



# Blockchain-Integrated Full-Stack Application

## With Cloud Services in the insurance industry

IS&T, Computer Science

### CONTRIBUTORS:

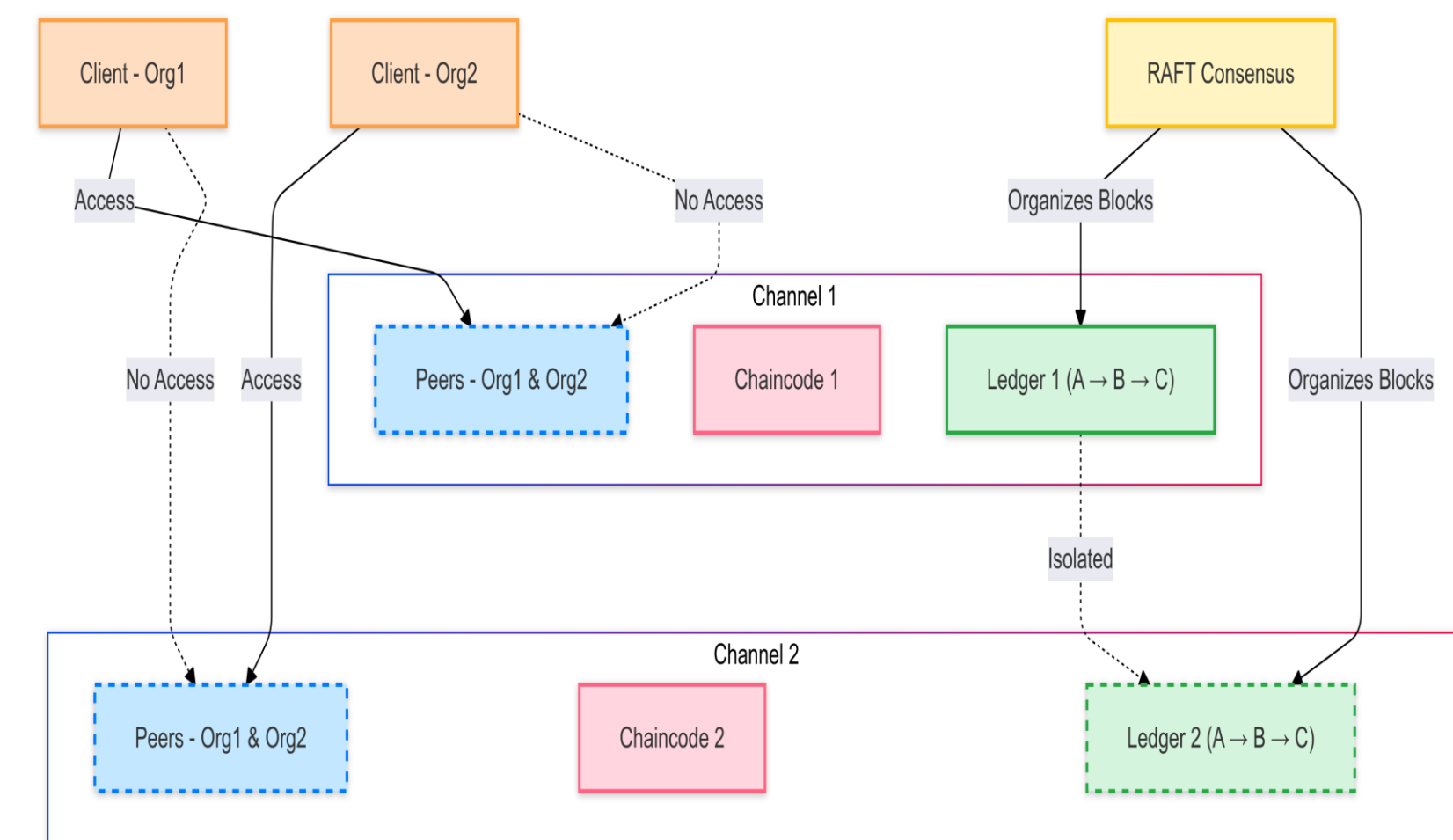
- Henry Nguyen
- Brian Tran
- Jonah Nathan
- Matthew James
- Dr. Alfredo Perez
- Dr. Cristian Mateos
- Toe Arkar

## Overview

We Care Insurance is creating a secure and efficient product to receive user insurance policies and record data. The solution leverages Hyperledger Fabric for blockchain-based policy exchanges, featuring:

- Insurance policies are typically exchanged via email, which lacks adequate security and efficiency for sensitive data.
- Private blockchain technology, such as Hyperledger Fabric, offers a secure, tamper-proof alternative.
- Blockchain ensures only authorized parties can access and share insurance policy data, reducing risks and enhancing trust.
- The application will replace email exchanges with a block-chain integrated solution
- The solution will utilize cloud services and automated text processing for secure, scalable exchange of mortgage insurance policies.

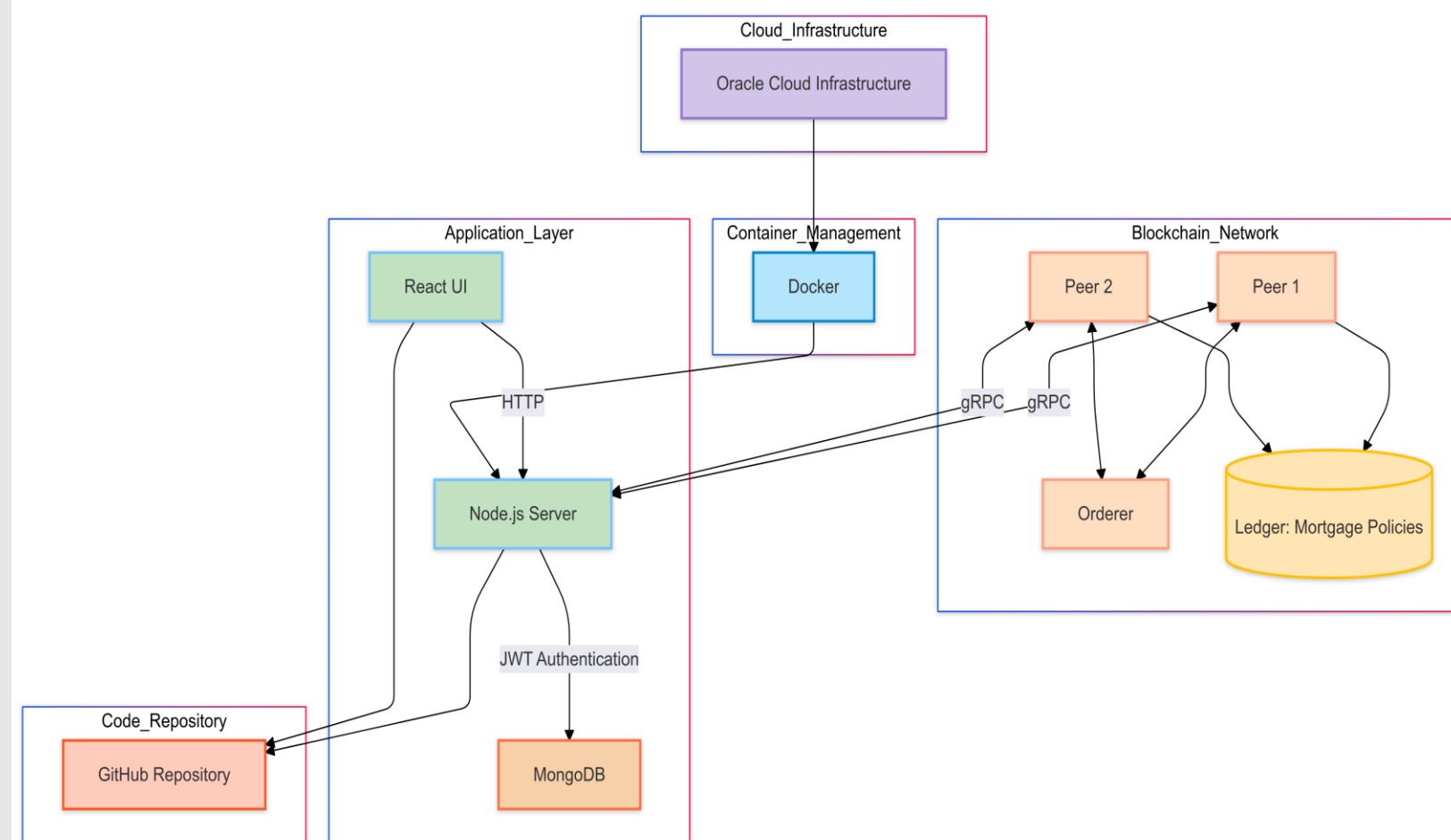
## Blockchain Network Overview



This flowchart outlines the architecture of a blockchain-based system for managing mortgage policies, providing a detailed view of its layered structure and operations.

- **Ledger Structure:**
  - **Linked List:** Blocks are sequentially chained (A → B → C), ensuring data integrity.
  - **Binary Format:** Data is stored efficiently for high performance.
- **Block Components:**
  - **Header:** Includes block number, previous hash, and transaction IDs.
  - **Data:** Contains transaction details.
  - **Metadata:** Holds block signatures and additional information.
  - **History:** Tracks changes and updates to transactions.
- **Consensus Mechanism:**
  - **Orderer:** Arranges and orders transaction blocks.
  - **Consensus Algorithm:** Uses RAFT to validate blocks through majority agreement.
- **Channels:**
  - **Peers:** Network nodes (e.g., Org1, Org2) that execute transactions.
  - **Chaincode:** Smart contracts defining business logic for transactions.
  - **Ledger:** Stores the sequence of blocks and transactions securely.
  - **Isolation:** Ensures ledgers and chaincodes are separate per channel for confidentiality.

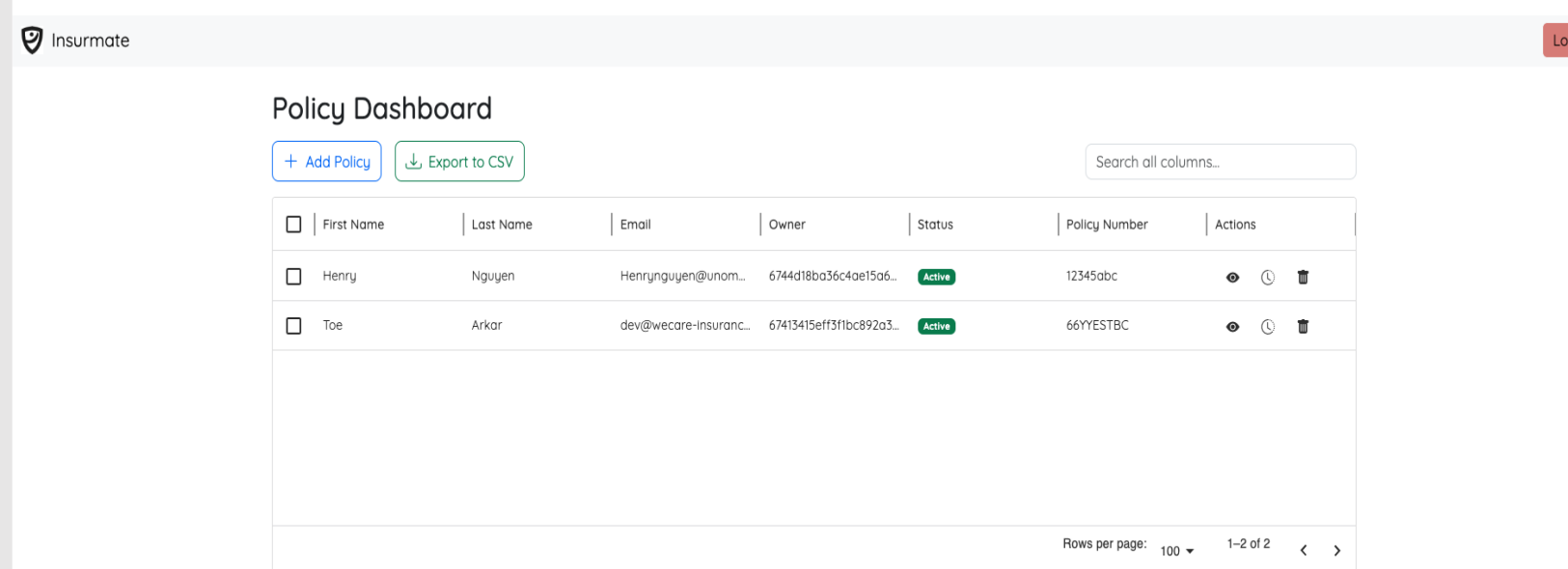
## Application



This flowchart outlines a blockchain-based architecture for managing mortgage policies, broken down into key components:

1. **Blockchain Network**
  - **Key Elements:** Peer 1, Peer 2, Orderer, and a Ledger that stores mortgage policies in binary format.
  - **Connections:** Peers communicate with the Orderer and interact directly with the Ledger for storing and retrieving data.
2. **Application Layer**
  - **Components:** Node.js Server, React UI, and MongoDB.
  - **Workflow:** React communicates with Node.js via HTTP for application logic, while MongoDB handles authentication and data storage using JWT.
3. **Cloud Infrastructure**
  - Uses **Oracle Cloud Infrastructure (OCI)** to host and manage the application's resources.
4. **Container Management**
  - **Docker** is employed for containerizing and running application services in isolated environments.
5. **Code Repository**
  - **GitHub** serves as the central repository for version control and source code management.

## Features and Achievements



- **Research:** Explored Hyperledger Fabric for secure exchange of mortgage insurance policies.
- **Account Management:** OTP verification, secure login, password reset, and session management with password hashing.
- **Security Enhancements:**
  - Helmet middleware for XSS and clickjacking protection.
  - XSS protection headers and rate limiting (1,000 requests/15 mins).
  - JWT authentication with Passport.js for secure API access.
- **React Frontend:** User-friendly interface for registration, login, password reset, and policy management.
- **Blockchain Integration:** RESTful endpoints for secure policy creation, updates, and deletions via Hyperledger Fabric.
- **Testing & Collaboration:** Completed unit tests and managed the project on GitHub for version control and collaboration.
- **Launched:** Minimal viable product live at [insurmate.app](https://insurmate.app).

## Future Enhancements

### Backend & Frontend Features:

- Generate one-time upload links for document submissions.
- Design a document upload interface.
- Integrate with a 3rd party API for address verification.

### LLM Integration:

- Fraud detection via inconsistency checks and policy verification.
- Expiration tracking using Optical Character Recognition (OCR) and Natural Language Processing (NLP) for policy renewals.
- Data validation to ensure database accuracy.

### Document Verification:

- Evaluate Amazon Textract API and alternative Large Language Model (LLM) solutions.

### Testing Improvements:

- Implement unit and integration tests for enhanced reliability.

### Future Scalability:

- Add features to address emerging needs and opportunities.

## Conclusion

Private blockchains provide a secure and efficient solution for handling sensitive data in industries like insurance. To leverage this technology, we created a user-friendly website that allows users to securely manage insurance data and interact with smart contracts. This integration ensures seamless access while maintaining data integrity and security.

## References

- Hyperledger Fabric: [https://docs.google.com/presentation/d/1Maqwoc0X94\\_GD73R2wzIeUJmM6\\_n48T9yGbWeIYUjI/edit#slide=id.g9ad9b01cb9\\_0\\_59](https://docs.google.com/presentation/d/1Maqwoc0X94_GD73R2wzIeUJmM6_n48T9yGbWeIYUjI/edit#slide=id.g9ad9b01cb9_0_59)
- Using Hyperledger Fabric in Node.js Application: <https://github.com/Toe12/blockchain-experimental>
- Amazon Textract: <https://aws.amazon.com/textract/>
- Verifying Insurance Documents using LLM: [https://github.com/Toe12/Ollama\\_PDF\\_Verification\\_App](https://github.com/Toe12/Ollama_PDF_Verification_App)
- Oracle Cloud Infrastructure: <https://www.oracle.com/cloud/>
- Oracle Cloud Infrastructure Object Storage: <https://docs.oracle.com/en-us/iaas/Content/Object/Concepts/objectstorageoverview.htm>
- Amazon S3 (Object Storage): <https://aws.amazon.com/s3/>



## Acknowledgements

We would like to express our sincere gratitude to We Care Insurance for providing the opportunity and use case that inspired this project. Their collaboration and insights were invaluable in helping us understand real-world challenges and how blockchain technology can address them.

## Application QR Code

