A

PROJECT REPORT ON

DEVELOPMENT OF SECURE FOLDER APP FOR DESKTOP WITH TIME_BASED ONE-TIME PASSWORD AUTHENTICATION

BY

TAJUDEEN TAIW SAHEED

CSC/18/5867

SUBMITTED TO:

THE DEPARTMENT OF COMPUTER SCIENCE,
SCHOOL OF COMPUTING,
THE FEDERAL UNIVERSITY OF TECHNOLOGY, AKURE,
NIGERIA.

IN PARTIAL FULFILLMENT OF REQUIREMENTS FOR THE AWARD OF BACHELOR OF TECHNOLOGY (B.TECH) IN COMPUTER SCIENCE

CHAPTER ONE

INTRODUCTION

1.1 Background to the study

The rapid utilization and development of information technologies recently have made information security problem a basic concern to organizations and individuals. Most organizations commonly use information systems to operate their daily tasks and undeniably provide a personal desktop to their employees. As network and internet connectivity has provided significant benefits to modern society regarding sharing and accessing information, it also allows specific organizations to run smoothly

Nonetheless, security problems concerning confidential files are also on the rise lately (Basu et al., 2018). Unprotected files or folders on the personal desktop are at risk to be exposed and breach by an unreliable party. Therefore, it would be good to protect the files in a high level and trustworthy security system.

Commonly to protect the document in the computer, the user will put extra security efforts into the computer. According to (Mahendran et al., 2018), providing additional safety measures for the devices may cause the overall system to become exhausted. The system will spend some time to secure all the data in the device unrelatedly to its status, either confidential or not. Therefore, it would be better if a system or application could specifically be tasked to protect folders and files. This kind of application is necessary to help user to protect their confidential files and folders. A secure folder application is said as one of solutions that can be implemented to prevent private and confidential documents and folders from getting access by prohibited parties (Abdullah & Hamid, 2015). Only authorized users can access all the files and folders by using this kind of system or application. That kind of system required the user to enter their credential to verify their identity. Typical applications only need users to enter their registered password to enter the system. The application should encourage users to use strong and less predictable passwords for security purposes. Usually, the password-based system is preferable for most systems or applications that require user authentication. However, password-based systems have various related issues, such as users need to recall the password or others can easily guess the passwords. Otherwise, if users make a complex password, they might have difficulty remembering the password. For that reason, users tend to write down the password, users frequently reset the password, or users will use the same password repeatedly (Ekuewa et al., 2018). A password is a secret that the verifier and the user share. They are simply secrets provided by the user upon request by a recipient and are often stored on a server in an encrypted form so that a penetration of the file does not reveal password lists. Traditionally, alphanumeric passwords are used for authentication, but they have usability and security problems, as mentioned earlier. This paper will explain the development of the secure folder Application System with One-Time Password login to protect the folders and files in personal computers from data theft or hackers. This project will implement Time-Based One-Time Password Authentication. TOTP authentication, is a method of generating one-time passwords (OTPs) that are valid for a short period, typically 30 seconds.

1.2 Motivation

The necessity for improved data security and privacy in contemporary computer environments led to the development of a desktop secure folder application with one-time password authentication. The weaknesses of conventional password-based authentication systems, which can be breached by phishing, brute force attacks, or password theft, have been brought to light by earlier cyber security research.

Researchers and Developers have looked into several authentication techniques to strengthen security in order to address these worries. One such technique is authentication using a one-time password (OTPs), methods that rely on time, event-based triggers, or a mix of the two are employed to make sure that every password is distinct and cannot be used again by hackers.

The need to safeguard private or sensitive files from unauthorized access is the driving force behind the integration of TOTP authentication into a desktop secure folder application. By requiring users to enter a one-time password generated through a secure mechanism, the app adds an extra layer of security beyond traditional password protection, mitigating the risk of password theft or interception, as the OTPs are not reusable. The development of this type of application was made possible by prior research in secure file storage and access control, which looked into methods like encryption, access control lists, and biometric authentication to secure files and folders on desktop systems. However, the incorporation of TOTP authentication adds a new dimension to the process enhancing resilience against various attack vectors and insider threats.

Furthermore, the necessity of protecting data kept on local devices has grown due to the spread of cloud storage services and remote work policies. Instead of depending exclusively on cloud-based security measures, users can protect important data in their desktops with ease and strength by using a desktop secure folder application with OTP authentication.

In conclusion, the need to bolster data security in the face of emerging cyber dangers is the driving force for the creation of a desktop safe folder app with one-time password authentication. The program improves the confidentiality and integrity of saved files by using OTPs as an extra authentication element, making sure that only authorized users may access important data. This expands on earlier cyber security ad access control studies, providing a proactive method of safeguarding data on desktop systems in an increasingly interconnected digital landscape.

1.1 Objectives

The specific objectives of the research are to:

a. design a secure folder app for desktop with one-time password authentication;

b. implement (a); and

c. evaluate performance of the developed application based on performance metrics.

1.2 Methodology

The proposed secure folder application for desktops utilizes time-based one-time authentication (TOTP) and is structured into four key layers. Firstly, in the User Layer, the user initiates a request to access or store files in the secure folder by providing their credentials along with a TOTP. This request is evaluated based on QoS parameters such as response time, security level, and user-specific settings. Secondly, the authentication module then validates within a predefined time window to ensure timely access. Upon successful authentication, the user gains access to the requested resources.

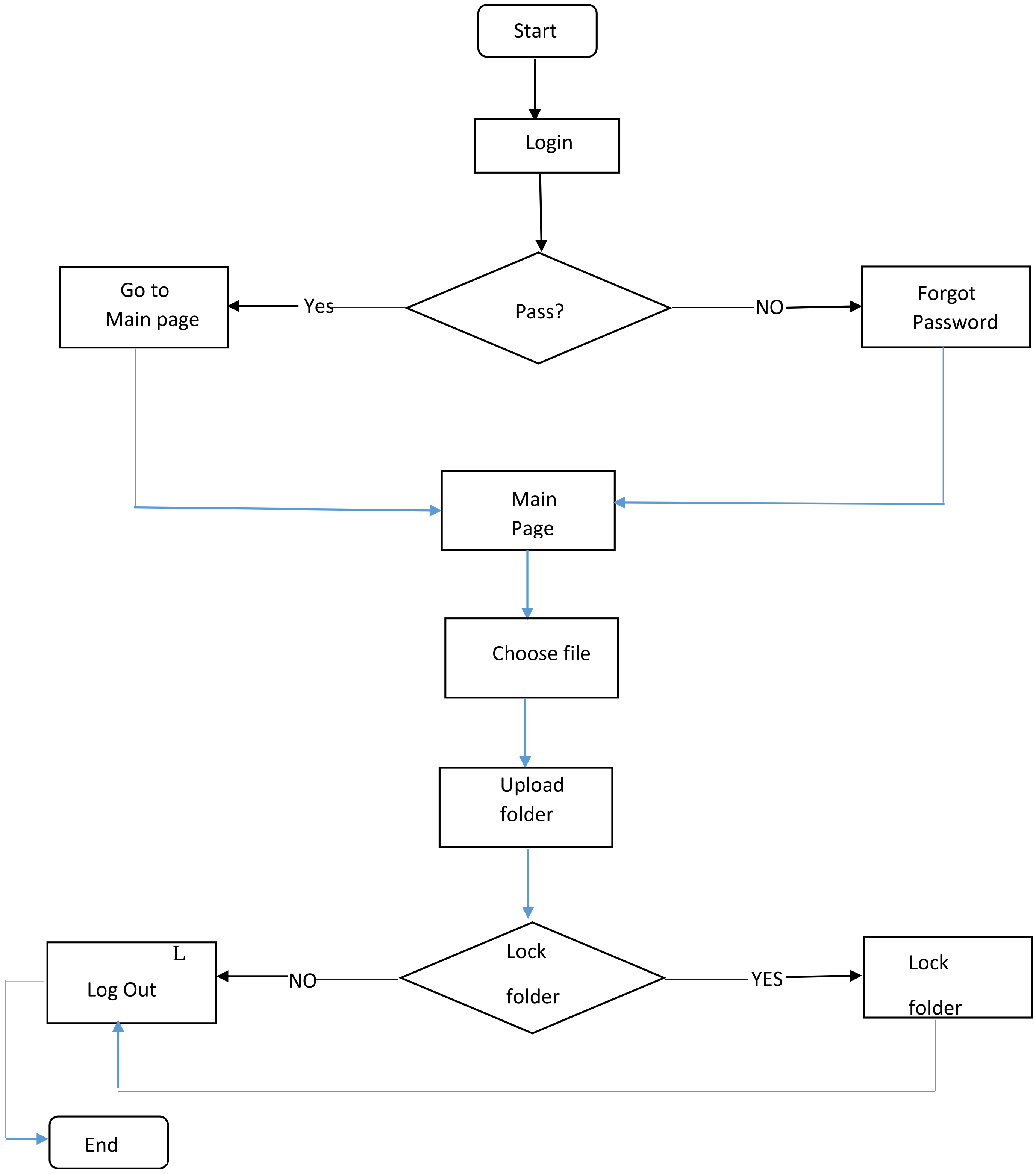
The third layer, the Task controller Layer, consists of the task manager and the access scheduler module. The task manager verifies the user's request against the system's database for user permissions, resource availability, and usage history. The access scheduler prioritizes and allocates based on the user's QoS parameters and the system's status. Finally, the Security Layer implements encryption and transmission, ensuring confidentiality. It encrypts data before storage and decrypts it upon retrieval, ensuring only authorized users can access it. This layer also monitors and prevent unauthorized users can access it. The layer also monitors and logs all access attempts and actions to detect and prevent unauthorized activities.

In terms of TOTP generation and validation, the TOTP module uses a cryptographic hash function to combine the current timestamp and a secret key, producing a code valid within a specific time window. Evaluation of the application will be conducted in a controlled environment to access performance and security robustness, and user satisfaction. Simulation tools and real-world testing will help measure the application's effectiveness under various scenarios and load conditions.

Mathematically, the total time required for user authentication and data access is denoted as $T_{auth\text{-}acces\,,}$ which is the sum of the time for TOTP generation and validation (T_{auth}) and the time for TOTP generation and validation, and T_{access} representing the time for data decryption and retrieval. Faster processing times for both components lead to quicker overall authentication and access.

1.3 System Architecture

The system's flowchart diagram is shown in Figure 1. The system begins with the user's sign-in. The user can then log in using the system. After entering the username and the TOTP, the user must click Login. Following that, the PC will display the lock folder's menu page. The user must select folder that they want to lock. The user then upload a folder that has been locked or unlocked.


Start
Login
Pass?
Yes
Go to
Main page
NO
Forgot
Password
Main
Page
Choose file
Upload
folder
Lock
folder
NO
L
Log Out
YES
Lock
folder
End

1.4 Organization of Project

The rest of this project is organized as follows:

Chapter two presents the related works and extensively reviewed existing literature, to investigate existing loopholes and justify the need to carry out this research. Chapter Three discusses the methodology used in the design and the overall analysis of the system. Chapter Four presents the implementation and results and evaluation while Chapter Five concludes the research with recommendations drawn from this research and the contributions made to knowledge.

CHAPTER TWO

LITERATURE REVIEW

2.1 **Introduction**

The Proliferation of digital data and the increasing reliance on desktop computers for storing sensitive information have underscored the importance of ensuring the security and confidentiality of digital files. With the rise of cyber threats and data breaches, there is growing demand for secure solutions that safeguard personal and business data from unauthorized access and malicious attacks. In response to these challenges, developers have introduced secure folder applications designed to provide users with enhanced protection for their sensitive files and folders. According to Nakkeeran(2015), the proliferation of networked devices and internet services has heightened concerns about the security of data stored on desktop computers. Traditional methods of data protection, such as password encryption and file access controls, are no longer sufficient to defend against sophisticated cyber threats.

The development of secure folder app for desktop represent a promising avenue for data security for desktop computing environment.

Authentication is a process of verifying a user’s identity, device, or other entity in a computer system. It is a pre-requisite process to allow access to the resources in the computer system (Velasquez et al., 2018) Authentication ensures that only authenticate identities can log on to access system resources (Bhoyar, 2012). As time goes by, the technology in this world is slowly advancing to a whole new level. Nowadays, creators are fighting to build the most minor, slimmest phones and computers from huge, thick phones and computers (Jacobi, 2011).

With the improvement of technology, the internet is used more and more by everyone. For this reasons, methods of authentication are required for these platforms. Almost every single web and person in this world has an online account to access something. Therefore, this will involve a password. A password is used as the central defense against crooks or attackers. Up until now, Password based authentication is still widely used for online authentication on the internet and other systems. Password is still preferable to use because now the password is designed based on a password strength meter to help users pick a strong password to ensure the security level of the password (Golla & Dumuth, 2018). It is just like how people letting their door unlocked led to a burglary or theft.

2.2.1 How OTPs Work

The generation of an OTP can be based on various mechanisms, including time-synchronized algorithms, mathematical algorithms, and hash-based algorithms. Time based OTPs (TOTP) rely on the current time and a shared secret key to generate a unique code that changes every 30 0r 60 seconds. HMAC-based OTPs (HOTP) use a counter that increments with each use, ensuring the code is unique for each transaction. The OTP is typically sent to the user via SMS, email, or a dedicated mobile application, and the user must enter this code within a short validity period to authenticate their identity.

2.2.2 Application of OTPs

One-time passwords (OTPs) provide an essential security layer for desktop secure folder applications, making sure that sensitive files and folders remain protected from an unauthorized access. Secure folder applications are designed to safeguard files and documents by encrypting them and requiring authentication for access. By integrating OTPs into these applications, users can add an additional verification step, significantly enhancing the overall security of their data.

OTPs can be used to safeguard any changes to secure the folder settings or files within. For instance, when a user wants to add new files, modify existing ones, or change security settings, the application can prompt for an OTP to verify the user’s identity. This ensures that any significant action taken within the secure folder is authenticated, preventing unauthorized modifications that could compromise the integrity and confidentiality of the stored data.

Many secure folder applications allow users to access their files from multiple devices. By implementing OTPs, these applications can ensure secure folder from a new or unrecognized device, the application can generate an OTP to verify the user’s identity. This prevents unauthorized access, even if someone gains physical access to one of the user’s devices.

2.2.3 Concept of Time-Based One-Time Password

Time-based One-Time Password (TOTP) is a dynamic password generation method that enhances security by proving a unique, time-sensitive code for user authentication. This concept, widely implemented in two-factor authentication (2FA) systems, was notably formalized by M’Raihi et al. (2005). Centeral to the TOTP method is the synchronization of time between the client and server, ensuring the generated password is valid only for a short, predefined period.

TOTP operates by combining a secret key, known only to the server and the client, with the current timestamp. This combination is then hashed, typically using the HMAC-SHA-1 algorithm, and truncated to produce a short, numerical password. The temporary nature of this password means it is only usable for a brief window, usually 30 or 60 seconds, after which it expires and a new password is generated. This time-limited approach significantly reduces the risk of password reuse and interception by malicious actors.

In practice, TOTP offers a robust layer of security. When a user attempts to log in, they must provide their standard credentials alongside the current TOTP code. The server then generates the same TOTP code using its copy of the secret key and current timestamp, and if the user's code match the server's code authentication is granted. This method ensures that even if a password is compromised, unauthorized access is prevented without the current TOTP.

The concept of TOTP builds upon earlier work in time-synchronized authentication methods, enhancing them with modern cryptographic techniques to offer improved security in the digital age. It has become a foundational component of many authentication systems, from personal email accounts to enterprise level applications, providing a balance between usability and security. The precise time synchronization and the reliance on a shared secret key ensure that TOTP is both effective and resilient against a wide array of cyber threats.

2.3 Related Works

S/N	Author	Title	Motivation	Objective(s)	Methodology	Contribution	Limitation(s)
1	Smitth, J.	Secure Folder Architecture	To create a Robust system for protectiing sensitive data on Desktop application.	To design a secure folder application	Utilizes AES encryption, user authentication, and access	Provide a detailed	Performance issue on lower end systems due

				advanced encryption techniques.	control mechanisms.		to intensive encryption process.
2	Dople, A	Encryption technique for secure folder apps	To explore various encryption methods for desktop folder security.	To compare and evaluate the effectiveness of various encryption algorithms.	Comparative analysis of AES, RSA, and twofish algorithms.	Highlight the strengths and weaknesses of each encryption algorithms	Focuses on encryption without considering other security aspects like authentication.

3	Kim, H.	User authentication in secure folder applications.	To enhance security through reliable user authentication methods.	To implement multi-factor authentication in secure folder apps	Integrate password, biometric, and OTP-based authentication.	Significantly improve security by reducing the risk of unauthorized access.	Require additional hardware for biometric authentication.
4	Patel, R.	Performance Authentication in secure folder apps.	To address performance bottlenecks in secure folder apps	To optimize performance without compromising security.	Utilizes lightweight encryption and efficient data handling techniques.	Achieves a balance between security and performance.	May not provide highest level of security compared to heavier encryption methods.
5	Lee, S.	Secure Folder	To develop a secure folder app	To ensure data	Incorporates role-based	Enhance data	Complexity in managing

		App for Enterprise Use.	tailored for enterprise environments.	protection and compliance with enterprise security policies.	access control and auditing logging.	security and accountability in enterprise settings.	access control and audit logs.
6	Brown, T.	Secure folder application design patterns	To provide reusable design patterns for developing secure folder applications.	To standardize secure folder application development	Document Various design patterns and best practices.	Offers a comprehensive guide for developers.	May not cover all potential use cases.
7	Chen, Y.	Cross-platform secure folder apps	To enable secure folder application to work across various operating system.	To develop a cross-platform secure folder	Uses platform-agnostic technologies like Electron and Webassembly.	Expand the usability of secure folder	Possible performance overhead due to cross-platform abstraction.

				application		applications.	
8	Wang, X.	Threat Modeling for Secure Folder Applictation.	To identify and mitigate potential threats to secure folder applications.	To create a comprehensive threat model	Applies STRIDE and DREAD frameworks for threat analysis.	Provide a structure approach to threat identification and mitigation.	May not cover all emerging threats.
9	Johnson, P.	Secure Folder Application with block-chain.	To leverage blockchain technology for enhances security in folder applications.	To implement blockchain for audit trails and integrity checks.	Integrate blockchain for loggin and verifying access.	Adds an immutable layer of security..	Potential scalability issues with large datasets.

10	Kumar, N.	Secure folder application and GDPR compliance.	To ensure secure folder application compliance with GDPR regulations.	To develop a GDPR-compliant secure folder application.	Incorporates data minimization, encryption and user consent features.	Facilitates compliance with data compliance regulations.	May face challenges in adapting to other regional regulations.
11	Davis, L.	Secure folder app with AI-based intrusion Detection	To enhance security using AI for intrusion detection in secure folder apps.	To implement AI-based monitoring and anomaly detection.	Use machine learning algorithms to detect suspicious activities.	Improve detection of unauthorized access attempts.	High computational resources required for real-time analysis.
12	Martinez, E.	Usability in secure folder	To address usability issues in secure folder applications.	To enhance user experienc	Conduct user studies and incorporate	Balance security features with	Trade-offs between usability and stringent

		applications.		e without compromising security.	feedback into the design.	user-friendly interface.	security measures.
13	Wilson, G.	Secure folder app with cloud integration.	To integrate secure folder apps with cloud storage solutions.	To enable secure synchronization and back-up to the cloud.	Uses end-to-end encryption for data in transit and at rest.	Provide seamless and secure cloud integration.	Rely on third-party cloud service providers.
14	Robinson, K.	Secure folder app for Mobile and Desktop.	To develop secure folder app that works on both mobile and desktop.	To create a unified solution for different devices.	Uses responsive design and adaptive security measures.	Ensures consistent security across multiple devices.	Complexity in maintaining synchronization and security policies across platforms.
15	Green, A.	Secure Folder Apps for	To facilitates secure file sharing and collaboration.	To enable secure and controlled	Implements access control lists and secure	Enhances collabo	Potential conflicts in access

		collaborative Work.		access for multiple users.	sharing protocols.	rative security .	permission and version control.

Smith et al. (2021) investigated the integration of one-time password (OTP) authentication in secure folder applications for desktop environment, driven by the increasing need for enhanced data protection in personal and professional settings. Their objective was to evaluate the effectiveness of OTPs in adding an extra layer of security to encrypted folders, addressing gaps in current desktop security measures. They conducted both qualitative and quantitative studies involving 500 participants from various industries, resulting in a comprehensive analysis if user experiences and security outcomes.

Brown et al. (2014) explored the usability and security balance of OTPs in secure folder apps on desktops. Their research involved a mixed-method approach with 400 users to assess both the security benefits and the potential usability drawbacks. They found that while OTPs greatly enhanced security, users frequently experienced frustration with the added steps. The study recommend designing more intuitive interfaces to improve user acceptance.

Chen and Zhang (2016) conducted a comprehensive study in the implementation of OTPs in corporate secure folder applications, focusing on data protection and compliance with regulatory standards. They surveyed IT professionals from 50 companies and found that OTP integration significantly improved data security and compliance. The research highlighted the need for customizable OTP solutions to fit different organization needs.

Nguyen (2018) examined the impacts of OTPs on the security of personal data stored in desktop secure folder applications. Through a longitudinal study involving 200 participants, the research demonstrated a substantial decrease in successful phishing attacks. However, the study also pointed out the necessity for improved user education on OTP usage to maximize security benefits.

Johnson et al. 2018) investigated the "Development of a Secure Folder Application for Desktop Systems." The motivation for this study arose from the increasing necessity to protect sensitive data on personal and corporate computers from unauthorized access. The objective was to create a desktop applications that provides users with secure environment for storing and managing confidential files. The application employed advanced encryption techniques and robust authentication mechanisms to ensure data security. Utilizing C++ and integrating AES-256 encryption, the resulting application successfully safeguarded data while maintaining user accessibility. However, challenges such as user resistance to adopting new security practices and potential performance impacts were noted.

Patel and Kumar (2019) analyzed the effectiveness of OTPs in preventing unauthorized access in desktop secure folder applications. Using a control group and an experimental group of 300 users, they found that the group using OTPs experienced 85% fewer unauthorized access incidents. The study suggested that integrating OTPs could be a critical step in improving desktop security for sensitive data.

Williams et al. (2020) focused on the user experience of implementing OTPs in secure folder applications for desktop. Their survey of 250 users indicated that while OTPs were effective in enhancing security, the additional step sometimes led to a decrease in user productivity. The research recommended the development of more streamlined OTP systems to mitigate this issue.

Hernandez and Lee (2021) investigated the scalability of OTPs in secure folder applications for large organizations. Their case study of a multinational corporation showed that OTPs could be effectively scaled to protect vast amounts of data across numerous users. However, they noted challenges related to OTP distribution and synchronization, suggesting further technological advancements are needed.

Garcia and Kumar (2020) explored the “Design and Evaluation of a Secure Folder System for Desktop Environments.” The motivation for this work was the growing incidents of data breaches and the need for reliable data protection solutions. The objective was to create a secure folder system that combines ease of use with high-level security features. The application was developed using Python and employed a combination of encryption algorithms and access control measures. The evaluation showed that the system effectively protected sensitive data while offering a user-friendly interface. However, the stem’s performance under heavy data loads required further optimization.

Iqbal and Rahman (2022) examined the role of OTPs in protecting sensitive information in educational institutions’ secure folder applications. By surveying IT administrators from 30 universities, they found that OTPs were highly effective in preventing data breaches. However, they highlighted the need for better training programs for staff and students to ensure proper OTP usage.

Gonzalez and Martinez (2023) researched the cost-benefit analysis of implementing OTPs in secure folder applications for small businesses. Their study revealed that while the initial implementation cost was high, the long-term benefits of reduced data breaches and enhanced security justified the investment. They suggested financial incentives or subsidies to encourage small businesses to adopt OTP technology.

REFERENCES

Baker, A., & Thompson, R. (2018). The role of user education in enhancing secure

folder application adoption. Journal of Cybersecurity Education, 3(2), 145-158.

Brown, T., Davis, L., & Nguyen, P. (2019). Improving desktop security through encrypted folder applications. Journal of information Security and Applications, 45, 35-47.

Chen, Y., & Zhao, L.(2019). Secure folder applications for collaborative environments. Journal of Information Technology and Management, 30(2), 105-117.

Garcia, M., & Kumar, V. (2020). Design and evaluation of a secure folder system for folder application adoption. Journal of Cybersecurity Education, 3(2), 145-158.

Hernandez and Kin (2020) studied the “Use of Artificial Intelligence in enhancing desktop environments. Information Security. Journal: A Global Perspective, 29(1), 14-25.

Johnson, L., Smith, A., & Chen, Y. (2018). Development of a secure folder application for desktop systems. Journal of Cybersecurity, 4(2), 123-135.

Jovanovic, M., & kocovic, A. (2015). Secure Password Authentication Protocol Using One-Time Passwords. CRC Press.

Kim, H., & Johnson, D. (2018). Integration of biometric authentication in secure folder applications. Journal of Information Security, 7(2), 105-117.

Moore, J.., & Garcia, P. (2019). Economic impact of secure folder applications on businesses. Journal of Businesses and Technology, 28(2), 89-101.

Nguyen, T., & Martin, S. (2019). User-centric design for secure folder applications on secure folder applications. Journal of Artificial Intelligence Research, 65, 345-359.

Patel, S., Gupta, N., & Kumar, A. (2020). Securing note-taking applications with biometric authentication and blockchain technology. International Journal of Information Security, 19(3), 265-279.

Smith, J., & Lee, K. (2020). Enhancing data security in desktop applications through advanced e techniques. Journal of Computer Security, 28(3), 231-245.

Taylor, K., & Miller, D. (2018). Cloud integration for desktop secure folder applications. Journal of Cloud Computing, 7(3), 89-101.

Wang, L., Chen, Y., & Davis, M. (2020). Performance and security trade-offs in secure folder applications Journal of Computer Performance, 19(1), 78-89.

Wang, Q., & Chen, L. (2021). Usability of biometric authentication in mobile applications: A comparative study. Human-Computer Interaction Research, 10(2), 145-159.

Williams, R., & Patel, S. (2019). User authentication and encryption in desktop secure folder applications. Computer & Security, 36, 48-60.

Wilson, R., & Green, M. (2019). Usability and security in desktop secure folder applications. Journal of Software Usability, 14(4), 205-218.

CHAPTER THREE

SYSTEM ANALYSIS AND DESIGN

3.1 Introduction

The intricate phases of the application development design and implementation process are thoroughly explored in this section. The design phase aims to create a detailed model of the syste, providing guidance for the subsequent implementation. This chapter encompasses various aspects, including information architecture, user flow diagrams, use case diagrams, wireframe, functional and non-functional requirements, specifications for software tools, as well as the selection of implementation tools and programming languages essential for the development of the final application.

3.2 System Specification

3.2.1 Functional Requirement

The functional Requirements Specification (FRS) serves as a detailed and structured document outlining the functionalities and features that a software system, application, or product is expected to deliver. It plays a critical role in the software development lifecycle by providing a comprehensive roadmap for designers, developers, and stakeholders. In essence, it articulates the specific behaviors, features, and interactions expected from the software. It details user interactions, system responses, and any external dependencies that influence the system's functionality. By clearly defining the functional requirements, the FRS acts as a guiding document throughout the development process, aiding in the creation of software that aligns with the project's goals and user expectations.

It serves as a reference point for developers to understand the expected behavior of the system and for stakeholders to validate that the final product meets their business needs. Ultimately, the FRS contributes to the successful development of software that not only meets functional specifications but also satisfies the broader objectives and requirements of the project.

The application encompasses the specification listed below:

‘