

FLAMO: Family Legacy And Memory Organizer

Mustafa Mustaqim Shaikh
Department of Computer Science
Abeda Inamdar Senior College
Pune, India

Shakila Siddavatam
Department of Computer Science
Abeda Inamdar Senior College
Pune, India

Abstract— People have always tried to keep family memories and heirlooms safe—mostly with boxes in the attic or by uploading photos and stories to whatever online platform seems popular. But those options fall short. They don't really give families proper control over who can access or change things. You can't always prove who owns what, and there's no guarantee a memory or heirloom hasn't been tampered with. That's where FLAMO (Family Memory and Legacy Organizer) comes in. It's a digital heritage management system that uses blockchain to lock down family memories and heirlooms—making sure they stay safe, organized, and real. FLAMO runs on a Model-View-Controller setup with Node.js, Express.js, MongoDB, and React, so it's built to scale. For security, it uses JSON Web Tokens for authentication, plus role-based controls so only the original uploader can change shared content. When someone adds a new heirloom or memory, FLAMO records its metadata on the Polygon blockchain. That way, you get a cryptographic proof that nobody's messed with it. Storage is smart, too—a deduplication system keeps things efficient by avoiding unnecessary copies. Tests show FLAMO delivers real improvements: stronger data integrity, clear ownership, and more trust than your average cloud storage solution. In short, FLAMO gives families a secure, scalable way to protect their digital legacy for the future..

Keywords— Digital Heritage; Blockchain Verification; Polygon Network; Family-Based Access Control; MVC Architecture; JWT Authentication; Data Deduplication; Digital Legacy Management; Decentralized Storage Framework

Digital technology keeps moving fast, changing the way we save and share the things that matter to us—photos, videos, legal papers, family keepsakes. More and more, these memories live in the cloud instead of a box under the bed. But here's the problem: most digital storage is really built for one person at a time. There's not much structure for families to share, prove ownership, or make sure nothing gets lost or tampered with down the line.

Sure, traditional cloud platforms make it easy to store and access files from anywhere, but they don't sort out who owns what when several people are involved. Shared accounts can get messy. People overwrite each other's files by accident, duplicates pile up, and sooner or later, there's confusion over

what's real or original. And don't forget—if something goes wrong at the cloud provider, your family's memories could be at risk.

Blockchain steps in with a different approach. It keeps records decentralized and locked in, so nobody can sneakily change things. In digital asset management, blockchain's already building more trust and keeping data honest. But so far, not many people have taken a close look at how blockchain could help families hold on to their digital heritage in a more structured way.

That's where this research comes in. We're introducing FLAMO—the Family Memory and Legacy Organizer. It's a platform built to keep your family's digital legacy safe and well-organized over the long haul. FLAMO uses JWT for secure sign-ins, sets up family roles so only the right people get access, stores files on AWS S3, and locks in authenticity with the Polygon blockchain. The whole thing runs on a modular MVC setup, which just means it's flexible and easy to grow. Everyone in the family can look back at shared memories, but only the original uploader can change them. Plus, FLAMO has a built-in filter to stop duplicate files from taking up space.

By combining the reliability of cloud storage with the transparency of blockchain, FLAMO gives families a solid, clear way to protect and share their digital stories for generations.

1.1 Problem Statement

There are plenty of digital storage options out there, but when it comes to preserving a family legacy, they really fall short. Most of these platforms focus on single users. They don't offer real family-friendly features like letting you set up different roles or share things in a way that makes sense for families. Then there's the problem with keeping things authentic. Right now, anyone can change digital heirlooms or important documents, and there's no way to see who did what or prove what's real. That's a recipe for arguments down the road. Ownership rules are pretty weak, too. There's nothing stopping someone from editing or deleting content, even if they weren't the one who uploaded it in the first place. Plus, everyone ends up uploading the same photos or files, so the system gets bogged down with duplicates. It's just not

efficient. And let's not forget — most of these platforms are centralized. If the main server goes down, or if you lose trust in the company, you're out of luck. All this shows we need something better: a secure, scalable, and decentralized way to manage digital family legacies.

1.2 Significance

FLAMO isn't just another place to store your family's digital stuff. It actually bridges the gap between old-school digital storage and the newer, more organized world of digital legacy management. Most platforms out there? They're built for individuals. You upload your files, and that's it. FLAMO flips that idea—this one's built for families. It takes ownership seriously, locks down who can access or change things, and checks authenticity at every turn. [1]. Digital Legacy That Actually Matters Let's be honest, most of our family memories—photos, stories, important records—are digital now. FLAMO steps up with a setup that keeps those memories safe and makes it clear who owns what. So when you trust your family history to FLAMO, you know it'll last. [2]. Blockchain You Can Actually Trust This is where things get interesting. FLAMO uses the Polygon blockchain to verify digital heirlooms. Every piece of metadata gets logged on a decentralized ledger. Nobody can sneak in and tweak things later. It's open, tamper-proof, and gives families a reason to trust the system. [3]. Real Family Access, Not a Chaos Zone With regular storage platforms, it's usually a free-for-all. FLAMO sets up a clear chain of command. Everyone can see the files, but only whoever uploaded something can change it. That keeps things honest and stops people from messing with each other's uploads. [4]. No Duplicate Clutter Usually, when families share storage, things get messy fast—same photos, same files, uploaded over and over. FLAMO's deduplication feature catches duplicates before they pile up. So you don't waste space or end up digging through five copies of the same picture. [5]. Built for Now, Ready for More FLAMO isn't just a flashy pitch. The team built it from scratch using React, Node.js, Express.js, and MongoDB, all tied together with a modular MVC setup. It works smoothly right now, and it's ready to scale up as families grow. [6]. Moving the Needle FLAMO isn't just a concept—it's actually making a difference: - Real-world blockchain verification for families - Secure group access, not just single-user control - Digital inheritance that actually works - Data management with families in mind Bottom line: FLAMO shows what's possible. It's not just an idea—it's out there, working, and delivering real value both in research and in real life.

1.3 Proposed Solution

FLAMO (Family Memory and Legacy Organizer) takes a new approach to digital family archives. Instead of the usual siloed or fragile storage systems, FLAMO brings in blockchain technology to keep memories safe, authentic, and easy to manage. Here's how it works. When you log in, FLAMO uses JWT authentication and bcrypt password hashing, so your family's stories stay private and secure. Everyone in the family can see shared memories, but only the person who uploads something can change or delete it. That way, the original owner always keeps control, and nothing gets lost or

changed by mistake. Authenticity matters, too. Every heirloom's metadata is stamped onto the Polygon blockchain. Register something, and FLAMO creates a unique transaction hash—it sits in the database as proof of origin. No one can tamper with it, so you always know your digital keepsakes are real. FLAMO also keeps things tidy behind the scenes. If someone tries to upload the same photo twice, the system catches it and avoids storing duplicates. This saves space and keeps the database running smoothly. The whole platform is built on a modern stack: React, Node.js, Express.js, and MongoDB, all organized with a modular MVC structure. That means FLAMO can grow with your family's needs while staying easy to maintain. By weaving blockchain verification together with thoughtful, family-first access controls, FLAMO gives families a secure, reliable way to preserve their digital legacies—no more worrying about lost files or who can see what.

2. Literature Review

Digital assets are growing fast, and that's got a lot of people looking into how we manage digital inheritance and family legacies. Sure, cloud storage is everywhere and pretty good at holding a lot of data, but it's really built for just one person at a time. It doesn't offer much if you want to share ownership with your family or prove something's genuine. And let's be honest, putting all your eggs in one basket with centralized storage means you risk things like hacking, tampering, or even just a total system crash. Lately, researchers have started looking at blockchain to tackle these problems. The big draw? Blockchain keeps records that can't be secretly changed, and everyone can see what's going on. It uses cryptography to make sure no one sneaks in edits or forges files, which helps families trust digital records. Plenty of people have come up with blockchain systems for inheritance—usually focused on money or legal stuff, not so much on keeping family memories safe and shared. Role-Based Access Control, or RBAC, is another tool that helps manage who can do what in a system. It works pretty well for controlling permissions in big organizations. But when it comes to family legacies, most RBAC setups don't really fit. Families need something more flexible, with different levels of access for different people. Then there's storage. Cloud systems use deduplication to avoid saving the same file over and over, which saves space and speeds things up. But you don't see these techniques much in platforms built for sharing and preserving family memories. So, despite all the progress in blockchain, access control, and storage, hardly anyone has put these ideas together into one system for families. That's where FLAMO comes in. It brings all the pieces together: - Blockchain for making sure files are real and unaltered - Access control that actually fits family hierarchies - Only uploaders can change their own files - Deduplication to save space and keep things running smoothly - A full-stack, scalable setup you can grow with Bottom line, FLAMO isn't just another storage platform. It's built as a practical, all-in-one solution for families who want to keep their digital heritage safe, organized, and easy to share..

Research Gap

People have poured a lot of effort into digital storage, blockchain asset management, and access control systems. But

honestly, when it comes to preserving a family’s digital legacy, there are still some big gaps. Most cloud storage platforms? They’re built for individuals. There’s no real setup for families to share ownership in an organized way. And when more than one person tries to add or change something, things can get messy fast—there’s nothing stopping someone from accidentally (or intentionally) overwriting someone else’s contribution. This can lead to arguments and, worse, you could lose important memories. Blockchain research usually zeroes in on money—like crypto wallets or automating legal stuff with smart contracts. Hardly anyone’s looked at how blockchain could help families protect their memories or keep family heirlooms authentic, especially when everyone wants to collaborate. Role-based access control is great for tightening up security, but it doesn’t really get the way families work. In a family, you need clear lines: who gets to see what, who can actually edit, who just watches. Those models don’t really tackle the messy, layered nature of family permissions. And while data deduplication—the art of cutting out duplicate files—is a big deal in the cloud world, hardly anyone brings that thinking into family memory management. So, there’s still no all-in-one system that truly nails: Real family-based access control Only letting the person who uploads something change it Making sure everything’s authentic using blockchain Saving space by cutting out duplicates And doing all of this in a way that actually scales and works in the real world That’s where FLAMO comes in. It brings all these pieces together into one practical system built just for families to manage their digital legacy—finally closing that gap.

3. Methodology (Development Process)

3.1 System Design Approach

FLAMO uses the Model-View-Controller (MVC) pattern, which keeps things modular, easy to scale, and simple to maintain. Here’s how it all comes together:

The frontend runs on React, so users get quick and interactive features. For the backend, Node.js and Express.js handle all the API requests and business logic. MongoDB takes care of storing user info, family details, memory metadata, and those blockchain transaction hashes. On top of that, the Polygon network handles decentralized heirloom verification.

Breaking things up like this keeps each part focused on its own job. It also makes it easier to manage and update the system down the road.

3.2 Authentication and Authorization Mechanism

We used JSON Web Tokens (JWT) for user authentication. When someone logs in, the system creates a secure token and sends it with their API requests.

Passwords? All hashed with bcrypt before they hit the database.

For authorization, we set up middleware to keep things tight: only logged-in users can reach protected resources, and only the person who uploaded a memory can change or delete it.

3.3 Family-Based Access Model Implementation

We set up a Family ID system so users can either start a family group or join one. Everyone in the group can see what’s shared, but only the person who uploaded

something can edit or delete it. That way, people stay responsible for their own uploads and nobody messes with someone else’s stuff without permission.

3.4 Blockchain Integration Process

Here’s how heirloom verification works:

First, the user shares the heirloom’s metadata. That info gets sent to the Polygon blockchain using Web3. Once the transaction goes through, it’s locked in on the blockchain. The system then stores the transaction hash in MongoDB, so it’s easy to check later. This way, heirloom records stay authentic and can’t be tampered with

3.5 Deduplication Mechanism.

To prevent redundant storage, uploaded files were checked against existing database records before saving. If duplicate metadata or file identifiers were detected, the upload was restricted. This improved storage efficiency and reduced redundancy.

3.6 Experimental Evaluation

We ran the system in a controlled development environment and checked a few key things: authentication security, how well access control worked, whether blockchain transactions were verified properly, deduplication accuracy, and how fast the system responded. The results showed stronger ownership integrity, less redundancy, and safer verification than you get with old-school centralized storage.

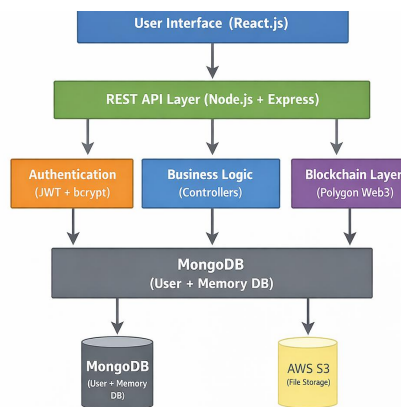


Figure 1 System Architecture

4.1 Technologies Used

FLAMO runs on a mix of modern full-stack web tools, decentralized blockchain tech, and cloud storage. Here’s what the team used to build it:

Table 1: Technology Stack for Fake Job Posting Detection System

Component	Technology Used
Frontend	React.js, HTML, CSS, JavaScript
Backend	Nodejs, Expressjs, MVC Architecture
Database Technologies	MongoDB

Component	Technology Used
Authentication and Security	JSON Web Token (JWT)
Cloud Storage	AWS S3
Blockchain Technology	Polygon Blockchain Network
API Communication	REST API

4.2 User Interface (UI) & Screenshots

FLAMO's interface runs on React.js, so it feels quick and easy to use right from the start. You can find your way around without hunting for anything—the navigation just makes sense. Families can interact with the system in a way that feels natural, and when it comes to verification, blockchain tech handles it smoothly in the background. Everything's split into different screens, each with its own purpose.

4.2.1 User Interface Overview

The system provides the following user interfaces:

- **Landingpage:**

The Landing Page is where everything starts. Right away, it tells you what FLAMO is all about—saving your digital legacy, locking it in with blockchain, and letting your family control who gets access. You'll see clear buttons to sign up or log in, so getting started is simple. as shown in Figure 2

- **Sign-Up Page:**

On the Registration Page, new users sign up by entering their personal information. The system hashes their passwords before saving anything, so your data stays safe. as shown in Figure 3

- **Login Page:**

When you log in, the Login Page checks your info. If everything matches, it hands you a JWT token. With that, you can get into the protected parts of the site. as shown in Figure 4

- **Dashboard:**

Once you log in, the Dashboard is your main hub. Here, you can manage your family, upload memories, register heirlooms, verify things on the blockchain, and tweak your settings. Everything's laid out clearly, so you'll find both your own content and anything your family shares, all in one place. as shown in Figure 5

- **Family Management Page:**

This module enables users to create or join a family using a unique Family ID. All family members can view shared content, while editing rights are restricted to original uploaders as shown in Figure 6

- **Memory Upload Page:**

Users can upload images and descriptions through this interface. Media files are stored in AWS S3, and metadata is saved in MongoDB. Duplicate uploads are prevented through a deduplication mechanism. as shown in Figure 7

- **Memories View Page:**

These page shows the memories stored by the user the owner of memory can update or delete memory other family member can only view memories as shown in Figure 8

- **Memories Preview Page:**

User can see the memories one by one and also can download single image if user wanted as shown in Figure 9

- **Heirloom registration and heirlooms page:**

The Heirloom Registration Page allows users to register heirloom metadata on the Polygon blockchain. A transaction hash is generated and stored for verification.

TheHeirlooms Page displays the heirlooms details to confirm authenticity as shown in Figure 10

- **Transfer Heirloom Page:**

In these page we can transfer the heirloom from one user to another but only when the next owner is from same family and younger than previous owner as shown in Figure 11

- **Edit Pages of Heirloom and Memory:**

In these pages user can edit the memories or heirlooms in memories he can add or delete the images as shown in Figure 12 and 13

- **Settings Page:**

Basic settings page shows the user data his family wallet address and family id as shown in Figure 14

4.2.2 UI Screenshots

The following figures illustrate the key user interface screens of the Fake Job Posting Detection system

Table 2: Description of System Interface Screens and Database Components

Figure No.	Description
Figure 2	Homepage displaying system overview, features, and register/login options
Figure 3	Sign up page for new user registration
Figure 4	Login page for registered users
Figure 5	Dashboard page users activity shown
Figure 6	Family Management Page
Figure 7	Memory upload page
Figure 8	Memory View page
Figure 9	Memory Preview page
Figure 10 & 11	Heirloom registration and heirlooms page
Figure 12	Transfer Heirloom Page
Figure 13 & 14	Edit Pages of Heirloom and Memory
Figure 15	Settings Page

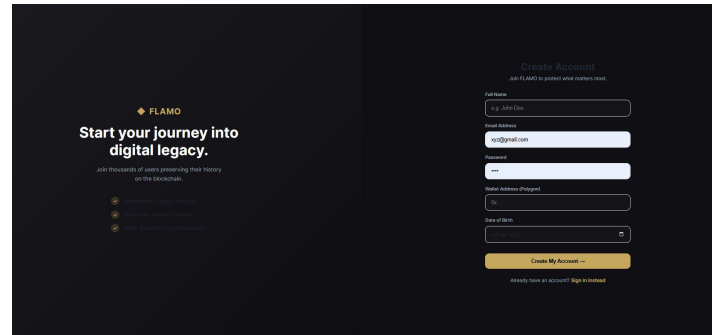


Figure 3

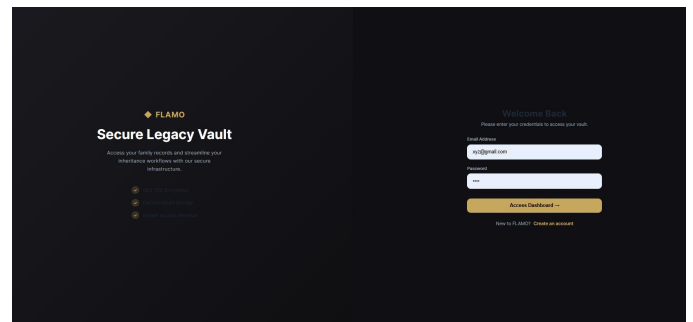


Figure 4

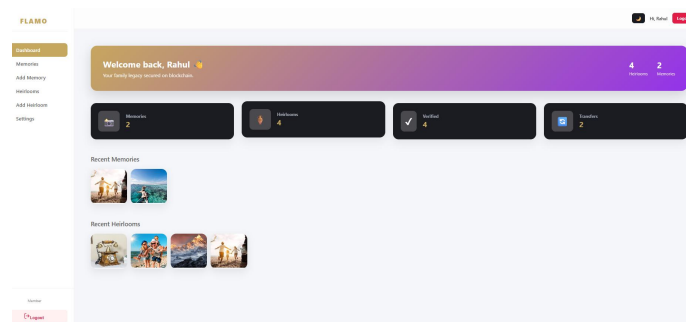


Figure 5



Figure 2

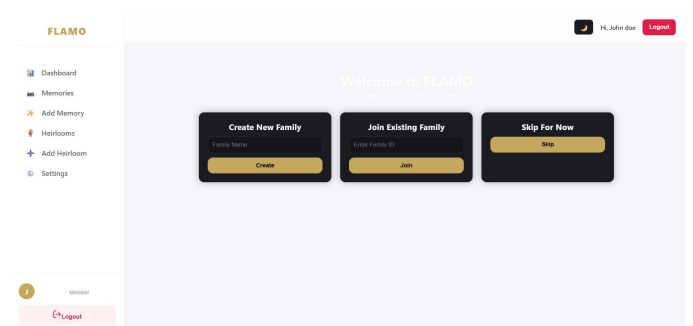


Figure 6

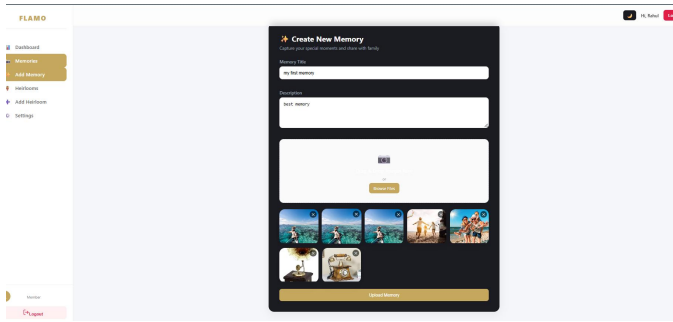


Figure 7

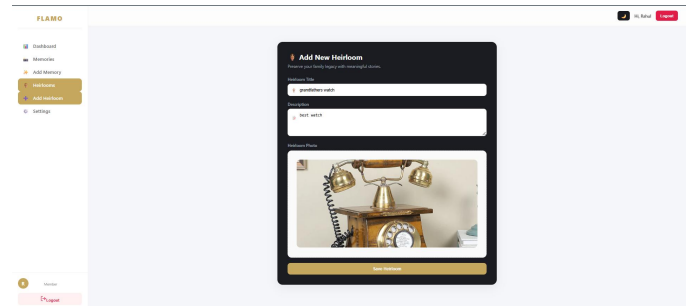


Figure 11

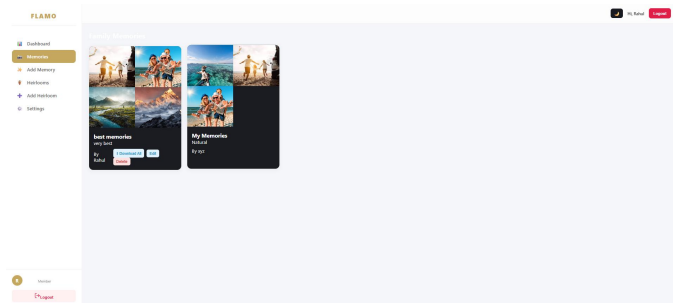


Figure 8

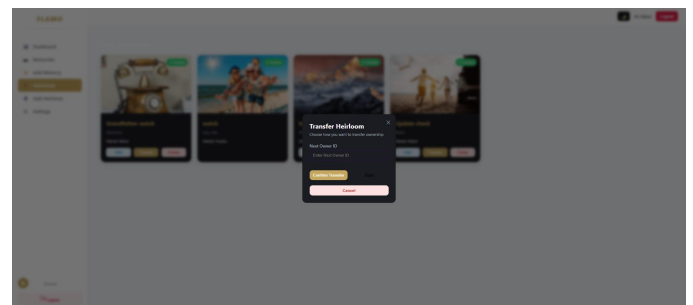


Figure 12

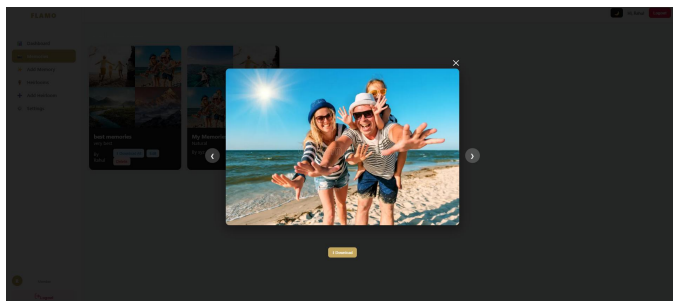


Figure 9

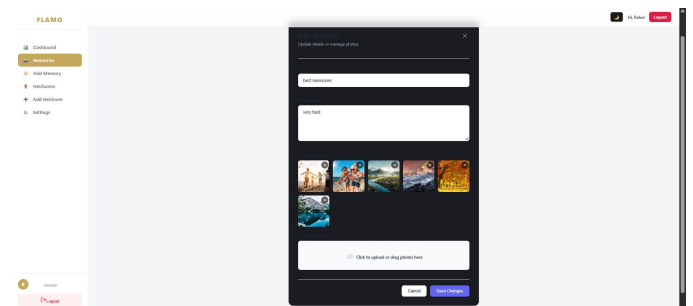


Figure 13

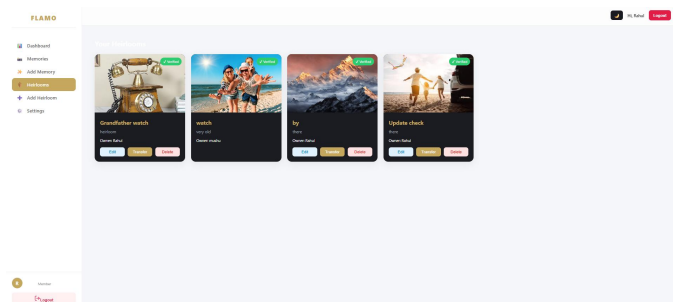


Figure 10

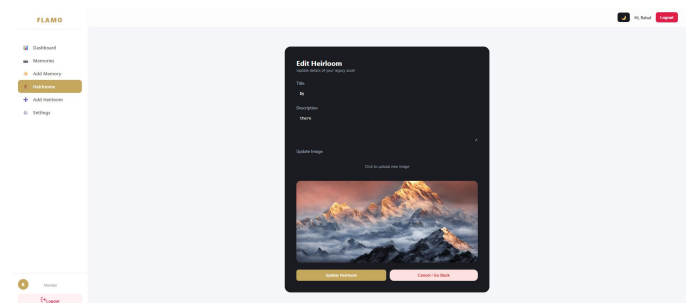


Figure 14

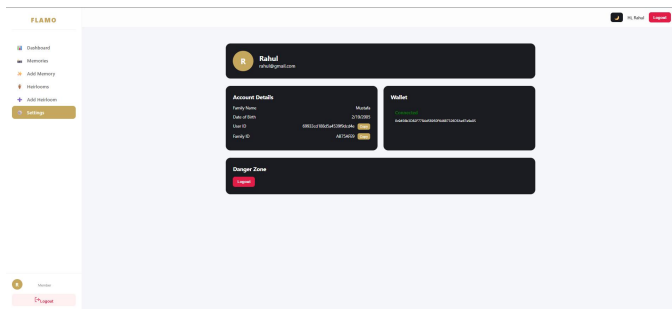


Figure 15

5. Discussion

5.1 Strengths of the System

FLAMO stands out in digital legacy management for a few big reasons.

For starters, it blends blockchain tech with family-based access control, so heirloom records stay untouched and authentic. Thanks to the Polygon network, you get decentralized verification without crazy transaction fees.

The family-centric access model is also well thought out. Only the person who uploads something can change it, but everyone in the family gets to view it. That keeps things safe from unwanted edits, while still letting families share memories together.

On the security side, FLAMO uses JWT authentication and bcrypt to lock down user credentials. That's real peace of mind. Storage is smarter, too: the system skips duplicate uploads, saving space and keeping things tidy.

And since it's all built on a modular MVC framework, FLAMO is easy to scale, maintain, and update. Everything stays organized and clear behind the scenes.

Bottom line: FLAMO gives families a secure, transparent way to preserve their digital heritage, and it's built to last

5.2 Challenges and Limitations

FLAMO's got a lot going for it, but it's not perfect.

Bringing blockchain into the mix means you get some transaction delays and gas fees. When people start registering a ton of heirlooms at once, things can slow down. Sure, using Polygon keeps costs lower than some other options, but you're still at the mercy of the network.

Right now, FLAMO checks for duplicates using metadata. That works most of the time, but if someone changes an image a lot, the system might miss it. Smarter, content-based checks would catch more of those sneaky duplicates.

FLAMO also keeps its metadata in MongoDB, so it's not totally decentralized. If you want to make the system tougher and less reliant on any one place, switching to a fully decentralized storage setup would help.

And one last thing: all the user testing happened in a controlled setting. Once FLAMO hits the real world and scales up, more performance challenges are bound to pop up.

5.3 Future Scope

Here's where FLAMO can really push forward.

First, bringing in IPFS means files aren't just stored—they're scattered across the network, locked in for good. No more worrying about data disappearing or getting tampered with.

Smart contracts step in next. With them, digital inheritance happens automatically, just as planned, no need for anyone to jump through hoops.

AI can make things smoother too. It sorts memories, adds tags, and makes searching a breeze. You don't have to dig around for hours to find what matters.

A mobile app? That's a no-brainer. People want to use this stuff anywhere, anytime. Easy access usually means more users coming onboard.

And if you're thinking bigger—like handling digital legacies for whole families or even companies—advanced encryption and layered structures can handle it. It keeps everything secure, organized, and ready for whatever comes next.

6. Conclusion

Online recruitment platforms have significantly simplified job searching; however, they have also created opportunities for fraudulent job postings that exploit job seekers. Manual moderation systems are inadequate to address the scale and evolving sophistication of job scams. To address this issue, this research proposed and implemented a Fake Job Posting Detection system using Machine Learning and Natural Language Processing techniques. The system analyzes job-related textual features and classifies postings as real or fake using a lightweight and interpretable Logistic Regression model. The integration of SHAP explainability ensures transparency in decision-making, enabling users to understand the factors influencing predictions. Although the system has certain limitations, such as dataset dependency and language constraints, it represents a reliable and scalable solution to a critical real-world problem. With further enhancements such as live platform integration, multilingual support, and advanced models, the proposed system has the potential to significantly improve online recruitment safety and user trust.

7. References

1. S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," 2008. Pillai, S., Fake Job Identification Using Bidirectional LSTM and Word2Vec, Journal of Intelligent Computing, 2023.
2. M. Crosby, P. Pattanayak, S. Verma, and V. Kalyanaraman, "Blockchain Technology: Beyond

- Bitcoin,” *Applied Innovation Review*, no. 2, pp. 6–19, 2016.
3. G. Wood, “Ethereum: A Secure Decentralised Generalised Transaction Ledger,” *Ethereum Project Yellow Paper*, 2014
 4. Polygon Technology, “Polygon Whitepaper,” Polygon Labs, 2021.
 5. J. Boneau et al., “Research Perspectives and Challenges for Bitcoin and Cryptocurrencies,” *IEEE Security & Privacy*, vol. 13, no. 4, pp. 104–113, 2015.
 6. Sandhu, E. Coyne, H. Feinstein, and C. Youman, “Role-Based Access Control Models,” *IEEE Computer*, vol. 29, no. 2, pp. 38–47, 1996.
 7. Bethencourt, A. Sahai, and B. Waters, “Ciphertext-Policy Attribute-Based Encryption,” *IEEE Symposium on Security and Privacy*, 2007.
 8. M. Bellare and P. Rogaway, “Introduction to Modern Cryptography,” UCSD Technical Report, 2005.
 9. D. Eastlake and P. Jones, “US Secure Hash Algorithm 1 (SHA1),” RFC 3174, IETF, 2001.
 10. A. Elmagarmid, P. Ipeirotis, and V. Verykios, “Duplicate Record Detection: A Survey,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 1, pp. 1–16, 2007.
 11. J. Gantz and D. Reinsel, “The Digital Universe Decade – Are You Ready?” IDC Report, 2010.
 12. T. Berners-Lee, R. Fielding, and H. Frystyk, “Hypertext Transfer Protocol – HTTP/1.0,” IETF RFC 1945, 1996.
 13. R. Fielding, “Architectural Styles and the Design of Network-based Software Architectures,” Ph.D. dissertation, University of California, 2000.
 14. E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, 1994.
 15. MongoDB Inc., “MongoDB Architecture Guide,” 2023.
 16. Amazon Web Services, “Amazon Simple Storage Service (S3) – Developer Guide,” AWS Documentation, 2023.
 17. M. Jones, J. Bradley, and N. Sakimura, “JSON Web Token (JWT),” IETF RFC 7519, 2015.
 18. Provos, N., and Mazieres, D., “A Future-Adaptable Password Scheme,” *USENIX Annual Technical Conference*, 1999. (bcrypt reference)