An Approach to Storing and Sharing Digitally Signed Documents using IPFS

Pratham H. Sunnal
CSE Department,
BMSIT & M, Bengaluru

Prajodh Pragath Sunder
CSE Department,
BMSIT & M, Bengaluru

Pralhad P. Teggi
CSE Department,
BMSIT & M, Bengaluru

ABSTRACT

There is a long and hectic process involved in getting the documents verified by the higher-ups and the documents undergo a series of changes and modifications before getting the final approval. Our proposed system provides a secure way to exchange, verify, modify, and comment out the changes performed on the document. The process begins with the creation of a digital document by a user. The document is then encrypted and stored on the IPFS. The user can then share the document with other authorized parties by using their unique ID's. When a party receives the document, they can verify its authenticity and contents using the cryptographic hash of the document stored on the IPFS. If the document meets their requirements, they can then sign and encrypt it using their private key. The signed document is then stored on the IPFS and made available to other authorized parties.

General Terms

Document Sharing, Digital Signature, Algorithms ,IPFS, Face Recognition , Key Generation, Fernet.

Keywords

Digital Signature , IPFS , Algorithm , Elliptical curve digital signature algorithm.

1. INTRODUCTION

The digital exchange of documents has become prevalent in today's world, necessitating secure and efficient file sharing systems. However, conventional methods often suffer from security vulnerabilities, slow speeds, and reliance on personal information. The Interplanetary File System (IPFS) offers a decentralized and secure solution for digital document exchange. IPFS operates on a peer-to-peer network architecture, where files are encrypted, split into chunks, and distributed across multiple nodes. This decentralized approach ensures the security and integrity of files, even if one node is compromised.

By utilizing IPFS for digital document exchange, users can securely store and share documents without relying on centralized servers or third-party services. This enhances security, privacy, and user control over data. The concept of digital exchange and biometric signing using IPFS combines the security and decentralization of IPFS with the accuracy and authenticity of biometric signatures. Biometric signatures leverage unique physical characteristics like fingerprints, facial recognition, or voice recognition to create digital signatures tied to individuals. This provides a highly secure method of identity verification, as biometric features are difficult to replicate or forge. The integration of biometric signatures with IPFS enhances the security and verification of digital document exchange. By employing biometric signatures for document signing on IPFS, the identity of the signatory can be accurately and verifiably confirmed, mitigating fraud and fostering trust in digital transactions.

IPFS uses content-addressing, which means that each document is uniquely identified by its content, rather than its location on the network. This allows documents to be stored and shared more efficiently, since identical documents are only stored once, even if they are shared by multiple users. When a document is shared using IPFS, it is encrypted and broken down into smaller chunks, which are then distributed across the network.

Each chunk is assigned a unique identifier, which is used to retrieve the chunk from the network when the document is accessed. IPFS allows users to host and receive content. It is built around a decentralized system of user-operators who hold a portion of the overall data, creating a resilient system of file storage and sharing. Any user in the network can serve a file by its content address, and other peers in the network can find and request that content from any node who has it using a distributed hash table (DHT). IPFS also supports versioning, allowing multiple versions of a document to be stored and accessed. This can be useful for tracking changes to a document over time, or for keeping a record of previous versions of a document.

A biometric digital signature is a method of digitally signing a document using biometric authentication, such as facial recognition. This approach ensures that the signature is unique to the signer and cannot be replicated or forged by anyone else. To use biometric digital signatures for document signing, the signer must first create a digital signature template, which contains their biometric information. This can include facial features, voice recognition, or other unique biometric identifiers.

2. LITERATURE SURVEY

The objective of this literature survey is to explore the research and advancements in the field of biometric digital signing and document sharing, particularly focusing on the utilization of the Inter Planetary File System (IPFS) technology. Several research papers have been analyzed to understand the current state of the art, challenges, and potential solutions in this domain.

No	Paper Title	Publications	Technology/Tools used	Contributors
1	**A Dynamic Resource Management Scheme for ContentBased P2P Networks: A Case Study of IPFS [1].(2021)**	**Published in IEEE Access. Bo Fu, et al. Switzerland**	**Load-Balancing Caching Replication Adaptive resource allocation**	**The authors evaluate the performance of the scheme through simulations and experiments on a real IPFS network and demonstrate its effectiveness in**

				improving system scalability and reducing resource waste.
2	**Exchanging Digital Documents Using Blockchain Technology [2]. (2021)**	**Proceedings of the 3 rd International Conference on Electrical, Communication and Computer Engineering (ICECCE). Anas Abu Talab et al. Jordan**	**Developed a decentralized system for exchanging digital documents, using Ethereum blockchain, Interplanetary File System (IPFS) Smart contracts and AES algorithm to overcome the problems of using trusted third parties in exchanging documents.**	**The model has satisfied the three main security. requirements which are confidentiality, integrity, and availability. The model was affordable (costed 0.0552$ per transaction**
3	**Blockchain based Framework for Student Identity and Educational Certificate Verification [4]. (2021)**	**Proceedings of the 2 nd Int. Conference on Electronics and Sustainable Communication Systems (ICESC-2021) Aastha Chowdhary et al Suratkal**	**Proceedings of the 2 nd Int. Conference on Electronics and Sustainable Communication Systems (ICESC-2021) Aastha Chowdhary et al Suratkal**	**They were able to link the certificates to the user Identity for more security and to develop a foul-proof system.**
4	**Software Development of Electronic Digital Signature Generation at Institution Electronic Document Circulation [7]. (2020)**	**IEEE East-West Design & Test Symposium (EWDTS) Olga A. Safaryan et al. Russia**	**Implemented in Python. Development environment used is Microsoft Visual Studio Code. The graphical interface of the software tool is implemented as web page using HTML, CSS and JS.Made use of Blockchain and PKI**	**The authors were the first to use blockchain technology for remote cloud generation and verification of electronic signatures. The elements were designed by keeping scalability and extensibility in mind**

2.1 Why IPFS ?

IPFS is a decentralized file system that enables the sharing and storage of files in a distributed manner, using content addressing to identify and retrieve files based on their content. It offers numerous benefits, including decentralized storage, content addressing, data integrity and authenticity, efficient content distribution, peer-to-peer communication, offline file access, versioning and deduplication, community and ecosystem support, cost effectiveness, and independence from third-party services. These advantages make IPFS an attractive option for secure and efficient document exchange and signature, as it eliminates the risks associated with centralized systems and offers enhanced privacy, scalability, security, and interoperability. IPFS provides a promising solution for various applications such as web hosting, data sharing, and distributed applications. In contrast, blockchain is a distributed ledger technology that enables secure and transparent transactions between parties without the need for a trusted third party, commonly used for applications such as cryptocurrency, supply chain management, and digital identity.

Reasons to consider:

1. Decentralized Storage: IPFS provides a decentralized storage solution, where files are distributed across multiple nodes in a peer-to-peer network. This decentralized approach eliminates the reliance on a central server, reducing the risk of single points of failure and enhancing data availability and resilience. In the context of document exchange, this ensures that documents are not stored in a single vulnerable location, enhancing security and reliability.

2. Content Addressing: IPFS uses content addressing, meaning files are identified by their content rather than their location. Each file is assigned a unique cryptographic hash based on its content, enabling efficient retrieval and verification. This content based addressing also ensures that files remain immutable, as any changes to the content will result in a different hash.

3. Data Integrity and Authenticity: Since IPFS is making use of content addressing (where files are identified by their content hash), this ensures that files remain immutable and tamper-proof. Any modifications to a document will result in a different hash, immediately indicating a mismatch. This feature ensures the integrity and authenticity of documents, which is crucial for digital signing and maintaining the trustworthiness of exchanged files.

3. SEQUENCE DIAGRAM

Fig .1 shows the sequence diagram that shows the interactions between objects or components in a system over time.

1. The user should register himself by providing his personal details such as email Id, username and password.

2. After providing the details he will be directed to the next page where he will register his face, which will be used to authenticate the user and the generate private key .

3. Once the private key is generated the user has to download the private key and keep it securely, as private is generated only once.

4. After registering the user can login and use all the services.

5. The user can upload the file and send to multiple users and in order to do so he has to provide his private key.

6. The file uploaded will be encrypted and will be uploaded to the ipf s server and the file hash (CID) of that file will be sent to the receiver.

7. The receiver can view the file file , download it and then sign the document if it is correct using his private key .

8. Once the document is signed it will be reflected to the sender, in the website and the sender can verify the authenticity of that file.

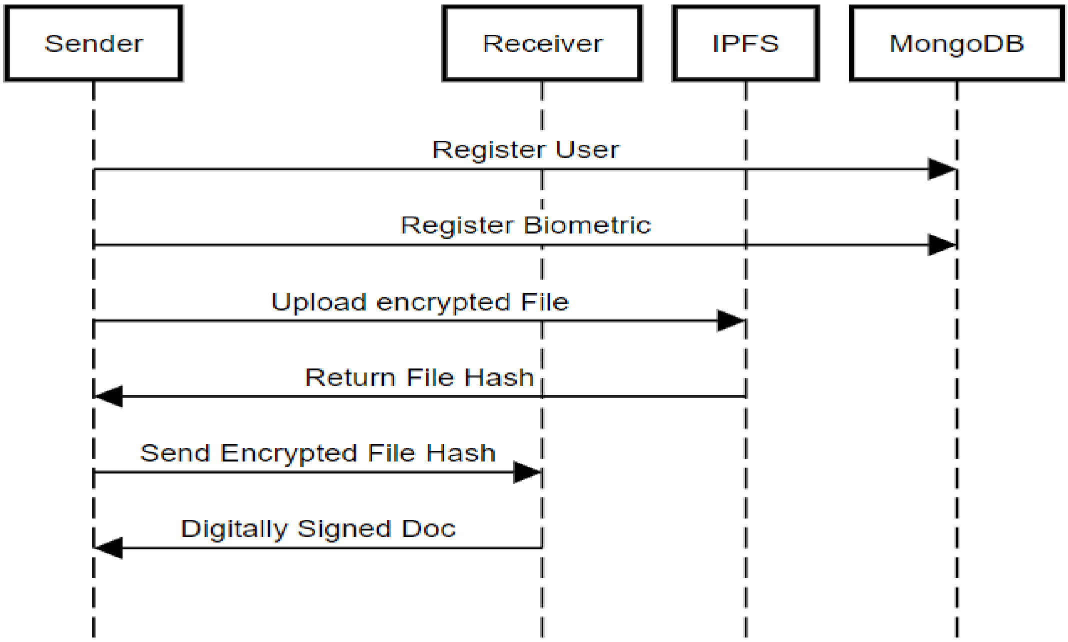

Fig 1: Sequence Diagram

4. PROPOSED SYSTEM

Flask is a Python web framework known for its lightweight and flexible nature. Created by Armin Ronacher in 2010, Flask enables developers to rapidly build web applications without the complexities typically associated with full-stack frameworks. One of the key features of Flask is its extensibility. It is built on top of the Werkzeug WSGI toolkit and the Jinja2 templating engine, allowing developers to easily incorporate thirdparty extensions for additional functionality. A wide range of extensions are available for Flask, covering areas such as database integration, authentication, testing, and more. Flask's minimalistic approach is another advantage. By focusing on core functionalities and avoiding unnecessary components, Flask offers developers greater control over their application's architecture. This aspect proves beneficial for small to medium-sized projects, where a lightweight framework like Flask can provide faster and more efficient performance compared to larger frameworks .In summary, Flask is a powerful and versatile Python web framework. Its minimalistic design, extensibility, and beginner-friendly nature contribute to its popularity among developers seeking a flexible and efficient solution for web application development.

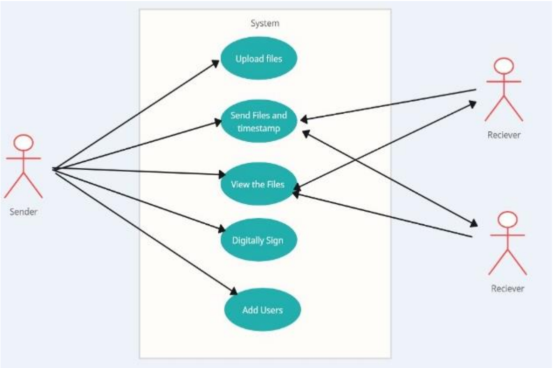

Fig 2: Use case Diagram

The above use case diagram in fig 2 shows how the users will interact with our project and its five main functionalities.

1. There are two main kinds of users :
   a. Those who send files
   b. Those who receive those files
2. The sender can view and perform all the activities but a receiver can only view files and react to that by either,
   a. Digitally signing those files, or
   b. By Adding comments in the box provided (in the website)

4.1 Algorithms

4.1.1 Fernet(AES)

Fernet is a widely used symmetric encryption algorithm that employs the AES (Advanced Encryption Standard) algorithm in CBC (Cipher Block Chaining) mode with PKCS7 padding. It is known for its simplicity, security, and efficiency in cryptography. Fernet is commonly utilized for securing message encryption and decryption, ensuring the confidentiality and integrity of data. The algorithm relies on a shared secret key that is used by both the sender and the receiver for encryption and decryption operations, guaranteeing that only authorized parties can access and decrypt the encrypted data. The decision to use Fernet (AES) is driven by its strong security features. AES is a trusted encryption algorithm that has been adopted as the standard for symmetric encryption by organizations like the U.S. National Institute of Standards and Technology (NIST). With AES employing a block cipher and offering various key sizes, it is highly resistant to brute-force attacks. Fernet enhances the security by utilizing AES in CBC mode, which introduces an additional layer of protection by leveraging the ciphertext of the previous block during encryption. This approach mitigates the risk of identical plaintext blocks producing identical ciphertext blocks, thereby bolstering security and preventing potential vulnerabilities.

4.1.1.1 Elliptic Curve Algorithm

The elliptic curve algorithm (ECA) is a widely used public-key cryptography algorithm that ensures secure communication over networks. It leverages the mathematical properties of elliptic curves and finds applications in digital signature schemes, key exchange protocols, and various cryptographic scenarios. ECA's core concept involves creating a public key that can be shared openly and a private key that remains confidential. The security of the system is based on the immense difficulty of deriving the private key from the public key, even with advanced computing capabilities. In the context of this project, the elliptic curve algorithm can be employed for secure key generation, key exchange, and digital signatures. Furthermore, the algorithm enables efficient and secure digital signatures, ensuring the integrity and authenticity of signed documents. To generate a public key, the owner selects a random integer as the private key and performs a multiplication operation with a base point on the curve. The resulting point becomes the public key. When encrypting a message, the sender employs the recipient's public key to derive a shared secret key for encryption. The recipient can then use their private key to derive the same shared secret key and decrypt the message. Signing a message involves using the sender's private key to generate a digital signature, which is attached to the message. The recipient can subsequently use the sender's public key to verify the signature and confirm the message's integrity.

4.1.1.2 ECDSA (Elliptic Curve Digital Signature Algorithm) Sign

PsuedoCode:

```
1. Calculate the message hash, using a cryptographic hash function like SHA-256: $h$ = hash($msg$)

2. Generate securely a random number $k$ in the range [1..$n$-1]

3. In case of deterministic-ECDSA, the value $k$ is HMAC-derived from $h$ + $privKey$

4. Calculate the random point $R = k \times G$ and take its x-coordinate: $r = R \times x$

5. Calculate the signature proof:

   $s = k^{-1}(h + r \times privKey)(mod\ n)$

   $s1 = s^{-1}(mod\ n)$

   Return the signature {$r$, $s$}.
```

The calculated **signature** {***r***, ***s***} is a pair of integers, each in the range [1...***n***-1]. It encodes the random point ***R*** = ***k*** * **G**, along with a proof ***s***, confirming that the signer knows the message ***h*** and the private key ***privKey***. The proof ***s*** is by idea verifiable using the corresponding ***pubKey***.

ECDSA signatures are **2 times longer** than the signer's **private key** for the curve used during the signing process. For example, for 256-bit elliptic curves (like secp256k1) the ECDSA signature is 512 bits (64 bytes) and for 521-bit curves (like secp521r1) the signature is 1042 bits.

4.1.1.3 ECDSA (Elliptic Curve Digital Signature Algorithm) Verify Signature

Psuedocode:

```
1. Calculate the message hash, with the same cryptographic hash function used during the signing: $h$ = hash($msg$)

2. Calculate the modular inverse of the signature proof:

   $s1 = s^{-1}(mod\ n)$

3. Recover the random point used during the signing:

   $R' = (h \times s1) \times G + (r \times s1)\ pubKey$

4. Take from $R'$ its x-coordinate: $r' = R'.x$

5. Calculate the signature validation result by comparing whether $r' == r$
```

The general idea of the signature verification is to **recover the point *R'*** using the public key and check whether it is same point ***R***, generated randomly during the signing process.

5. PROCEDURE & SCREENSHOTS

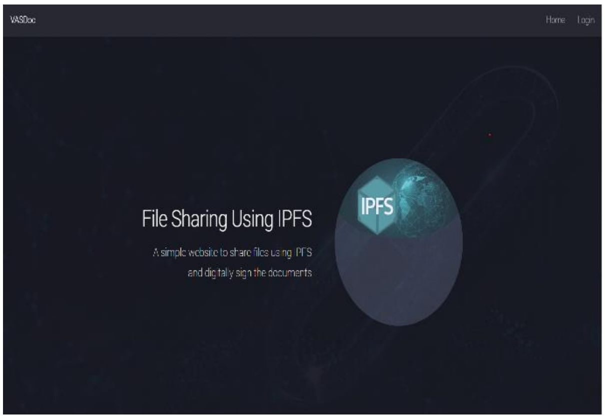
VASDoc
Home
Login
File Sharing Using IPFS
A simple website to share files using IPFS
and digitally sign the documents
IPFS

Fig 2: Screenshot of login page

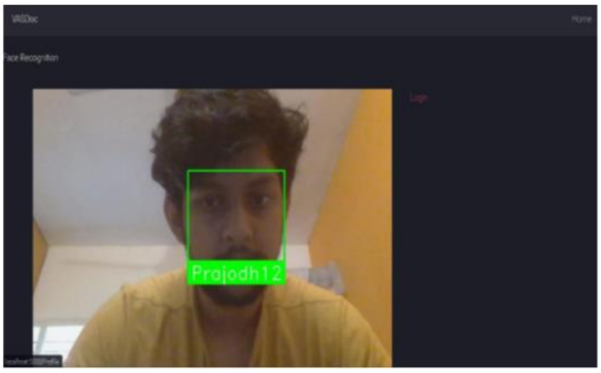

Fig 2: Screenshot of image capturing.

- The user first registers himself on the website (if he's accessing it for the first time) or logs in to the website (if he's already registered).
- He then records/registers all the recipients with whom he is sharing the file.
- He can add or remove the receivers by searching for their username.
- The file is then uploaded to the IPFS server by clicking on the upload button.
- The user can view his past uploads and the current status of the file, as well as check 20 the files he has received from other users.
- The receiver can access the uploaded file by using the CID (Content Identifier) sent by the sender.
- The receiver now has two ways to send the file:
- He can do a normal file transfer where the file is encrypted and sent back to the sender for further corrections and modifications.
- Or, he can digitally sign the document and send it back to the sender.
- The receiver can send the required corrections by using the comment box provided on the website.
- The process is complete only when the file has been digitally signed

6. CONCLUSION

As presented, the existing systems for exchanging digital documents are centralized systems, thus these systems are vulnerable to be unavailable for users due to many problems such as (single-point of failure, natural disasters, privacy issues etc..). Using a decentralized system such as IPFS can solve the problems of a single point of failure and natural disasters. This project has focused on the development of a system for biometric digital signing and exchange of documents using IPFS. The system utilizes facial recognition technology for biometric signing and IPFS for document exchange, ensuring secure and reliable transmission of sensitive documents. The use of elliptic curve cryptography ensures stronger security compared to traditional RSA encryption methods. Throughout the course of the project, we have explored the limitations of existing systems and the motivation for using IPFS and biometric authentication. We have also discussed the software requirements and specifications, as well as the challenges faced during the development process. Overall, the system has the potential to revolutionize the way documents are exchanged and signed, particularly in industries such as finance, healthcare, and legal services. It provides a secure and reliable platform for transmitting sensitive documents, while also enhancing the efficiency and speed of the signing process.

7. REFERENCES

[1] "#What Is Ipfs." IPFS Docs, n.d. https://docs.ipfs.tech/concepts/what-is-ipfs/.

[2] Krishnan, Armin (2020). "Blockchain Empowers Social Resistance and Terrorism Through Decentralized Autonomous Organizations". Journal of Strategic Security. 13 (1): 41–58. doi:10.5038/1944-0472.13.1.1743.

[3] Interplanetary file system (2023) Wikipedia. Wikimedia Foundation. Available at: https://en.wikipedia.org/wiki/InterPlanetary_File_System .

[4] "Reading Your Face: How Does Facial Recognition Work?" ARATEK, n.d. https://www.aratek.co/news/how-does-facial-recognition-work.

[5] Asraa Ahmed, Taha Hasan, Firas A. Abdullatif, Mustafa S. T, Mohd Shafry Mohd Rahim (2019). "A Digital Signature System Based on Real Time Face Recognition". IEEE 9th International Conference on System Engineering and Technology (ICSET 2019), 7 October 2019, Shah Alam, Malaysia. 2 (2): 17-24.doi: 10.1109/ICSEngT.2019.8906410.

[6] Bo Fu, et al. "A Dynamic Resource Management Scheme for Content-Based P2P Networks: A Case Study of IPFS". IEEE Access, 9:75515-75526, 2021.

[7] Yasmeen shaher Alslman and Anas Abu Taleb, "Exchanging Digital Documents Using Blockchain Technology", Jordan, 2021.

[8] Aastha Chowdhary, Shubham Agrawal and Dr. Bhawana Rudra, "Blockchain based Framework for Student Identity and Educational Certificate Verification", India, 2021.

[9] Nikita I. Chesnokov, Denis A. Korochentsev, Larissa V. Cherckesova, Olga A. Safaryan, Vladislav E. Chumakov, Irina A. Pilipenko, "Software Development of Electronic Digital Signature Generation at Institution Electronic Document Circulation", Russia, 2020.

[10] Anastacio Antolino Hernández, Juan Carlos Olivares Rojas, "Management of digital documents with encrypted signature, through the use of centralized PKI, and distributed using blockchain for a secure exchange", Mexico, 2019.

[11] R. P. Pasupulati and J. Shropshire, "Analysis of centralized and decentralized cloud architectures," SoutheastCon 2016, Norfolk, VA, USA, 2016, pp. 1-7, doi: 10.1109/SECON.2016.7506680.

[12] IvanOnTech. "Interplanetary File System Explained - What Is Ipfs?" Moralis Academy, December 16, 2021. https://academy.moralis.io/blog/interplanetary-filesystem-explained-what-is-ipfs.

[13] Doan, Trinh Viet, Vaibhav Bajpai, Yiannis Psaras, and Jörg Ott. "Towards decentralised cloud storage with IPFS: Opportunities, challenges, and future directions." arXiv preprint arXiv:2202.06315 (2022).

[14] Benet, Juan. "Ipfs-content addressed, versioned, p2p file system." arXiv preprint arXiv:1407.3561 (2014).

[15] IvanOnTech. "Interplanetary File System Explained - What Is Ipfs?" Moralis Academy, December 16, 2021. https://academy.moralis.io/blog/interplanetary-filesystem-explained-what-is-ipfs.

[16] Faundez-Zanuy, Marcos. "Biometric security technology." IEEE Aerospace and Electronic Systems Magazine 21, no. 6 (2006): 15-26.

[17] Silicon Mechanics five advantages of IPFS and how to achieve them. Available at: https://www.siliconmechanics.com/news/5-advantages-of-interplanetary-filesystem.

[18] Garcia, A. (2023) What is biometric authentication? Klippa. Available at: https://www.klippa.com/en/blog/information/biometric-authentication/.

[19] Rastogi, Neha. "Biometrics Technology and Its Scope in Future." Engineers Garage, n.d. https://www.engineersgarage.com/biometrics-technology-and-itsscope-in-future/.

[20] "Patent, #8220;System, and, Method, Digitally, Signing, Documents, Using, Biometric, Data, in, a, Blockchain, or, PKI, #8221;" PatentPC. @PatentPC, January 6, 2023. https://www.patentpc.com/blog/patent-for-system-and-method-fordigitally-signing-documents-using-biometric-data-in-a-blockchain-or-pkius10516538b2.

[21] Editor. "Non-Functional Requirements: Examples, Types, How to Approach." AltexSoft. AltexSoft, February 12, 2020. https://www.altexsoft.com/blog/nonfunctional-requirements/.

[22] "Why Python Is the Best Programming Languages for Web Development." Probytes Web Development Company, June 11, 2018. https://www.probytes.net/blog/python-web-development/.

[23] "Introduction to Flask." Introduction to Flask - Python for you and me 0.5.beta1 documentation, n.d. https://pymbook.readthedocs.io/en/latest/flask.html.

[24] Cobb, Michael. "What Is the RSA Algorithm? Definition from Searchsecurity." Security. TechTarget, November 4, 2021. https://www.techtarget.com/searchsecurity/definition/RSA.

[25] Rebecca. "RSA Encryption Explained – Everything You Need to Know." History, March 3, 2023. https://history-computer.com/rsa-encryption/.

[26] Frankel, Sheila, Rob Glenn, and Scott Kelly. The AES-CBC cipher algorithm and its use with IPsec. No. rfc3602. 2003.

[27] Ch Sekhar, P Sai Meghana (2020). "A Study on Backpropagation in Artificial Neural Networks". Asia-Pacific Journal of Neural Networks and Its ApplicationsVol.4, No.1 (2020), pp.21-28. doi: 10.21742/AJNNIA.2020.4.1.03

[28] Pradeep. "Building Your First Neural Network with Tensorflow – Deep Learning 2." The Geek's Diary. The Geeks Diary, March 24, 2023. https://thegeeksdiary.com/2023/03/24/building-your-first-neural-network-withtensorflow-deep-learning-2/.

[29] "Blockchain - Elliptic Curve Cryptography." GeeksforGeeks. GeeksforGeeks, November 17, 2022. https://www.geeksforgeeks.org/blockchain-elliptic-curvecryptography/.