within video files using techniques like Least Significant Bit (LSB) insertion and Discrete Cosine Transform (DCT), enabling covert communication and digital watermarking.

Application of Steganography in Relation To Key Management

The combination of cryptography and steganography provides a potent method of information security. Steganography is employed in this context to conceal cryptographic elements within

cover media, such as encryption keys or encrypted data itself (Raphael and Sundaram, 2011).

Some of the uses includes:

- i. Hiding encryption keys: To further strengthen key management systems' security, cryptographic keys can be inserted into seemingly innocent photos or audio files (Gupta and Shoeb, 2019);
- ii. Hiding encrypted data: The encrypted data itself can be inserted steganographically to provide both confidentiality and covertness, either in place of or in addition to hiding keys (Muhammad *et al.*, 2018); and
- iii. Steganographic key exchange: By concealing important elements within cover media, steganography can help with secure key exchange protocols (Cutillo and Manulis, 2016).

Advantages of Steganography

Some advantages of steganography includes:

- i. Improved security by obfuscation: Steganography greatly improves key management system security by hiding sensitive cryptographic material's existence. This method of "security through obscurity" is an addition to conventional cryptographic protection;
- ii. Diminished visibility of security measures: Steganography makes it harder for attackers to identify targets by concealing keys or other cryptographic elements within seemingly normal files (Zielińska *et al.*, 2014);
- iii. Adding another line of defense: Extracting and accurately interpreting concealed data is a problem even for attackers who may guess its existence (Li *et al.*, 2011);
- iv. Plausible deniability: In some cases, the use of steganography might give plausible deniability, as the existence of hidden data can be difficult to show definitively (Kumar and Pooja, 2010); and
- v. Resistance to traffic analysis: Steganographic techniques can assist protect against traffic analysis assaults by obscuring the fact that secure communication is taking place.

Drawbacks of Steganography

- i. Limited data capacity: Steganographic techniques often have restrictions on the amount of data that can be hidden without noticeably altering the cover medium;
- ii. Vulnerability to detection: Advanced steganalysis techniques can potentially detect the presence of hidden data, especially if the steganographic method is known;
- iii. Susceptibility to manipulation: The hidden data can be damaged or destroyed if the stego-object undergoes common processing operations like compression or format conversion;
- iv. Lack of robustness: Many steganographic methods are not robust against intentional attacks aimed at removing or corrupting the hidden information; and
- v. Dependence on secrecy of method: The security of some steganographic systems relies on the secrecy of the hiding algorithm, which contradicts Kerckhoffs's principle in cryptography.

2.4.8 Implementing a Secure Personal Document Vault

Several essential elements go into implementing a safe Personal Document Vault with steganography and hybrid cryptography. Modules for document storage, encryption/decryption, key management, and user interface should all be included in the system architecture (Goyal *et al.*, 2021). Every document undergoes two rounds of encryption: first, the document is encrypted using the randomly generated AES key, and second, the AES key is encrypted using the user's RSA public key. Next, separate storage would be used for the encrypted document and the encrypted AES key (Kaliski, 2021). Using steganography for key management involves an extra step. It is possible to integrate the encrypted AES key using methods like least significant bit (LSB) insertion into a cover image. After that, the generated stego-image would be saved, and its position would be noted in a different, secure index (Lou and Liu, 2017). For a system like this, security and usability must be balanced in the user interface design. To make sure that security features don't get in the way of the user's workflow, user-centered security design principles should be included (Garfinkel and Lipford, 2014).

2.5 Related Works

An overview of relevant research articles that were evaluated for the "Personal Document Vault" project is given in this section. The table that follows highlights the background and contributions that were pertinent to the creation and improvement of the document vault system and lists the authors, methodologies, major conclusions, and gaps that were found in each study.

Table 2.3 Summary of Related Papers

Author	Methodology	Findings	Gaps
Neha, 2016	Evaluating hybrid encryption algorithms (AES with Twofish and AES with Blowfish) using Eclipse and Java.	AES with Twofish is more efficient than AES with Blowfish. Twofish has better performance characteristics.	Limited to small text files. Limited performance metrics. Lack of detailed quantitative results. No discussion of potential vulnerabilities.
Sharma <i>et al.</i> , 2021	Dividing data into three parts. Encrypting using 3DES and Blowfish. Implementing on Amazon Web Services.	Hybrid cryptographic approach using 3DES and Blowfish for enhanced security. Implementation on AWS.	Limited security analysis. Key management issues not addressed in depth. Scalability challenges not discussed.
Mishra and Levkowitz, 2021	Proposing blockchain-based Personal Data Vault (PDV) framework using Hyperledger Iroha and predictive prefetching.	Three-tier architecture for PDV. Use of lossless compression. Markov model for predicting data requests.	Scalability issues not addressed. Limited evaluation of predictive prefetching. Interaction with existing cloud services not discussed.

Henderson, 2010	Mixed-methods approach with field studies and surveys. Developing conceptual model of document management strategies.	Identified three strategies: piling, filing, and structuring. Developed user personas. Provided UI guidelines.	Small sample size. Discrepancies between survey and field study results. Strategies are general categories. Limited focus on collaborative scenarios.
Poduval <i>et al.</i> , 2019	User registration with steganography. File encryption using AES, 3DES, RC6. Key storage in steganographic image.	Secure storage and retrieval using hybrid cryptography and steganography. Improved data integrity and security.	Public key cryptography not included. Limited performance analysis. Scalability not discussed.
Selvanayagam <i>et al.</i> , 2018	Using symmetric-key cryptography (AES, DES, RC2) and potentially public-key cryptography for key exchange.	Proposed method for secure cloud storage using symmetric key cryptography. Use of ECC encryption.	Limited group sharing capabilities. Lack of extensive performance benchmarks. Focus on static data storage. Limited discussion on user management.

Sehgal and Goel, 2014	<p>Secret sharing phase: Generating and shares from the secret image using polynomial functions.</p> <p>Steganography phase: Hiding the shares into and cover images using an integer wavelet-based steganography method.</p> <p>Data extraction phase: Extracting the shares from the stego images and rebuilding the original secret image.</p>	<p>The paper discusses different aspects of steganography, including the general methodology of secret sharing, steganography, and data extraction phases, as well as evaluation tools to assess steganographic techniques. It also reviews related work on steganography techniques that combine methods like LSB, DCT, and compression to improve security and capacity, as well as techniques using secret sharing and integer wavelet to enhance security.</p>	<p>Need for better information security and confidentiality, potentially through the use of a verification code. Optimization required for trade-offs between key requirements of steganography (imperceptibility, robustness, and insertion capacity).</p> <p>Increase in embedding capacity needed by studying and exploiting human perceptual models.</p>
Muhammad <i>et al.</i> , 2018	<p>Granular ACLs.</p> <p>Trace-audit for visibility. Rule Recommender for high-level sharing policies.</p>	<p>PDV architecture decouples data capture from sharing.</p> <p>Fine-grained control over data sharing.</p> <p>Trace-audit mechanism for logging.</p>	<p>Existing systems often corporate-focused.</p> <p>Need for simpler personal document management.</p> <p>Challenges in database connection and design.</p>

Gamido <i>et al.</i> , 2023	Rapid Application Development (RAD) methodology, consisting of four phases: requirements planning, system design, construction, and cutover Agile and iterative development process Involving end-users in the development process from the planning stage to integrate their requirements and feedback Thorough testing of the system's functionality, user interface, and performance User acceptance testing (UAT) to ensure the system met the requirements and expectations of the end-users	The implementation of the EDMS significantly improved the efficiency of document management and dissemination processes, leading to increased productivity and reduced operating costs. The EDMS has been well-received by the employees and has resulted in improved work efficiency and reduced operating costs for document management and dissemination.	The EDMS was only tested by three colleges within the university, limiting the generalizability of the findings The EDMS should be upgraded to be more centralized and efficient for monitoring and tracking documents
--------------------------------	---	--	--

Shastri and Sharma, 2016	Review of data theft challenges and existing protection technologies. Proposing "Data Vault" conceptual model.	Data theft is a major cybersecurity issue. Existing technologies insufficient. Data Vault model aims to protect data in three states.	Difficulty in identifying unstructured sensitive data. Challenges in accurate detection. Integration issues. Need for research on mobile device security. Lack of practical implementation.
--------------------------	--	---	---

2.6 Proposed Solution

After the comprehensive review of the various literature, issues such as security, accessibility and management were identified. In order to close major gaps and apply industry best practices in encryption, access control, scalable architecture, and user interface design, the suggested solution makes use of the insights gleaned from the literature review. The Personal Document Vault (PDV) system seeks to deliver a secure, effective, and user-friendly document management solution that fulfills the needs of diverse users across multiple fields by focusing on strong security, scalability, intuitive interfaces, and productivity advantages.

CHAPTER 3

SYSTEM DESIGN AND METHODOLOGY

3.1 Overview of System Design and Methodology

This chapter highlights the methodology used in the software design process and in the development of a software system for Personal Document Vault. The approach combines established software engineering practices with specific considerations for secure document management, emphasizing data privacy, user authentication, and efficient document storage and retrieval.

3.2 Software Development Life Cycle

The Software Development Life Cycle (SDLC) is a systematic approach to creating high-quality software that meets user requirements (Khan *et al.*, 2020). It encompasses various phases, including planning, analysis, design, coding, testing, and implementation (Sharma, 2017). Several SDLC models exist, such as Waterfall, Spiral, V-Model, Iterative, Big Bang, Agile, and Rapid Application Development, each with its own advantages and disadvantages (Khan *et al.*, 2020). It plays a crucial role in creating reliable, secure, and efficient software products that can be maintained and upgraded over time (Shetty *et al.*, 2023). Agile development was used in the development of this project.

3.3 Agile Software Development

Agile software development is a lightweight, iterative approach that emphasizes flexibility, collaboration, and rapid delivery of functional software (Waja *et al.*, 2021). It emerged as an alternative to traditional methodologies, aiming to address their limitations and reduce overhead costs. Agile methods are characterized by frequent reassessment, adaptation of plans, and division of tasks into shorter iterations. While agile approaches offer advantages in flexibility and customer satisfaction, they may present security concerns due to changing requirements and variable team sizes (Manchanda *et al.*, 2017). Research on agile development has explored its foundations, practical applications, and future challenges (Dingsøyr *et al.*, 2010). The selection of an appropriate agile methodology depends on factors such as

project requirements, product sensitivity, and organizational structure (Al-Saqqa *et al.*, 2020). The reasons for choosing agile development for this project include:

- i. It reduces project development and completion time, as deliverables are isolated and worked on independently;
- ii. Customer feedback is taken into consideration in each iteration, ensuring customer satisfaction with the product developed; and
- iii. It facilitates changing customer requirements as problem and proposed solution become more apparent.

Figure 3.1 shows the various stages involved in agile software development process.

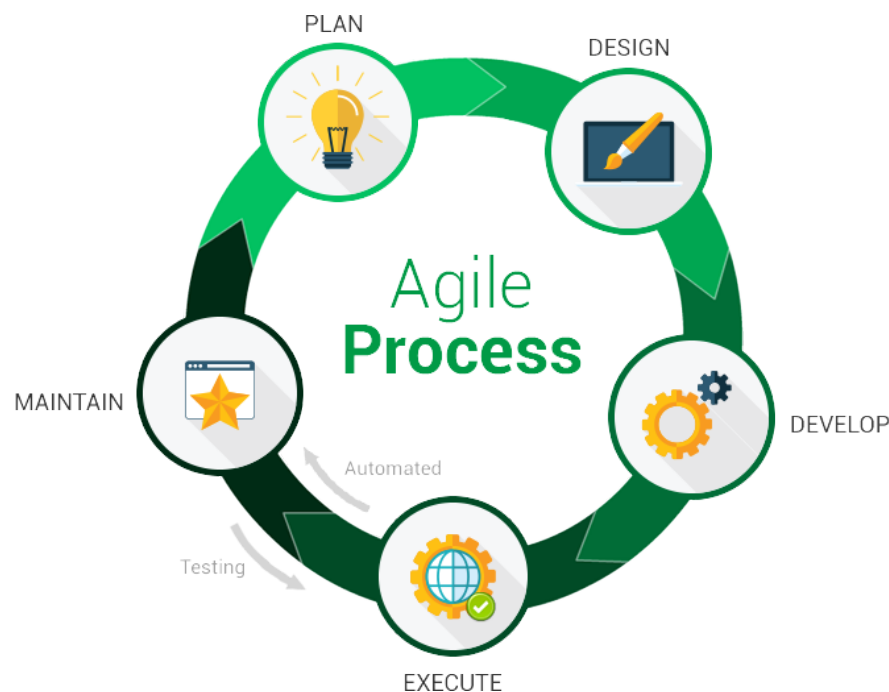


Figure 3.1 Stages of Agile Software Development (Source: Ewin, 2024)

3.3.1 Plan Stage

At the planning stage for the Personal Document Vault system, the project's scope and objectives were clearly defined. The main goal is to create a secure, user-friendly system that allows individuals to safely store, organize, and access their sensitive documents. The requirements of the system were collated using various requirement elicitation techniques. Requirements can be classified into functional and non-functional requirements. Functional

requirements describe the intended purpose of the system. They are requirements without which the system is incomplete,for my system, it includes:

- i. Encryption of document on upload;
- ii. Secure sharing of files;
- iii. Strict user authentication and multi-factor authentication;
- iv. Document categorization; and
- v. Search functionality.

Non-functional requirements, on the other hand, are qualities the system must possess or constraints the system has to satisfy. For my system,it includes:

- i. Security;
- ii. Usability;
- iii. Performance;
- iv. Reliability; and
- v. Scalability.

which ensure that the system operates effectively and meets user expectations.

The methods used for requirement gathering in this project include questionnaires, both open-ended and closed-ended questionnaires for key stakeholders, that is, primarily individual users concerned about document security. The product backlog will include features such as encrypted document upload, secure sharing, multi-factor authentication, and document categorization. Priority was given to those features based on security needs and user convenience.

3.3.2 Design Stage

Software design is a crucial phase in software development, translating requirements into detailed system representations (Abilov, 2013). It encompasses architectural design, which defines the high-level structure and components, and detailed design, which specifies component construction (Felici, 2010). The design process varies across different development approaches, with traditional models following a sequential flow and agile methods employing

iterative refinement (Subhan *et al.*, 2015). System design supports stakeholder communication, system analysis, and large-scale reuse, utilizing various structures and viewpoints (Felici, 2010). The success of software development heavily depends on effective requirements analysis and design phases, which are closely linked to other stages in the development lifecycle (Subhan *et al.*, 2015).

Three Tier Architecture

Figure 3.2 shows the configuration for a Personal Document Vault web system. A PDV client with a computer that has access to the Internet, running a browser. The client communicates with the application server via the Hypertext Transfer Protocol (HTTP). The application server in turn executes commands against the database, formats the result in Hypertext Markup Language (HTML), and return the result to the client.

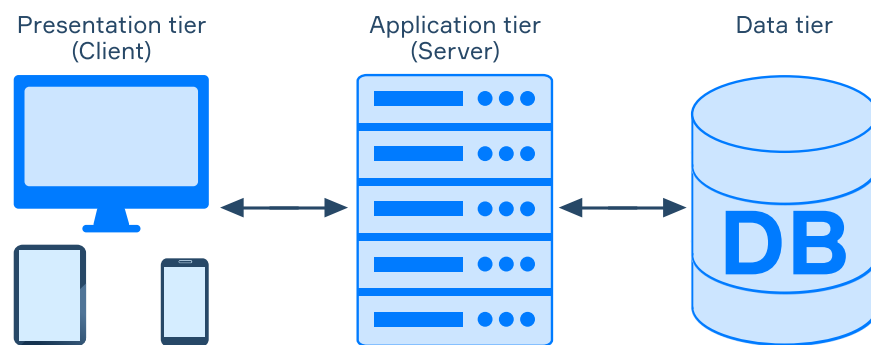


Figure 3.2 Three Tier Architecture (Source: McCall, 2024)

Use Case Diagram

Use case is a methodology used in system analysis to identify, explicate and organise system requirements (Aleryani, 2016). Use case diagrams are used in Unified Modelling Language (UML), a standard notation for modelling objects and systems, to depict how a system interacts with various external entities. The proposed system has only one actor, that is a PDV user. The PDV user has to create an account and sign-in before he can access the following pages:

- i. Dashboard to view security tips, vault statistics, info about the user and track user's activity;
- ii. On the folder page, the user can search for folders, create new folders, and when a folder is clicked, a popup for a Google authentication code will appear. If true, open

the folder; otherwise, the prompt is wrong. key. Inside a folder, the user can create subfolders and upload files. For every subfolder, you can perform the following actions: view description, update folder, and delete folder. On the delete folder, you can move it to trash or delete it permanently. For every file, you can decrypt the file. If decryption is successful, a popup for sharing or downloading appears. It also has an update file and a delete file, where on the delete file you can trash it or delete it permanently;

- iii. On key page user can search a particular key with the name used in uploading the file if found a qr code image will return, user drag the image into uploading field and the private key is displayed in code blocks which there is a button to be able to copy and send it to decrypt file to decrypt;
- iv. On tools page you can upload any image with text to use ocr to extract the text, the text can be copied into an editor on same page, edit it and export it;
- v. On settings page users can change the google authentication ID used in generating TOTP by scanning a QR code image, verify it for the MFA settings to reflect; and
- vi. On trash page, users can view temporarily deleted folder and files and they can decide to delete it temporarily or restore them back to their respective folders.

Figure 3.3 shows the use case diagram of the proposed system, indicating the PDV actor and how they can interact with the system.

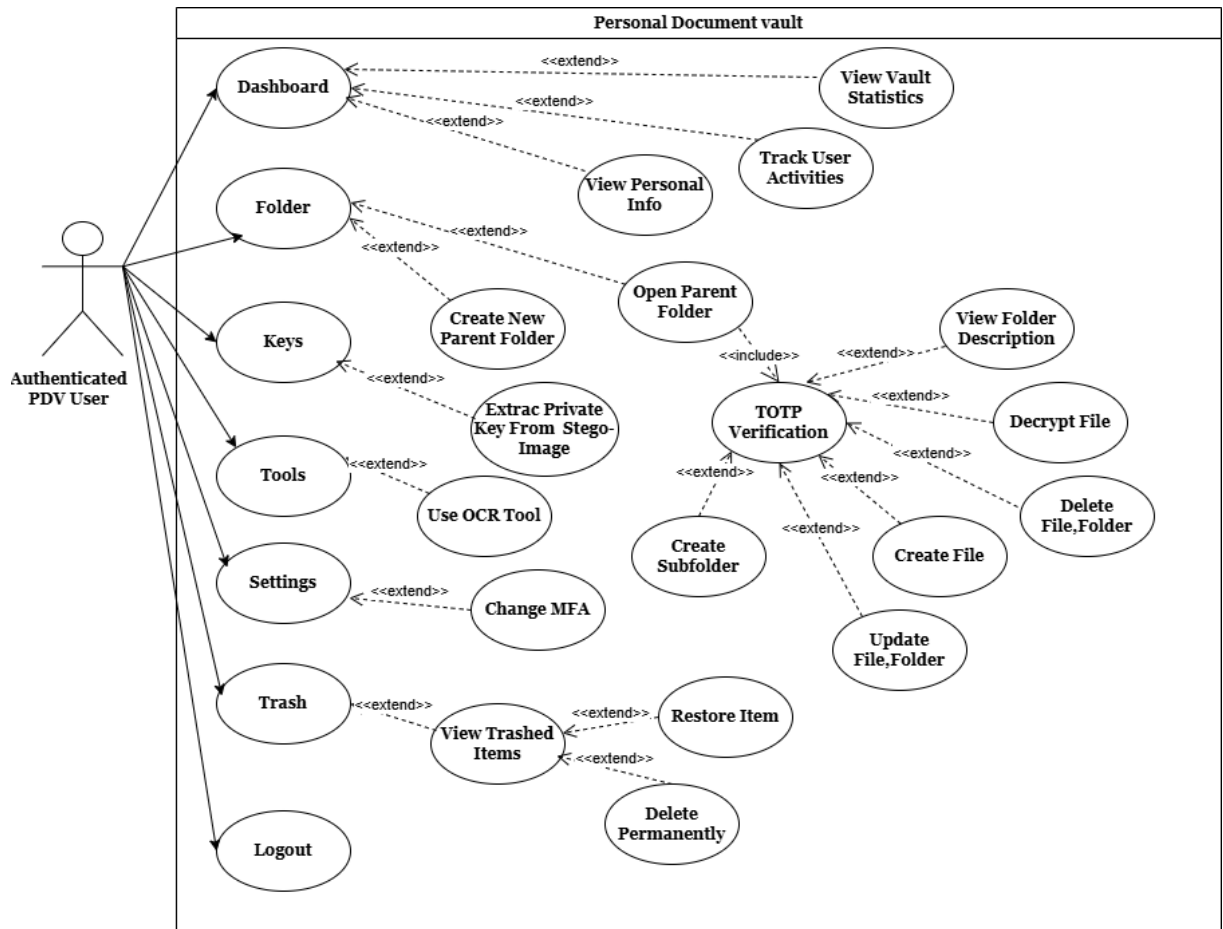


Figure 3.3 PDV Use Case Diagram

Flowchart

Flowcharts are graphical representations used to describe algorithms, procedures, and business processes. They employ symbols to illustrate step-by-step sequences of operations, making complex systems more understandable (Aloun *et al.*, 2017).

Figure 3.4 illustrates the program flowchart of the document encryption process. The process begins with uploading a file **F**, followed by reading its content. Concurrently, two separate operations are performed: generating an RSA key pair and an AES 256 key. The RSA key pair consists of a private key **K(private)** and a public key **K(public)**. The private key is concealed into a QR code image using steganography techniques. The AES 256 key is encrypted using the public RSA key. The content of file **F** is then encrypted using the AES 256 key. Finally, the encrypted file content, the encrypted AES 256 key, and the QR code image containing the concealed private RSA key are stored in the database. This encryption scheme combines asymmetric and symmetric encryption methods along with steganography

to enhance data security. The process concludes once all these items are successfully stored in the database.

As shown in Figure 3.5 depicts the document decryption process, it begins by retrieving data from the database, including the Stego-QR code image and the encrypted AES 256 key (eK) along with the encrypted file content (eF). The RSA private key $K(\text{private})$ is extracted from the Stego-QR code image. This private key is then used to decrypt the encrypted AES 256 key. If the decryption is successful, the decrypted AES 256 key is used to decrypt the encrypted file content. The decrypted content is then written into the respective file format as originally uploaded. Finally, the decrypted file is made available for download. If the decryption of the AES 256 key is unsuccessful, the process discards the current attempt and requests the $K(\text{private})$ again to initiate another decryption attempt.

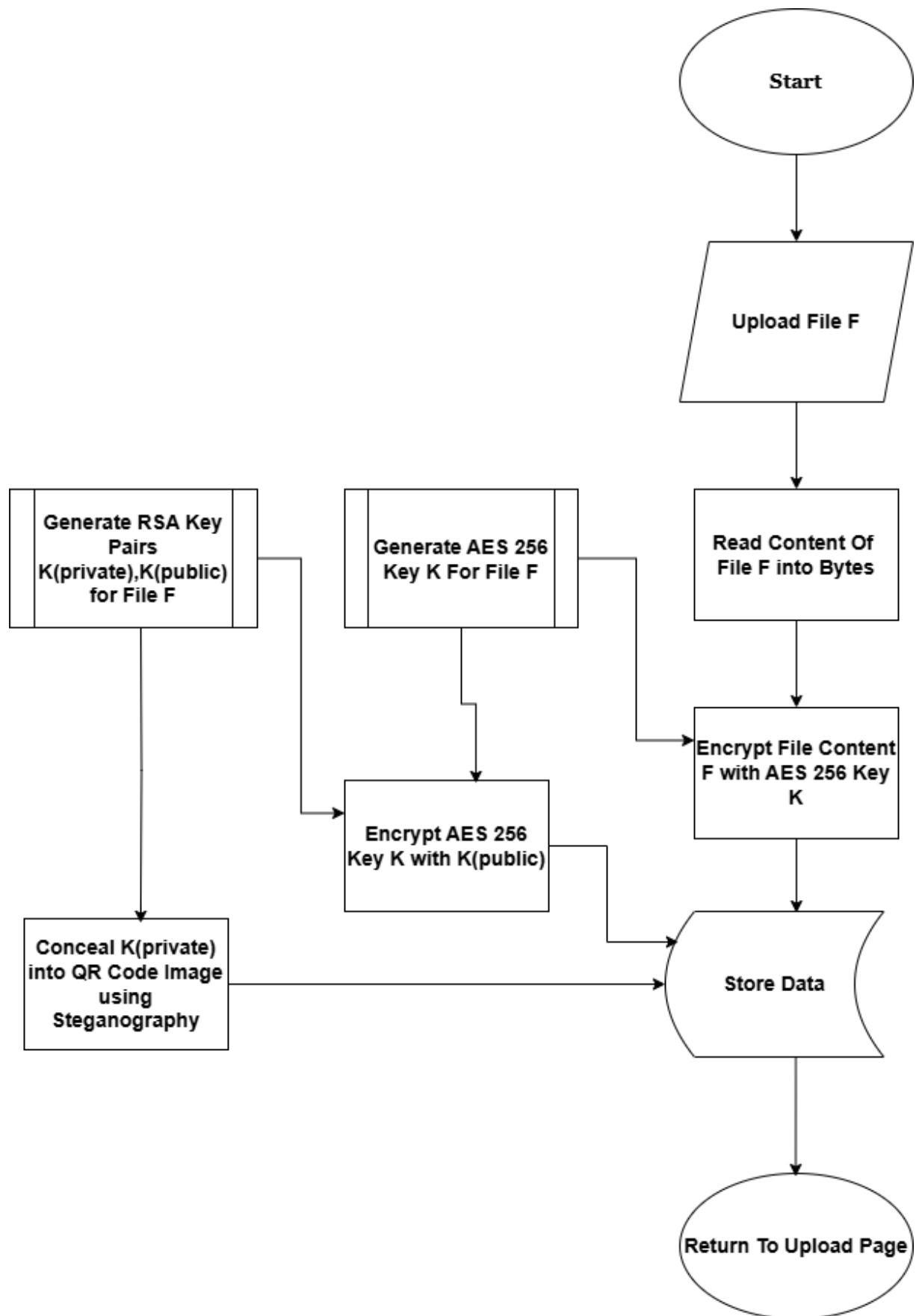


Figure 3.4 Document Encryption Process

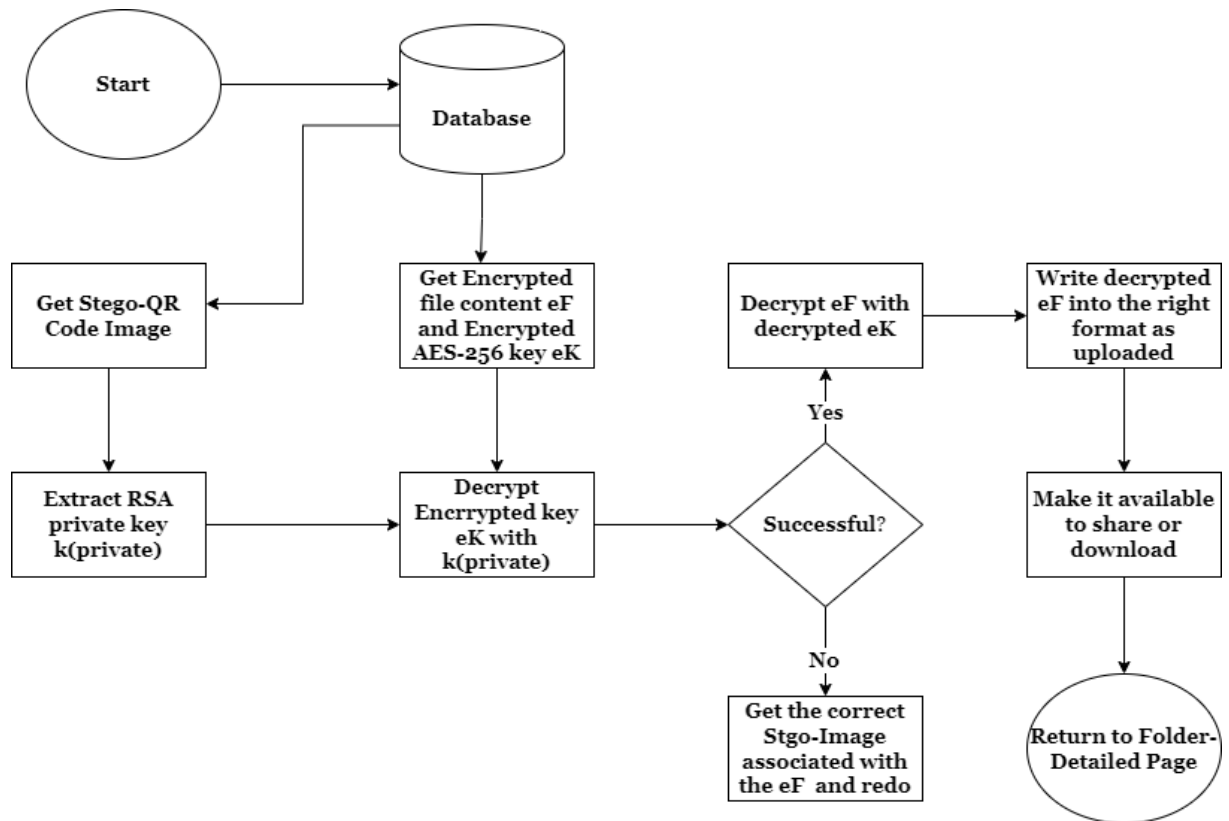


Figure 3.5 Document Decryption Process

3.4 Development Stage

System implementation is another name for the development stage. At this point, the system design was transformed into a working software system from a conceptual framework. The frontend, backend, and database are the three primary parts of the software system.

Frontend

The frontend is the user-facing part of the software system, serving as an interface between PDV users and the backend. It was developed using standard web technologies and popular libraries to create a responsive and interactive user experience. Table 3.2 shows the frontend technologies used in this project and their application.

Table 3.1 Frontend Technologies Used and Their Applications

Technology	Application
HTML	Structure and layout of web pages
CSS	Styling of HTML elements for visual presentation
JavaScript	Adds interactivity and dynamic behavior to web pages
Bootstrap	Provides responsive design framework and pre-built components for faster development
Font Awesome	Offers a library of scalable vector icons to enhance visual elements
Dropzone	Implements drag-and-drop file upload functionality for improved user experience
DataTables	Enhances HTML tables with advanced interaction controls like pagination and instant search
SweetAlert	Creates customizable JavaScript alert boxes for better user interaction

This combination of these technologies enables the creation of a modern, responsive, and feature-rich frontend. That makes it compatible on different platforms and improves the user experience.

Backend

The backend of the proposed system was developed using Django, a high-level Python web framework. Django follows the model-template-view architectural pattern and provides a robust set of tools for building web applications. In addition to Django, several specialized Python libraries were utilized to implement specific functionalities. Table 3.1 provides a summary of the key libraries used and their applications in the implementation of the backend of the software system.

Table 3.2 Key Libraries Used in the Backend

Library	Application
Python cryptography	Implementation of cryptographic protocols and algorithms for secure data handling
Stegano	Steganography operations, allowing data hiding within digital files
PyOTP	Generation and verification of one-time passwords for enhanced authentication
PyTesseract	Optical Character Recognition (OCR) to extract text from images

Database

The backend of the proposed Personal Document Vault system utilizes PostgreSQL as its database management system. PostgreSQL is a powerful, open-source relational database that offers robust features for data integrity, security, and performance.

PostgreSQL was chosen for its ability to handle complex data structures, which is crucial for storing various document types and associated metadata in the vault. Its advanced security features, including row-level security and strong encryption support, align well with the high security requirements of a personal document storage system.

The database schema was designed to efficiently store and retrieve user information, document metadata, access logs, and encryption keys. PostgreSQL's ACID compliance ensures that all transactions related to document uploads, updates, and access are processed reliably, maintaining data consistency even in case of system failures.

To optimize query performance, especially for full-text search capabilities within stored documents, PostgreSQL's built-in indexing and full-text search features were leveraged. This allows users to quickly locate specific documents or content within their vault.

Additionally, PostgreSQL's extensibility allowed for the implementation of custom functions to handle document versioning and audit trails, providing users with a comprehensive history of their document activities.

The scalability of PostgreSQL also ensures that the Personal Document Vault can grow with user needs, supporting an increasing number of documents and users without compromising on performance or data integrity.

3.4.1 Execute Stage

At the execute, stage, the system was tested and deployed.

Testing

The Personal Document Vault system was extensively tested at this point to guarantee its safe and faultless functioning. To confirm the system's usability, security, and functioning, several types of testing were carried out. The tests listed below were used:

i. Unit Testing

Evaluates performance and functionality of each component. It includes encryption modules, authentication mechanisms, and document management features.

ii. Integration Testing

Verifies compatibility and seamless interaction between system components

iii. Security Testing

Assesses robustness of encryption methods (AES-256 and RSA), tests effectiveness of QR code-based key management and evaluates strength of authentication methods including 2FA.

iv. System Testing

Evaluates system functionality including document encryption, secure communications, version control, and activity logging.

v. Performance Testing

Verifies system's efficiency in handling document uploads, searches, and retrievals.

vi. Usability Testing

Assesses user-friendliness of interface and intuitiveness of document management features.

The tests aim to verify if Personal Document Vault effectively delivers on its goals of security, efficient document management, and enhanced productivity while maintaining a user-friendly interface.

Deployment

At this phase, the Personal Document System was hosted online at Heroku.

3.4.2 Maintain Stage

After Personal Document Vault system was deployed, it was subjected to undergo recurring maintenance and improvements depending on user input to boost efficiency.

CHAPTER 4

SYSTEM TESTING AND RESULTS

4.1 Overview of System Testing and Results

This chapter presents the findings of the system testing of the proposed Personal Document Vault (PDV) system, evaluating its compliance with the system requirements. The PDV system has one main end-user: the owner managing its personal documents. The system primarily consists of a single interface for users to securely store, organize, and manage their documents, enhancing productivity and workflow efficiency.

4.2 Personal Document Vault User Interface

The Personal Document Vault Interface was made specifically for the intended user. The system starts with a landing page before they can access other pages from the 'start securing button,' as shown in Figure 4.1.

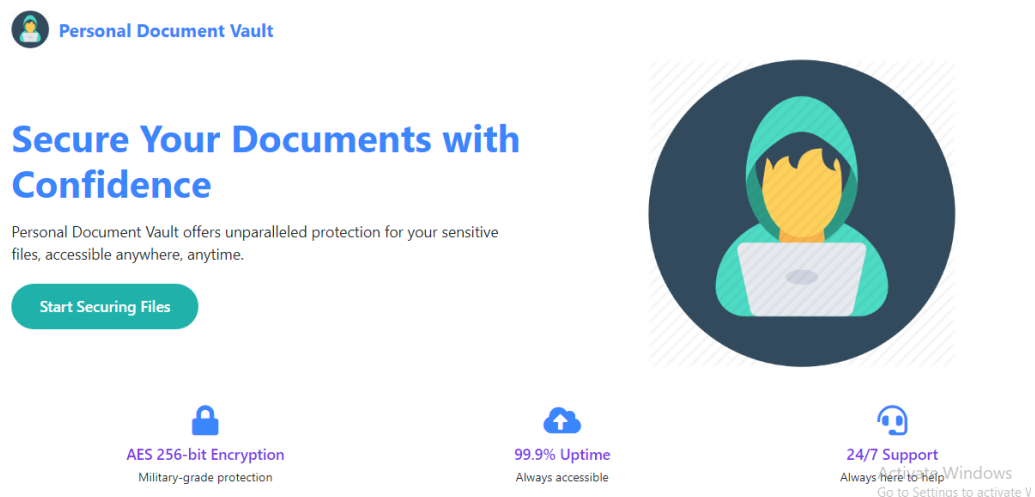


Figure 4.1 PDV Landing Page

4.2.1 Authentication

A PDV user must be authenticated before accessing the vault pages. From the landing page, the user will be directed to a log-in page, where they can sign in with a social account (GitHub or Google) or PDV account credentials. If the user has forgotten their PDV password, they can recreate a new one; otherwise, they can register for a PDV account. These activities are shown in Figures 4.2, 4.3, and 4.4, respectively.

Also, a registered user needs to pair their Google Authenticator mobile app to the system for receiving TOTP's and verifying them before proceeding to the main page, which is the vault as shown in Figure 4.5 and Figure 4.6 respectively.

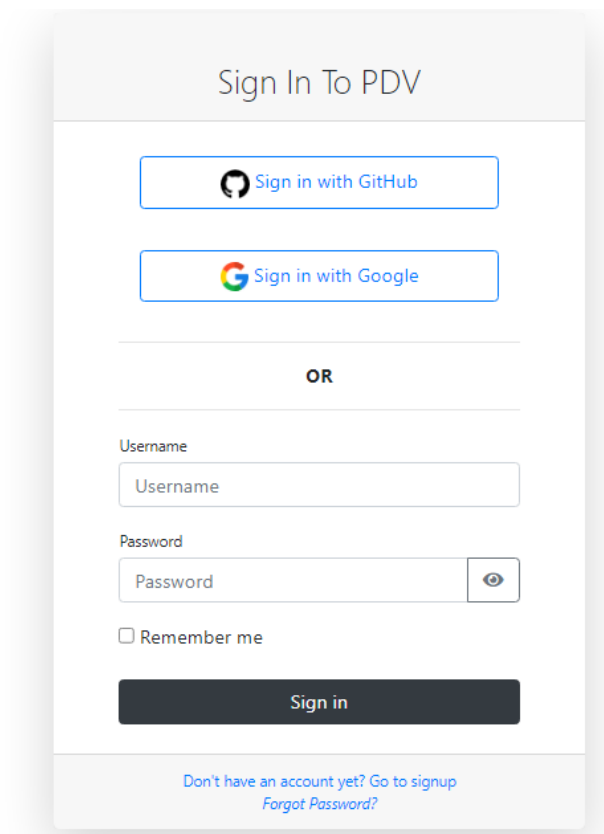
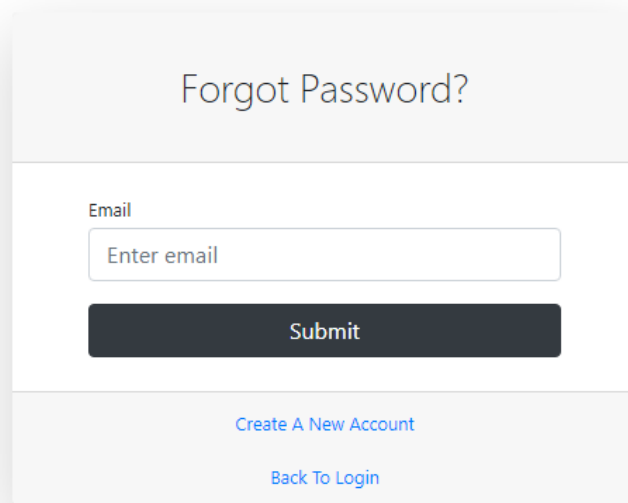
The image shows a mobile app login screen titled "Sign In To PDV". At the top, there are two buttons: "Sign in with GitHub" and "Sign in with Google". Below these is a horizontal line with the word "OR" in the center. Underneath the line are two input fields: "Username" and "Password". The "Password" field has a toggle icon (an eye) to its right. Below the input fields is a checkbox labeled "Remember me". At the bottom of the form is a dark grey button labeled "Sign in". At the very bottom of the screen, there are two links: "Don't have an account yet? Go to signup" and "Forgot Password?".

Figure 4.2 Login Page



Forgot Password?

Email

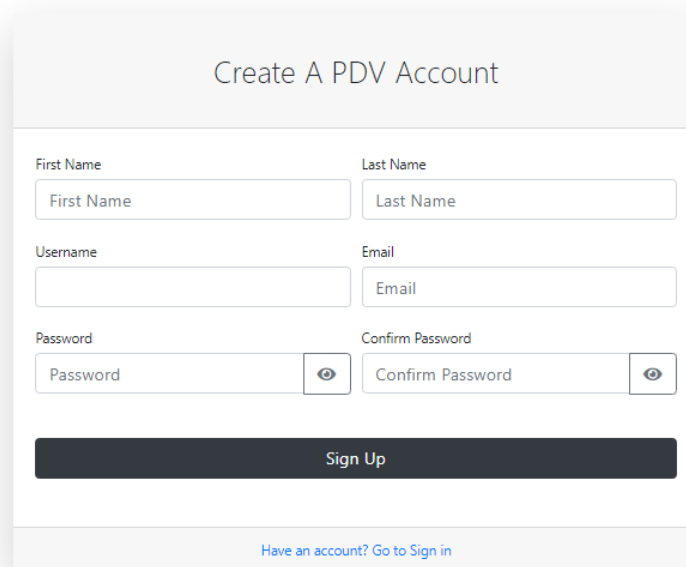
Submit

[Create A New Account](#)

[Back To Login](#)

This is a 'Forgot Password?' form. It features a light gray header with the title. Below is a white form area with an 'Email' label, a text input field containing the placeholder 'Enter email', and a dark gray 'Submit' button. At the bottom of the form area, there are two blue links: 'Create A New Account' and 'Back To Login'.

Figure 4.3 Forgot Password Page



Create A PDV Account

First Name Last Name

Username Email

Password Confirm Password

Sign Up

[Have an account? Go to Sign in](#)

This is a 'Create A PDV Account' form. It has a light gray header with the title. The form area is white and contains several input fields: 'First Name', 'Last Name', 'Username', 'Email', 'Password', and 'Confirm Password'. The 'Password' and 'Confirm Password' fields have eye icons for toggling visibility. A dark gray 'Sign Up' button is at the bottom of the form area. Below the form area, there is a blue link: 'Have an account? Go to Sign in'.

Figure 4.4 Register Page



Figure 4.5 Account Pairing Page

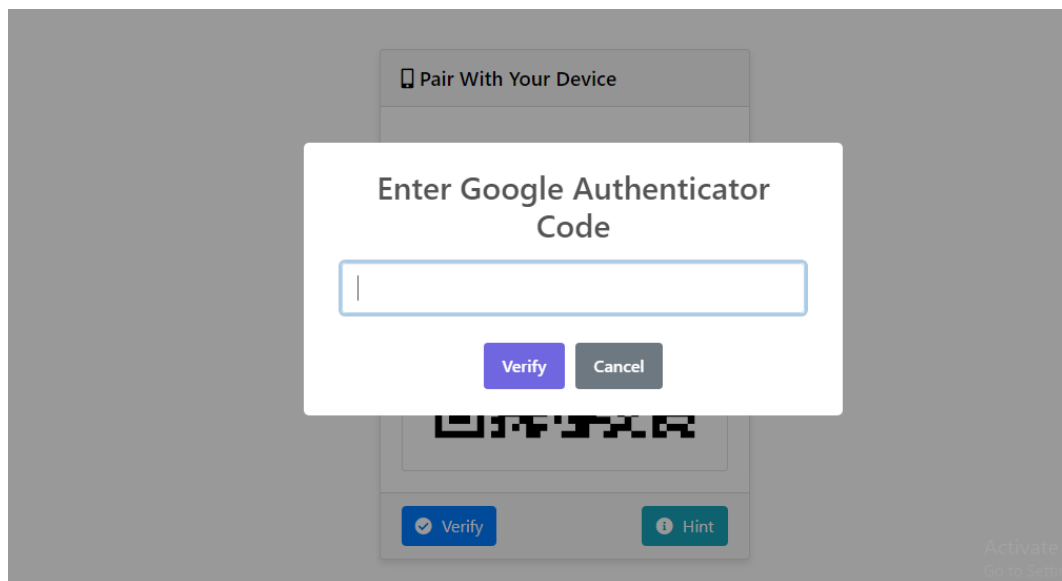


Figure 4.6 Verify Page

4.2.2 Vault

Upon successful login, PDV users are redirected to the main page called the vault, where they can:

- i. **Access Dashboard:** PDV users can view vault statistics, track activities performed on the vault, and view personal info as shown in Figures 4.7, 4.8, and 4.9;
- ii. **View Folders:** Users can create new folders, view parent folders, and enter folders using TOTPs. Inside folders, users can create subfolders, upload files, delete and rename files and folders, and decrypt files, as shown in Figures 4.11 and 4.12;
- iii. **Extract Keys:** Users can search for a file to decrypt, drag the QR Code image into the file uploader, and copy the keys for decryption. Figure 4.13 depicts this;
- iv. **Use Tools:** Users can use OCR to extract text from an image, edit it, and export it as a PDF, as shown in Figure 4.14;
- v. **Change MFA Settings:** Users can request a new TOTP if needed and will be required to provide a 6-digit code sent to their email. The page is shown in Figure 4.15; and
- vi. **View Trashed Items:** Temporarily deleted folders or files can be restored or permanently deleted, as shown in Figure 4.16.

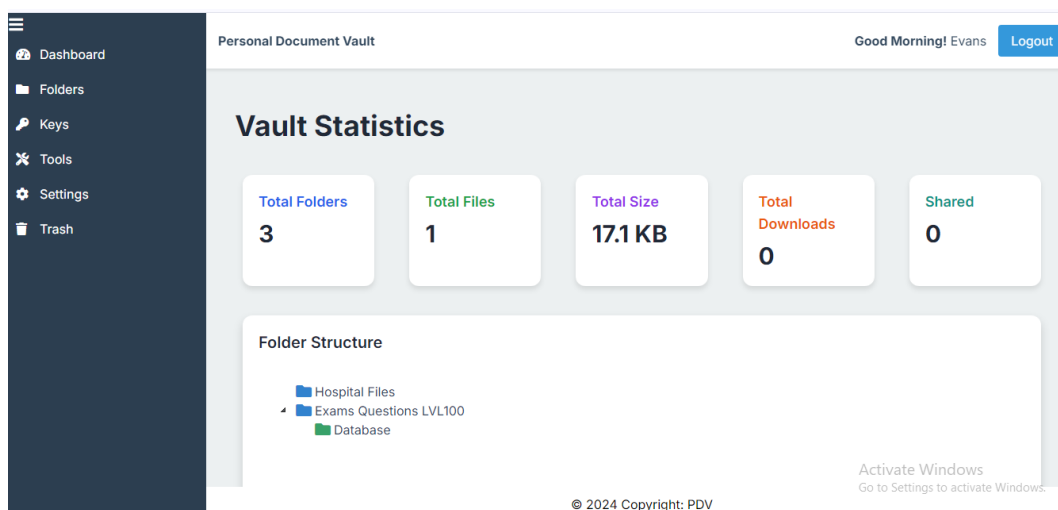


Figure 4.7 Vault Statistics

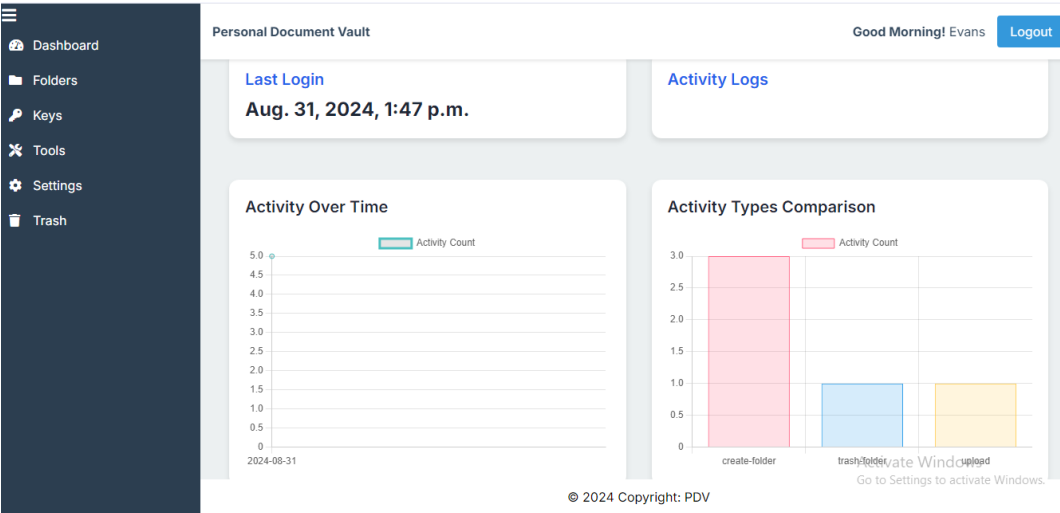


Figure 4.8 Activity Tracking Graphs

Personal Document Vault

Good Morning! Evans Logout

Activity Log

Copy Excel CSV PDF

Search

Activity ID	Action Performed On	Activity Type	Time Performed
8e93c8a6-bc0a-4bc0-b411-c3c4212a66a8	Hospital Files	create-folder	Aug. 31, 2024, 1:50 p.m.
99795ca2-3903-47d0-839e-ae79a20548ce	Database	trash-folder	Aug. 31, 2024, 2:05 p.m.
ba48bf6c-e2b3-4f45-819b-65c888c4d1ae	Database	create-folder	Aug. 31, 2024, 1:58 p.m.
c59adedd-c797-4c26-8c69-a29bb0d053e6	Exams Questions LVL100	create-folder	Aug. 31, 2024, 1:51 p.m.
ca31ec0c-2be1-4bf5-9861-fb16b6d45ee8	Info	upload	Aug. 31, 2024, 1:58 p.m.

Showing 1 to 5 of 5 entries

Previous 1 Next

Activate Windows
Go to Settings to activate Windows.

© 2024 Copyright: PDV

Figure 4.9 Activity Tracking Table

Personal Document Vault

Good Morning! Evans Logout

Welcome, Evans!
Here's an overview of your account information.

Email
arthurtakyprince1234@gmail.com

Joined Date
August 31, 2024

MFA Status
Enabled

Last Login
August 31, 2024 13:47

Account Actions
Edit Profile Change Password

Activate Windows
Go to Settings to activate Windows.

© 2024 Copyright: PDV

Figure 4.10 User Info

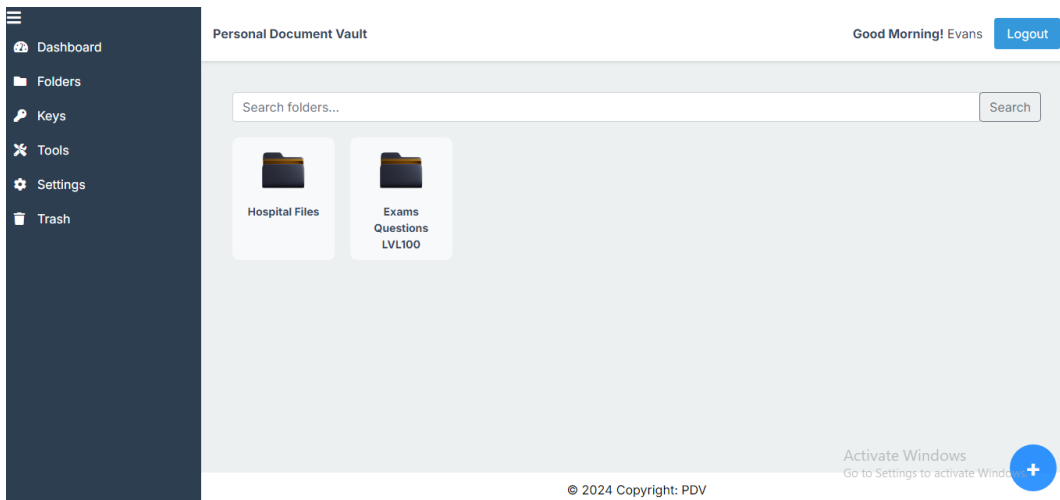


Figure 4.11 Folders

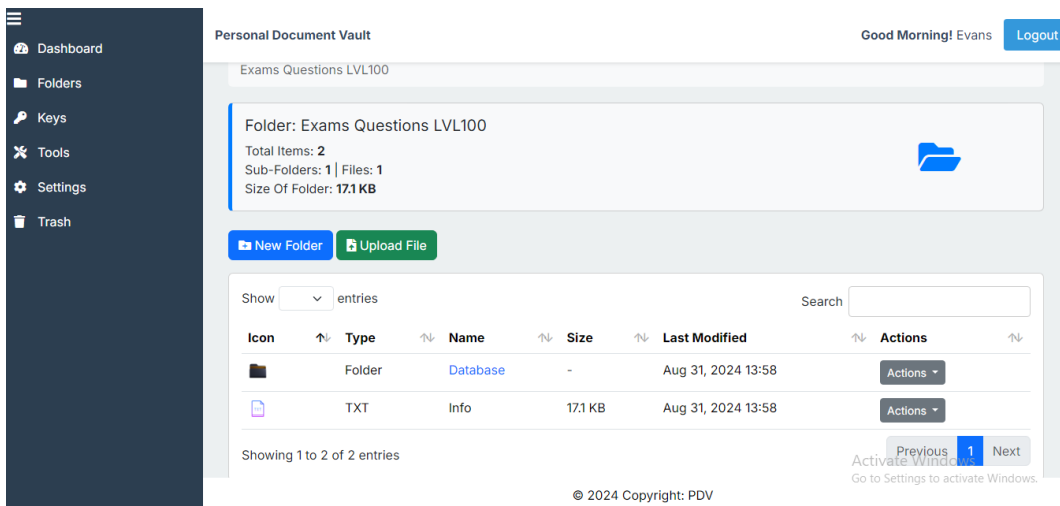


Figure 4.12 Inside a Folder

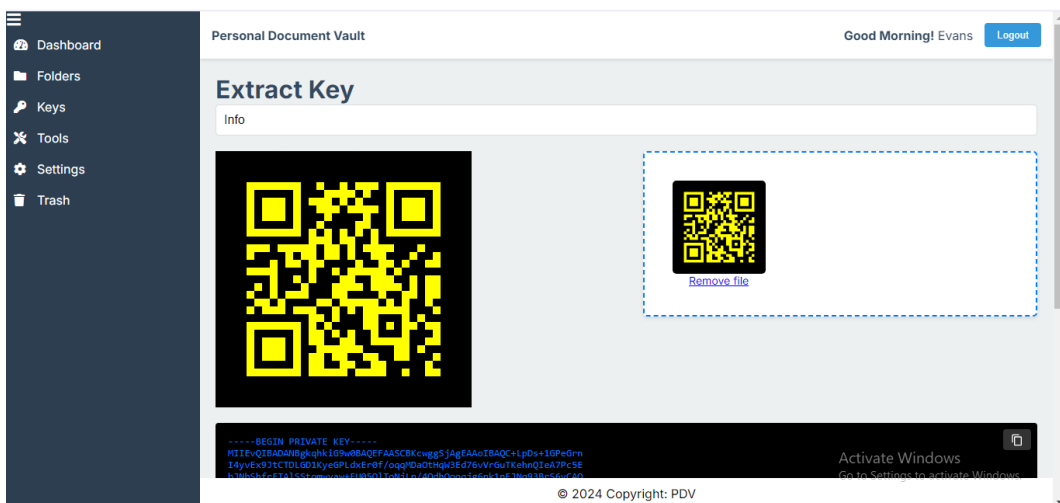


Figure 4.13 Extract Key

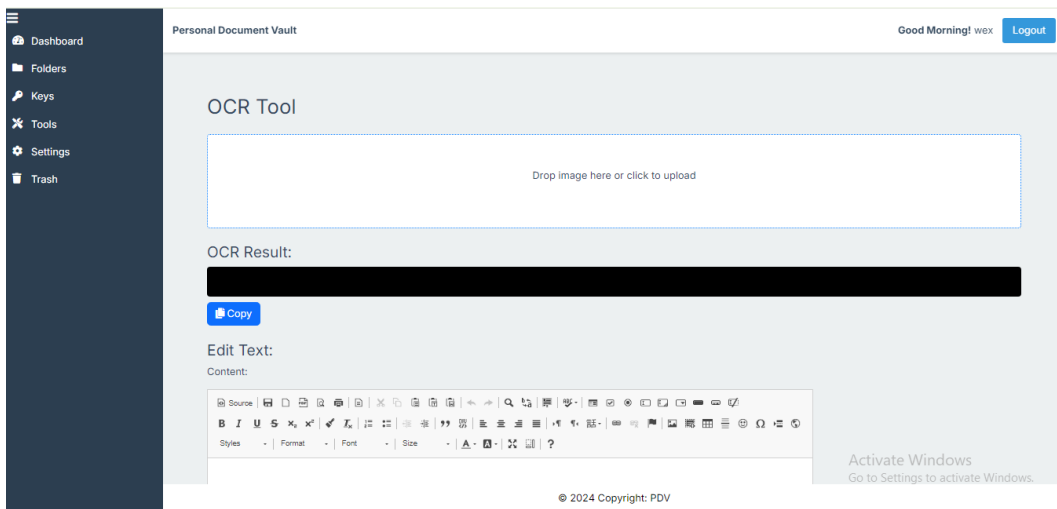


Figure 4.14 OCR and Editor Tool

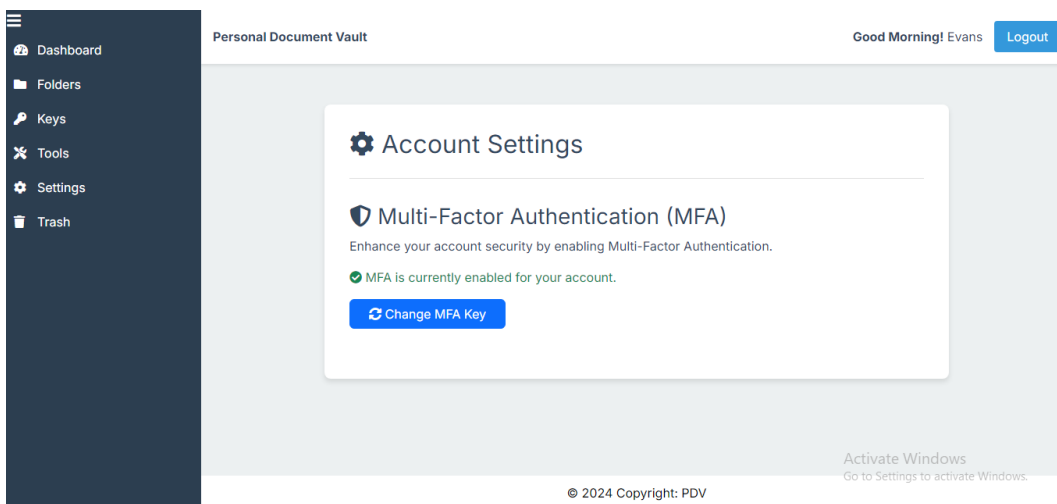


Figure 4.15 MFA Settings

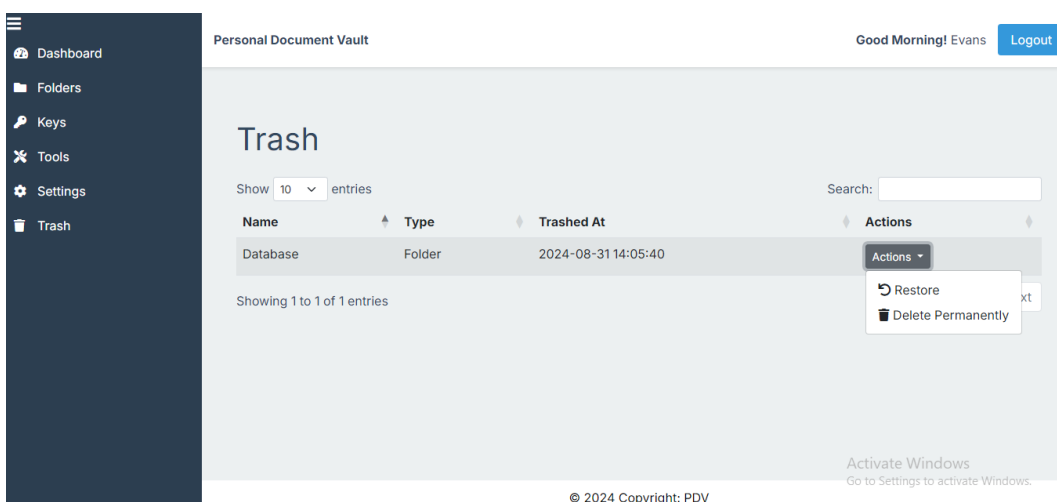


Figure 4.16 Trash

4.3 Error Pages

The Personal Document Vault's error pages ensure a smooth user experience by offering clear communication, common error guidance, and system security. They include:

- i. Page Not Found (404): When a user attempts to access a document that no longer exists. This can occur due to changes in the document's unique identifier, deletion, relocation, or outdated link, as depicted in Figure 4.17.
- ii. Internal Server Error (500): Indicates a server-side problem, possibly due to database connection failures, server overload, or server-side code conflicts, as depicted in Figure 4.18.
- iii. Forbidden (403): When a user attempts to access a document or perform an action without the necessary permissions, as depicted in Figure 4.19.
- iv. Bad Request (400): When a user sends an invalid request to the server, potentially due to unsupported file types, malformed data, or outdated client-side applications, as depicted in Figure 4.20.

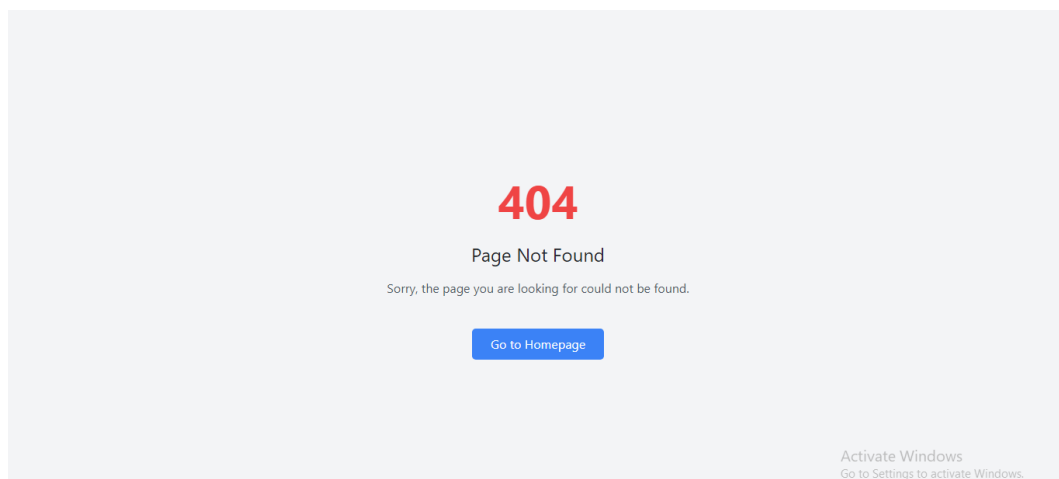


Figure 4.17 Page Not Found (404)

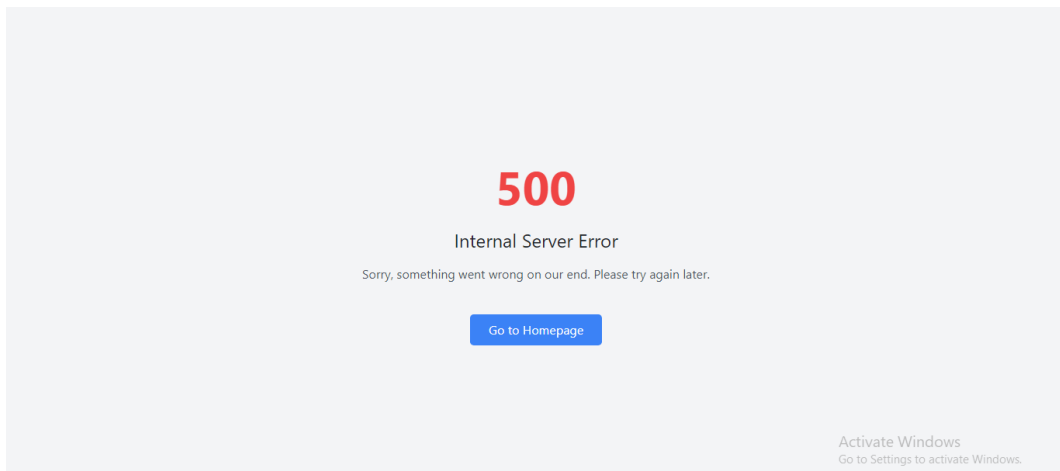


Figure 4.18 Internal Server Error (500)

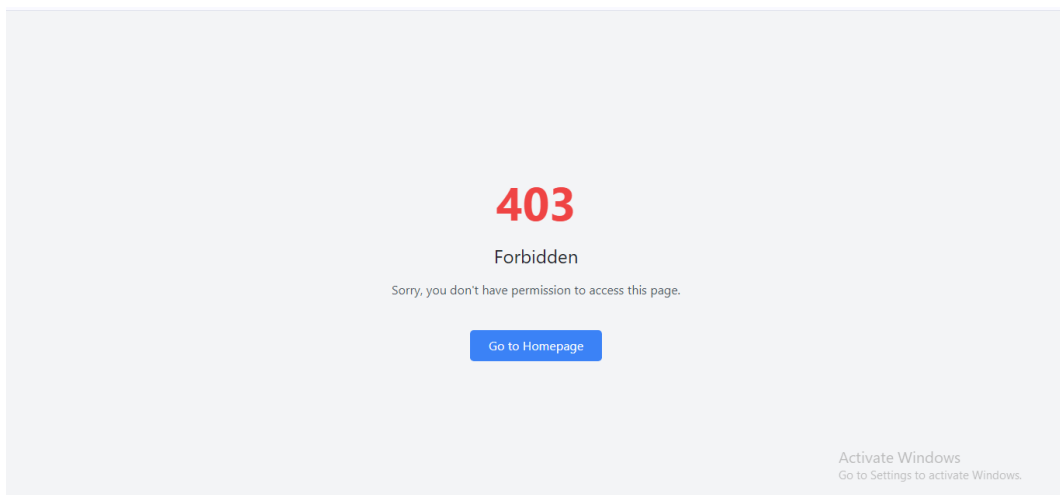


Figure 4.19 Forbidden (403)

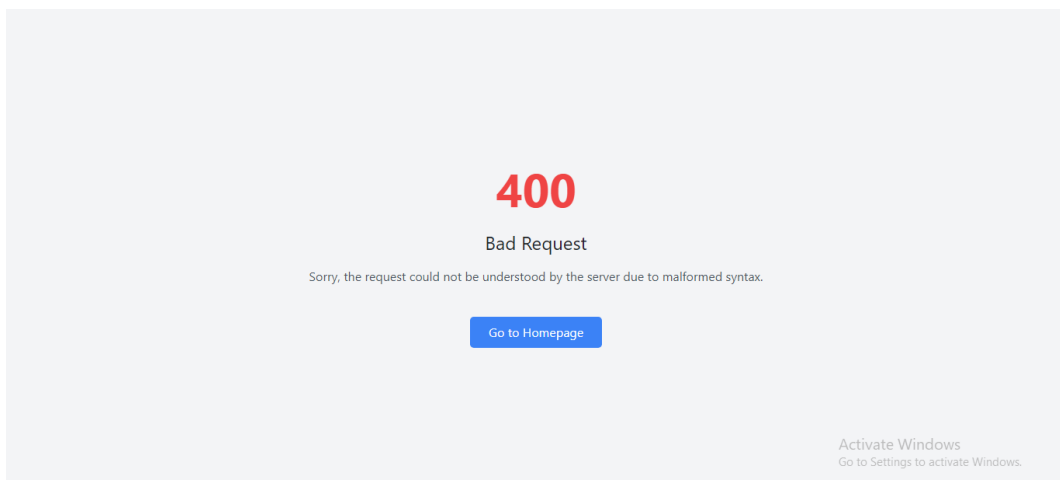


Figure 4.20 Bad Request (400)

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

Conclusion

A PDV system was developed to provide secure, efficient management of personal documents in the digital age. The Personal Document Vault successfully implements multi-layered security measures, a centralized document repository, intuitive organization features, and a user-friendly interface. The system addresses challenges such as inadequate protection and inefficient workflow in digital document management, ensuring confidentiality and streamlining document handling processes.

This solution facilitates easy access to and management of personal documents, improving user efficiency and reducing the risk of data loss or unauthorized access. By enhancing document security and accessibility, the Personal Document Vault aims to increase user confidence in digital document storage, potentially leading to wider adoption of digital document management practices.

The Personal Document Vault project lays the groundwork for future advancements in safe, user-focused digital solutions by showcasing how technology can be used to solve current issues with personal information management.

5.2 Recommendation and Future Works

For the Personal Document Vault system to be more effective and adaptable to future needs, the following recommendations are suggested for future research and development:

- i. **Cloud Integration and Synchronization:** Provides safe synchronization of documents between several devices.
- ii. **Collaborative Features:** The vault's usefulness may be expanded to small company or home environments with its secure sharing and collaborative editing features.

- iii. **Blockchain Integration:** Increases security and transparency by improving audit trails and document verification.
- iv. **Automated Version Control and Backup:** Maintains track of document modifications over time and rolls back to earlier iterations.
- v. **Machine Learning for Personalization:** To provide individualized recommendations, this technique examines usage trends and user behavior.

REFERENCES

- Abilov, M. (2013), “Bridging the gap between requirements and object-oriented models”, *Journal of Software Engineering*, , Vol. 45, No. 3, pp. 215–230.
- Aditya, S., Ved, M., Shashikant, K., Samadhan, K., and Shilpa, M. A. (2024), “Image Steganography Using Least Significant Bit”, *International Journal of Research Publication and Reviews*, URL: <https://api.semanticscholar.org/CorpusID:269182726>.
- Agarwal, M. (2013), “Text steganographic approaches: a comparison”, *International Journal of Network Security & Its Applications*, , Vol. 5, No. 1, pp. 23–91.
- Al-Saqqa, S., Sawalha, S., and Abdelnabi, H. (2020), “Agile Software Development: Methodologies and Trends”, *Int. J. Interact. Mob. Technol.*, , Vol. 14, pp. 246–270, URL: <https://api.semanticscholar.org/CorpusID:225548331>.
- Alagic, B. and Gorjan, C. (2020), *Status report on the second round of the NIST post-quantum cryptography standardization process*, tech. rep., NIST.
- Aleryani, A. Y. (2016), “Comparative Study between Data Flow Diagram and Use Case Diagram”, *International Journal of Scientific and Research Publications*, , Vol. 6, No. 3, pp. 124–127.
- Aloun, P., Malík, M., Andresic, D., and Nespesny, D. (2017), “Using eyetracking to analyse how flowcharts are understood”, *2017 IEEE 14th International Scientific Conference on Informatics*, pp. 394–399, URL: <https://api.semanticscholar.org/CorpusID:4564094>.
- Aslantaş, F. and Hanilçi, C. (2022), “Comparative Analysis Of Audio Steganography Methods”, *Journal of Innovative Science and Engineering (JISE)*, URL: <https://api.semanticscholar.org/CorpusID:247340110>.
- Barker, E. (2020), *Recommendation for key management: Part 1 General*, tech. rep. Special Publication 800-57 Part 1 Revision 5, NIST.
- Berlin, K and Dhenakaran, S. (2017), “An Overview of Cryptanalysis of RSA Public key System”, *International journal of engineering and technology*, , Vol. 9, pp. 3575–3579, URL: <https://api.semanticscholar.org/CorpusID:51987116>.
- Brown, S. (2021), “Vulnerabilities in personal device document storage”, *Cybersecurity Today*, , Vol. 18, No. 2, pp. 78–92.

- Cheddad, A., Condell, J., Curran, K., and Mc Kevitt, P. (2010), “Digital image steganography: Survey and analysis of current methods”, *Signal processing* , Vol. 90, No. 3, pp. 727–752.
- Committee on National Security Systems (2015), *Use of Public Standards for the Secure Sharing of Information Among National Security Systems*, tech. rep. CNSS Policy No. 15.
- Cuttillo, L. A. and Manulis, M. (2016), “Covert steganographic network: A peer-to-peer covert information-sharing system”, *Security and Trust Management: 12th International Workshop, STM 2016*, Springer International Publishing, pp. 15–30.
- Davis, R. (2023), *Designing secure personal document management systems*, tech. rep. NIST SP 800-204, Gaithersburg, MD: National Institute of Standards and Technology.
- Dimou, A. and Syropoulos, A. (2021), “Digital Documents”, *Encyclopedia of Information Science and Technology, Fifth Edition*, URL: <https://api.semanticscholar.org/CorpusID:242801232>.
- Dingsøyr, T., Dybå, T., and Moe, N. B. (2010), “Agile Software Development - Current Research and Future Directions”, URL: <https://api.semanticscholar.org/CorpusID:2824890>.
- Eken, S., Menhour, H., and Köksal, K. (2019), “DoCA: A Content-Based Automatic Classification System Over Digital Documents”, *IEEE Access* , Vol. 7, pp. 97996–98004, URL: <https://api.semanticscholar.org/CorpusID:199440728>.
- Ewin, M. (2024), *Agile Development stages*, Website, Accessed: 14 August 2024, URL: <https://www.pngwing.com/en/free-png-cbfck>.
- Felici, M. (2010), “Software Design and Class Diagrams”, URL: <https://api.semanticscholar.org/CorpusID:60048457>.
- Gamido, M. V., Gamido, H. V., and Macaspac, D. J. P. (2023), “Electronic document management system for local area network-based organizations”, *Indonesian Journal of Electrical Engineering and Computer Science*, URL: <https://api.semanticscholar.org/CorpusID:259497647>.
- Garfinkel, S. and Lipford, H. R. (2014), “Usable security: History, themes, and challenges”, *Synthesis Lectures on Information Security, Privacy, and Trust* , Vol. 5, No. 2, pp. 1–124.
- Garg, V. and Carsten, R. (2016), “Hybrid Cryptosystem for Secure Data Transmission”, *International Journal of Computer Applications* , Vol. 153, No. 4, pp. 1–4.

- Goyal, V., Malik, A., and Aggarwal, D. (2021), “Personal Cloud Storage Security: The Next Frontier”, *Cloud Computing Security*, Springer, pp. 175–194.
- Gueron, S. (2010), *Intel® Advanced Encryption Standard (AES) New Instructions Set*, tech. rep., Intel Corporation.
- Gupta, V. K. and Shoeb, M. (2019), “Key Hiding of Multimedia Applications in Multimedia File”, *International Journal of communication and computer Technologies*, URL: <https://api.semanticscholar.org/CorpusID:62360672>.
- Henderson, S. (2010), “Personal document management strategies”, *Proceedings of the 10th International Conference NZ Chapter of the ACM’s Special Interest Group on Human-Computer Interaction*, pp. 69–76.
- Johnson, A., Williams, B., and Thompson, C. (2023), *Digital Document Security: Challenges and Solutions*, New York: CyberPress.
- Kaliski, B. (2021), *The Mathematics of the RSA Public-Key Cryptosystem*, tech. rep., RSA Laboratories.
- Katz, J. and Lindell, Y. (2014), *Introduction to modern cryptography*, CRC press.
- Khan, M. E., Shadab, S., and Khan, F. (2020), “Empirical Study of Software Development Life Cycle and its Various Models”, URL: <https://api.semanticscholar.org/CorpusID:229377654>.
- Kocher, K. and Paul, D. (2011), “Introduction to differential power analysis”, *Journal of Cryptographic Engineering*, Vol. 1, No. 1, pp. 5–27.
- Kumar, A. and Pooja, K. (2010), “Steganography- A Data Hiding Technique”, *International Journal of Computer Applications*, Vol. 9, pp. 19–23, URL: <https://api.semanticscholar.org/CorpusID:15028787>.
- Lai, J.-F. and Heng, S.-H. (2022), “Secure File Storage On Cloud Using Hybrid Cryptography”, *Journal of Informatics and Web Engineering*, Vol. 14, No. 6, pp. 3–18.
- Lee, M. and Wang, L. (2024), “The impact of disorganized digital documents on workflow efficiency”, *Proceedings of the International Conference on Information Systems*, Singapore, pp. 156–170.
- Li, B., He, J., Huang, J., and Shi, Y. Q. (2011), “A survey on image steganography and steganalysis”, *Journal of Information Hiding and Multimedia Signal Processing*, Vol. 2, No. 2, pp. 142–172.

- Lou, D.-C. and Liu, J.-L. (2017), *Steganography and Digital Watermarking*, Nova Science Publishers.
- Manchanda, H., Agarwal, A., Bhati, D. K., and Ilango, P. (2017), “Agile Methods for Software Development”, *Imperial journal of interdisciplinary research* , Vol. 3, URL: <https://api.semanticscholar.org/CorpusID:116015873>.
- McCall, J. (2024), *Three Tier Architecture*, Website, Accessed: 14 August 2024, URL: <https://hyperskill.org/learn/step/25083>.
- Mishra, N. and Levkowitz, H. (2021), “PDV: Permissioned Blockchain based Personal Data Vault using Predictive Prefetching”, *Proceedings of the 2021 3rd Blockchain and Internet of Things Conference*, pp. 59–69.
- Muhammad, K., Ahmad, J., Farman, H., Jan, Z., Sajjad, M., and Baik, S. W. (2018), “A secure method for color image steganography using gray-level modification and multi-level encryption”, *IEICE Transactions on Information and Systems* , Vol. 101, No. 5, pp. 1337–1345.
- Nađ, I. (2014), “DES i AES”, URL: <https://api.semanticscholar.org/CorpusID:164635700>.
- Neha, M. K. (2016), “Enhanced security using hybrid encryption algorithm”, *International Journal of Innovative Research in Computer and Communication Engineering* , Vol. 4, No. 7, pp. 13001–13007.
- Poduval, A., Doke, A., Nemade, H., and Nikam, R. (2019), “Secure file storage on cloud using hybrid cryptography”, *International Journal of Computer Science and Engineering* , Vol. 7, No. 01, pp. 587–591.
- Rao, U. H. and Nayak, U. (2014), *The InfoSec handbook: an introduction to information security*, Apress.
- Raphael, A. J. and Sundaram, V (2011), “Cryptography and steganography–A survey”, *International Journal of Computer Technology and Applications* , Vol. 2, No. 3, pp. 626–630.
- Rescorla, E. (2018), *The Transport Layer Security (TLS) Protocol Version 1.3*, tech. rep. RFC 8446.
- Seggern, D. von, Posselt, K., Hassan, T., and Zellmann, T. (2019), “More than just digital paper-hidden features of the PDF format”, *Proceedings of the ACM Symposium on*

- Document Engineering 2019*, URL: <https://api.semanticscholar.org/CorpusID:202728546>.
- Sehgal, N. and Goel, D. A. (2014), “Evolution in Image Steganography”, URL: <https://api.semanticscholar.org/CorpusID:18578349>.
- Selent, D. (2010), “ADVANCED ENCRYPTION STANDARD”, URL: <https://api.semanticscholar.org/CorpusID:61226717>.
- Selvanayagam, J., Singh, A., Michael, J., and Jeswani, J. (2018), “Secure file storage on cloud using cryptography”, *Int. Res. J. Eng. Technol* , Vol. 5, No. 3, p. 2044.
- Sharma, M. K. (2017), “A study of SDLC to develop well engineered software”, *International Journal of Advanced Research in Computer Science* , Vol. 8, pp. 520–523, URL: <https://api.semanticscholar.org/CorpusID:64742110>.
- Sharma, V., Chauhan, A., Saxena, H., Mishra, S., and Bansal, S. (2021), “Secure file storage on cloud using hybrid cryptography”, *2021 5th International Conference on Information Systems and Computer Networks (ISCON)*, IEEE, pp. 1–6.
- Shastri, A. and Sharma, P. (2016), “Data vault: A security model for preventing data theft in corporate”, *Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies*, pp. 1–5.
- Shetty, Y., Sander, P., Raugh, A., and Gadiyar, M. (2023), “Software Development Life Cycle (SDLC) in Software Engineering – A Brief Review”, *Journal of Computer Science and System Software*, URL: <https://api.semanticscholar.org/CorpusID:260211585>.
- Smith, J. (2022), “The proliferation of digital documents in modern sectors”, *Journal of Information Management* , Vol. 45, No. 3, pp. 215–230.
- Stallings, W. (2017), *Cryptography and Network Security: Principles and Practice*, 7th ed., Pearson.
- Subhan, Z., Bhatti, A. T., and Pakistan, L. W. (2015), “Requirements Analysis and Design in the Context of Various Software Development Approaches”, URL: <https://api.semanticscholar.org/CorpusID:114493425>.
- Tyagi, A., Singh, R. V., and Sharma, S. (2020), “Data Hiding Techniques Using Steganography Algorithms”, URL: <https://api.semanticscholar.org/CorpusID:235868923>.

- Waja, G., Shah, J., and Nanavati, P. (2021), “AGILE SOFTWARE DEVELOPMENT”, URL:
<https://api.semanticscholar.org/CorpusID:239738666>.
- Zielińska, E., Mazurczyk, W., and Szczypiorski, K. (2014), “Trends in steganography”,
Communications of the ACM , Vol. 57, No. 3, pp. 86–95.