Central Bank of Nigeria

eNaira System Architecture Document

Version 1.0

May 2023

DOCUMENT INFORMATION

Document Reference:	eNaira: System Architecture
Author(s):	LUKMAN, SADIQ (SLUKMAN@CBN.GOV.NG) DANIA, JOHN (JODANIA@CBN.GOV.NG) UMAR, ASMAU A. (AAUMAR1@CBN.GOV.NG)
Date Created:	MAY 2023
Reviewers:	FADELE, ADEOLU ISAAC (AIFADELE@CBN.GOV.NG)
Authorized By:	MUSA, MOHAMMED HAMISU (MHMUSA@CBN.GOV.NG)

Distribution:	Business Relationship Management Division (BRMD);
	Project Management Office (PMO);
	Application Management Division (AMD);

SPONSOR INFORMATION	
Department:	Information Technology Department (ITD)
Contact Person Name:	MOHAMMED, RAKIYA SHUAIBU
Email:	RSMOHAMMED@CBN.GOV.NG

Document Control

This document is under the control of its author(s) until its approval.

CHANGE HISTORY

Date	Reviewed by	Version	Comments	Approved by

APPROVAL

S/No	Approver	Signature
1 F	Director, Information Technology Department (DITD)	OKSamuel
2	Deputy Director, Architecture & Strategy Division (DD ASD)	4-08-2023
3	Assistant Director, IT Architecture Office (AD ITA)	Mhmusa

This page intentionally left
blank

TABLE OF CONTENTS

1. Executive Summary

1.1 Background

The Central Bank of Nigeria (CBN) uses the Digital Currency Management System (DCMS) to mint and issue eNaira. The DCMS platform also houses eNaira wallets that may be used by many stakeholders, such as the Federal Government (FG), financial institutions (FIs) and other financial institutions (OFIs); ministries, departments, and agencies (MDAs); businesses; and consumers, among others, for transactions.

1.2 Problem Statement

The DCMS, the foundation upon which the eNaira CBDC was constructed, was created and is currently managed by an external vendor, which practically presents some issues, including but not limited to restricting the CBN's usability in terms of abrupt improvement, change implementations as and when due, performance issues, also taking into account the annual cost of operations, and live support of the CBDC project.

1.3 Proposed Solution

In view of the aforementioned, the Management of the CBN has mandated the creation and deployment of eNaira, a system whose improvement will be primarily targeted at improving and stabilizing the performance of the Layer 1 and 2, and every other service on the DCMS.

1.4 Value Proposition

eNaira introduces several key enhancements. The Routing Table Service to enable efficient wallet location retrieval, the Transaction Processing Engine to support concurrent transaction processing, and Wallet Management is improved by shifting wallets to Layer 2.

Additionally, Layer 1 of eNaira comprises the Key Vault, CA System, Identity Database, and Balance Database. By leveraging a fast database like RDBMS instead of a blockchain, eNaira achieves improved speed and efficiency while maintaining the necessary security measures for the CBDC system.

In conclusion, this solution architecture provides: a very high-level overview of the current architecture; the target architecture of the anticipated solution; a consideration of possible delivery choices; and a suggestion of an effective delivery option.

2. Introduction

2.1 Purpose of Document

This document outlines the system/solution architecture for eNaira. It further defines a tactical approach to describing the elements that address specific Central Bank Digital Currency components and business process requirements. This is done by describing the structure and behavior of the entire solution including processes that describes the work required to deliver the solution.

2.2 Scope

The scope of this document encompasses the AS-IS architecture of eNaira and a target architecture of the anticipated solution. The current challenge in meeting the business objective of the concerned stakeholders will be distilled and adequate references made, where necessary, to documented business processes and functional requirements as contained in the signed-off Business Domain Analysis and Requirement Specification Document titled “BRD: eNaira”.

The current architecture is represented only in as sufficient detail as serves the purpose of comparative analysis with anticipated changes in the target solution. At the strategic level, the solution architecture summarizes the drivers for the envisaged solution and the key objectives that it must support as well as the anticipated benefits. At business level, it maps services to the business unit stakeholders and their concerns, showing the interaction of various parts of the enterprise in using and supporting the envisaged CBDC Solution.

2.3 Resources

The information in this document was gathered from the requirement documents. Additionally, the solution architect used industry understanding of similar systems, best practices, discussions with the business teams and technical team supporting the existing process, and discussions with other stakeholders to ensure that the conceptual architecture is feasible and practical.

2.4 Compliance

The envisaged solution shall comply, at minimum, with the architecture recommendations as contained in this document. The solution shall conform to the components defined for the technical system and their interactions with each other, as well as relationship with other systems (technological and business) outside of the system. Any changes to the architecture definitions contained in this document are subject to the approval and ratification of the ITD Change Management Board prior to implementation.

2.5 Architecture Principles

The relevant sections as contained in the approved CBN Architecture Principles v1.0 are applicable.

2.6 Policies

The following policies and standards shall apply to configuration/set up of solution components:

i. IT Security policies
ii. CBN Data Privacy policy
iii. Third-party access policy
iv. CBN IT Standards
v. Other CBN applicable Policies

2.7 Document Maintenance

This document will be reviewed in the early stages of the project and updated as needed, as the project proceeds through each phase of the software development life cycle (SDLC). During the later stages of the project the conceptual model will be reviewed and managed via the governance and change management processes followed by the project.

This document contains a revision history log. When changes occur, the document's revision history log will reflect an updated version number as well as the date, the owner making the change, and change description will be recorded in the revision history log of the document.

3. Conceptual Architecture

This chapter introduces the high-level architecture of eNaira. It provides an overview of the Digital Currency Management System (DCMS) and highlights the additional features introduced in eNaira.

DCMS, the existing system, is composed of three layers. The first layer, called Layer 1, is responsible for transaction settlement and finality. It operates a private blockchain using Hyperledger Fabric and stores wallets and their balances. Layer 2, known as Byefrost 1, contains various business logic components, including user management, wallet service, compliance service, AML (Anti-Money Laundering) service, transactions, and charges service. Layer 3 comprises the channels (web and mobile app wallets, USSD channel, and APIs) that enable access to the CBDC.

eNaira, the enhanced system, will also consists of three layers. Layer 1 will undergo significant changes, while Layer 2 will be upgraded to Byefrost 2. Layer 3 remains unchanged, serving as the channels for CBDC access.

3.1 Layer II Components

The additional services are introduced to Byefrost II in eNaira:

i. **Routing Table Service:** The Routing Table Service is a new component in eNaira. Its purpose is to store and retrieve the location of wallets, indicating whether they are stored in the DCMS system or eNaira. This service enables efficient routing and retrieval of wallet information within the CBDC system.

ii. **Transaction Processing Engine:** eNaira introduces a robust Transaction Processing Engine capable of concurrently processing multiple transactions without compromising performance. To achieve this, it utilizes a Multiversion Concurrency Control (MVCC) mechanism. Additionally, a Global State Database is incorporated to track concurrent processes effectively. Implementing best practices, such as utilizing efficient data structures and optimizing transaction handling, can enhance the performance of the Transaction Processing Engine.

iii. **Wallet Management:** In DCMS, wallets reside in Layer 1, but eNaira improves upon this architecture. In the new system, wallets are created and managed at Layer 2. This shift allows for better separation of concerns and facilitates enhanced wallet management functionalities. By managing wallets at Layer 2, eNaira achieves a more streamlined and efficient wallet management process.

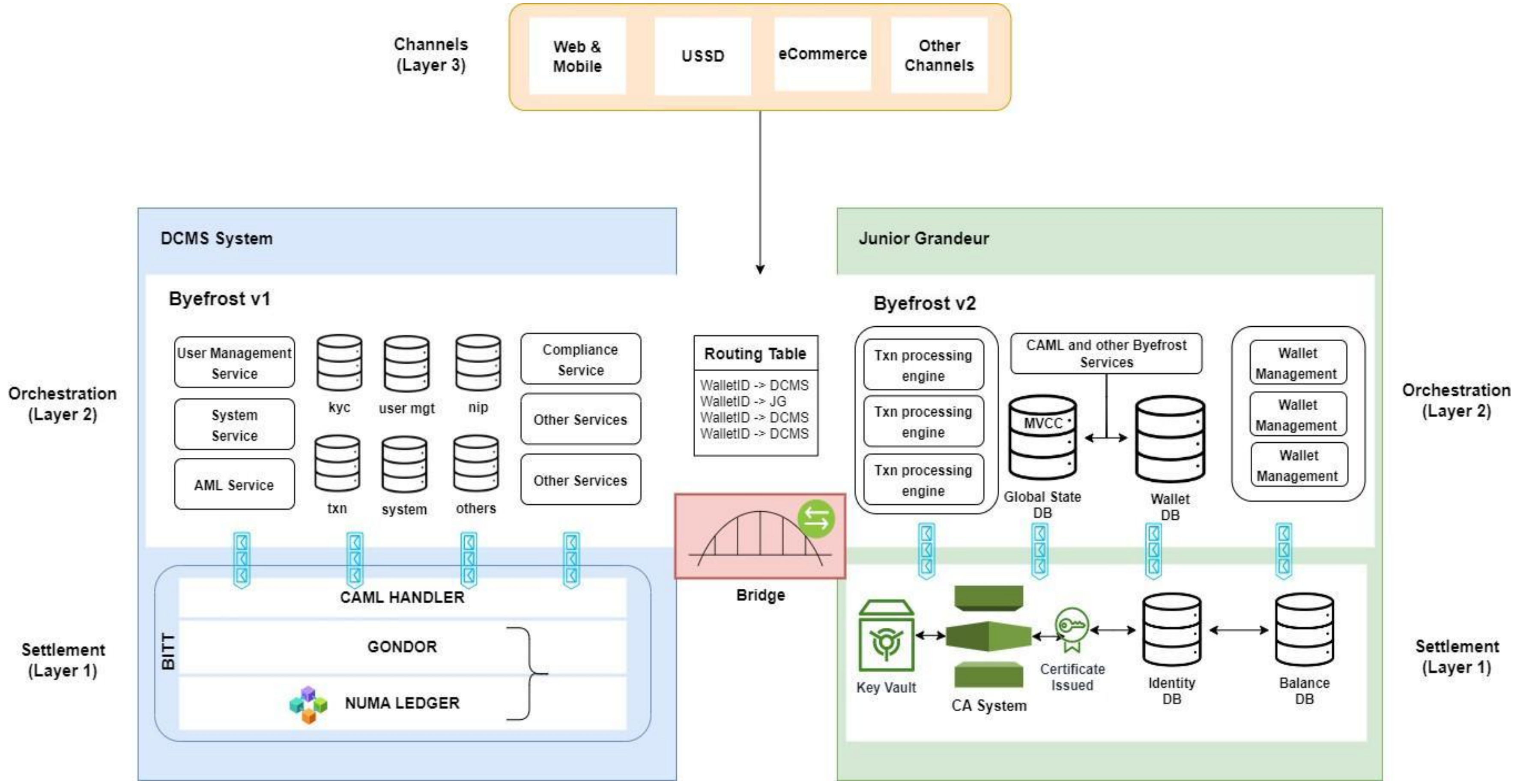


Figure 1. eNaira System Diagram

3.2 Layer I Components

eNaira's Layer 1 consists of the following components:

i. **Key Vault:** The Key Vault component manages the storage and management of cryptographic keys used for user and application authentication, encryption, and digital signatures. It ensures the security and integrity of the keys used within the CBDC system.

ii. **CA System:** The CA (Certificate Authority) System issues X.509 certificates to users and applications within the system. These certificates provide secure identification and enable secure communication between different entities. The CA System plays a vital role in maintaining the trust and security of the CBDC infrastructure.

iii. **Identity Database:** The Identity Database is responsible for storing the identities of each user within the CBDC system. After a certificate has been issued by the CA System, the corresponding user identity is stored in this database. The Identity Database facilitates authentication and authorization processes for secure interactions within the CBDC ecosystem.

iv. **Balance Database:** The Balance Database stores the balance information of each wallet within the system. This database ensures accurate and up-to-date records of wallet balances, allowing for seamless transaction processing and monitoring.

3.3 Cross-Chain Transaction Bridge

The Cross-Chain Transaction Bridge acts as a vital link between the DCMS and eNaira ecosystems, enabling interoperability and smooth exchange of assets and information. This bridge ensures that transactions within each ecosystem remain local chain transactions, while providing a secure and reliable channel for cross-chain transactions when the need arises.

Benefits of the Cross-Chain Transaction Bridge:

- Interoperability: The bridge enables seamless interoperability between the DCMS and eNaira ecosystems, allowing for cross-chain transactions without compromising security or efficiency.
- Asset Exchange: Users can easily transfer assets between the two systems, leveraging the strengths and features of each ecosystem.
- Flexibility: The bridge accommodates the unique requirements and protocols of both ecosystems, ensuring compatibility and smooth transaction flow.
- Security: The Cross-Chain Transaction Bridge incorporates robust verification mechanisms and asset locking to prevent double spending and ensure the integrity of transactions

4. Business Architecture

This chapter elaborates on the business case and requirements document by modeling the business context or environment within which the envisaged solution will operate. The selected artifacts illustrate business structures, functions, and their interactions to deliver the anticipated capabilities of the envisaged solution. The section sets out the justification for implementing the solution and highlights the motivations of the major stakeholders and anticipated users of the solution.

4.1 Business Motivation Model

This business motivation model provides an overview of the vision, mission, goals, and key stakeholders involved in the eNaira CBDC system. It highlights the focus areas and objectives that drive the development and implementation of the system to create a secure, inclusive, and efficient digital currency ecosystem.

Table 1 shows the business motivation.

BUSINESS MOTIVATION MODEL		
Vision	To be a people-focused Central Bank promoting confidence in the economy and enabling an improved standard of living	Enable a secure, efficient, and inclusive digital currency ecosystem.
Mission	To ensure monetary, price and financial system stability as a catalyst for inclusive growth and sustainable economic development.	Develop and implement the eNaira CBDC system to facilitate seamless digital transactions, enhance financial inclusion, and ensure trust and transparency.
Goals	Security and Trust	Establish a robust security framework to safeguard digital transactions, user identities, and funds. Foster trust and confidence in the eNaira CBDC system through strong encryption, authentication, and authorization mechanisms.
	Financial Inclusion	Promote financial inclusion by providing accessible and user-friendly digital wallet solutions. Ensure equal access and opportunity for all individuals and businesses to participate in the digital economy.

	Interoperability	Enable seamless interoperability between the DCMS and eNaira systems to facilitate cross-chain transactions. Establish a bridge adapter to ensure smooth transaction routing, validation, and settlement across the ecosystem.
	Operational Excellence	Implement efficient operations management practices to ensure system availability, performance, and scalability. Optimize transaction processing and settlement to achieve faster and more reliable transaction finality.
	User Experience	Deliver intuitive and user-friendly interfaces for wallet management, transaction processing, and account administration. Prioritize user-centric design principles to enhance the overall experience and adoption of the eNaira CBDC system.
	Compliance and Governance	Comply with relevant regulations, standards, and legal frameworks to ensure a secure and compliant digital currency ecosystem. Establish robust governance structures and policies to monitor and enforce compliance across the system.
	Innovation and Continuous Improvement	Foster a culture of innovation to adapt to emerging technologies and evolving user needs. Continuously enhance the eNaira CBDC system through feedback, research, and technological advancements.
Stakeholders	Central Bank	Responsible for overseeing and governing the eNaira CBDC system, ensuring monetary stability and regulatory compliance.

	Financial Institutions	Collaborate with the CBDC system to enable seamless integration and interoperability with existing financial infrastructure.
	Businesses and Merchants	Utilize the eNaira CBDC system to accept digital payments, streamline transactions, and enhance financial services.
	Individuals	Access the CBDC system to manage personal finances, make digital payments, and participate in the digital economy.
	Technology Providers	Contribute to the development, maintenance, and enhancement of the CBDC system's underlying technologies and infrastructure.

4.2 Stakeholder Map Matrix

Stakeholder management is an important discipline that successful architecture practitioners can use to win support from others. Identifying key stakeholders and their roles within a project and deployment cannot be overemphasized.

Table 2 shows the stakeholder map and interests.

Stakeholder	Involvement	Class	Relevant Artefact	Type
CBN Management	Increased adoption of the cashless policy, Increased financial inclusion etc.	Keep Satisfied	Business Architecture	**Internal**
Information Technology Department	To capture requirements necessary for the development of JG To co-develop the solution with a local Vendor	Manage Closely	Information System Architecture, Technology Architecure	**Internal**
eNaira TWC	Develop a high-level design document which includes the necessary technology stack, security protocols, usability features and all necessary considerations required for the implementation of JG	Manage Closely	Information System Architecture	**External**

Outsourced Vendors	To develop a solution to meet requirements of The eNaira Solution	Manage Closely	Information System Architecture, Technology Architecure	**External**

eNaira Consumers	To have the best possible experience when using the eNaira channels	Keep Satisfied	Application Architecture	**External**
Partner Agents	To onboard new customers to the eNaira Ecosystem Adequate registration, License and reward system to perform optimally.	Manage Closely	Integration Architecture	**External**

4.3 Process Flow Diagram

The eNaira digital currency is minted, issued, and distributed within the DCMS Network. Within this network, local chain transactions occur as peer-to-peer transactions between individual and business wallets.

For cross-chain transactions between the DCMS and the eNaira network, a bridge is utilized to facilitate the movement of funds. Let's consider the process steps involved in a transaction from a Sender in the DCMS to a Receiver in the eNaira network:

1. The Sender's wallet in the DCMS Network initiates a transfer of funds to the eNaira CIC wallet within the DCMS Network.
2. The funds are then transferred from the eNaira CIC wallet to the MHA CIC Wallet, which resides within the DCMS Network.
3. A message is sent to the Bridge Adapter, providing essential details such as the origin network (DCMS), destination network (eNaira), the transfer amount, sender wallet, receiver wallet, narration, timestamps, and other pertinent information.
4. The MHA Virtual CIC Wallet within the eNaira Network is subsequently credited with the transferred amount.
5. From the MHA Virtual CIC wallet, the funds are further transferred to the designated Receiver's wallet in the eNaira Network.

The same process applies in reverse when conducting a transaction from the eNaira network to the DCMS network.

Through the utilization of the bridge and the defined process steps, funds can be securely and efficiently moved between the DCMS and eNaira networks, enabling cross- chain transactions and facilitating seamless interoperability between the two ecosystems.

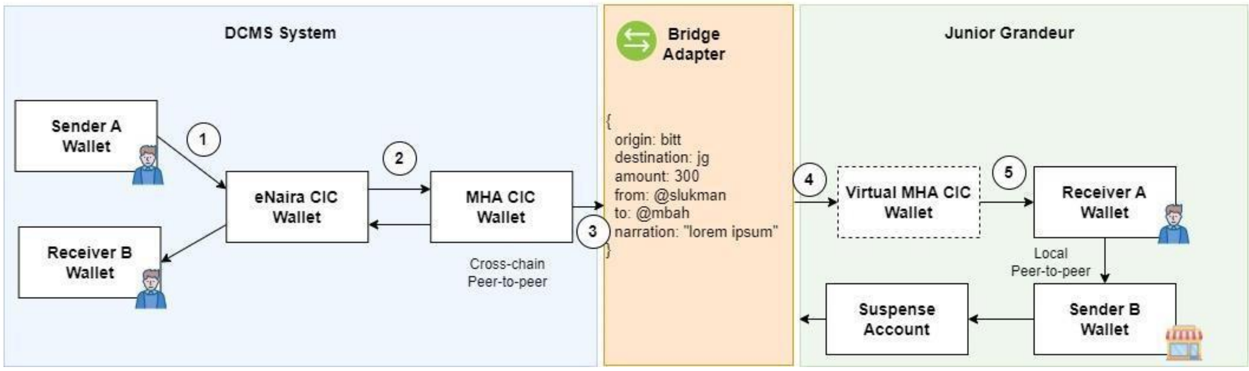


Figure 2. Process Flow Diagram for eNaira

4.4 Wallet Types

This section shows the types of wallets supported by the system.

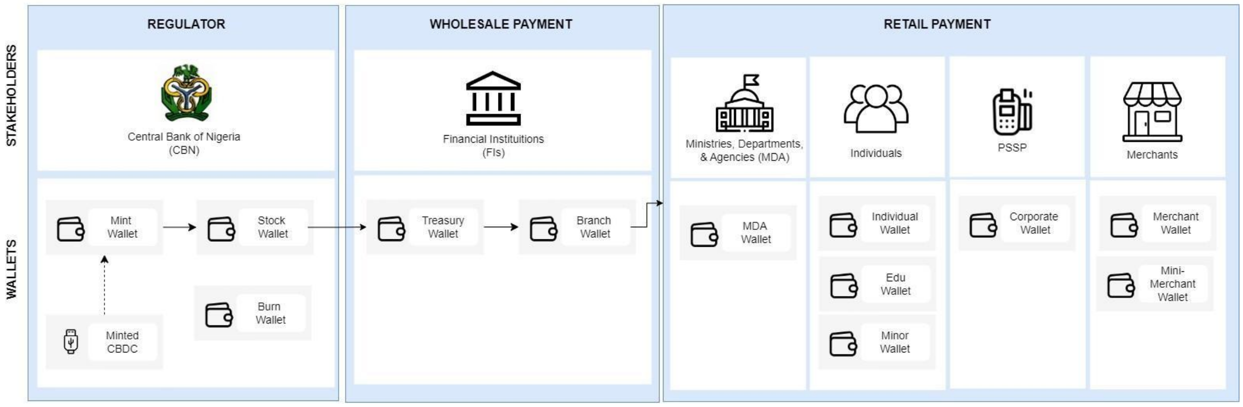


Figure 3. Wallet Types Diagram

4.5 Business Capability Model

The capability model shows the decomposition of what the business does and can do in a logical and granular grouping. A business capability model is one of the critical business architecture deliverables and a bridge between business and IT, and a foundation for enterprise transformation.

Table 3 shows the business capability of eNaira.

S/No	Capability Model		Capability
1	Service Account Management	Managing user accounts and access privileges within the eNaira CBDC system.	- User registration and onboarding. - User authentication and authorization. - User account maintenance and administration. - Role-based access control (RBAC) management.

			- Account deactivation and removal.
2	Identity & Access Management	Ensuring secure and authorized access to the eNaira CBDC system	- Identity verification and authentication. - User provisioning and deprovisioning. - Single sign-on (SSO) integration. - User attribute management. - Access control policy enforcement.
3	Key Management	Securely managing cryptographic keys used within the eNaira CBDC system	- Key generation, distribution, and rotation. - Key storage and protection. - Key usage and access control. - Key revocation and retirement. - Key recovery and backup.
4	Operations Management	Managing and monitoring the operational aspects of the eNaira CBDC system	- System monitoring and performance optimization. - Incident management and troubleshooting. - Logging and auditing. - System maintenance and upgrades. - Capacity planning and scalability.
5	Wallet Management	Managing the creation, storage, and usage of digital wallets within the eNaira CBDC system	- Wallet creation and setup. - Wallet balance management. - Transaction history tracking. - Wallet security and encryption. - Wallet integration with user interfaces (web, mobile apps, etc.)
6	Transaction Management	Facilitating and managing transactions within the eNaira CBDC system	- Transaction initiation and processing. - Transaction validation and authorization. - Transaction tracking and status updates. - Transaction fee calculation and deduction. - Transaction settlement and confirmation.
7	Bridge Adapter for Interoperability of DCMS and JG	Enabling cross-chain transactions between the DCMS and eNaira systems	- Interfacing with both systems for transaction routing. - Transaction translation and compatibility management.

		- Cross-system transaction validation and authorization. - Ensuring transaction consistency and integrity. - Handling transaction reconciliation and settlement.
8	Bulk Payment	Facilitating the processing of large-scale payment transactions within the eNaira CBDC system	- Batch processing of payment transactions. - Bulk payment file validation and formatting. - Automated payment reconciliation and reporting. - Handling of payment exceptions and errors. - Scalability and performance optimization for processing high-volume transactions
9	Batch Wallet Creation	Streamlining the creation of multiple wallets within the eNaira CBDC system	- Bulk wallet creation based on predefined parameters. - Secure generation of wallet addresses and associated keys. - Wallet identity and attribute assignment. - Validation and verification of created wallets. - Integration with wallet management systems for seamless provisioning
10	Faster Settlement and Finality of Transactions within the Cross-Ecosystem	Accelerating the settlement and finality of transactions across the DCMS and eNaira systems	- Implementing optimized transaction validation and confirmation mechanisms. - Minimizing transaction settlement time and latency.

			- Ensuring transaction finality and irreversibility. - Coordinating and synchronizing transaction settlement between systems.

5. Information System Architecture

This section details the Information System Architecture that enables the Business. It contains a detailed description of the design for the application and data architecture. The specification for the information systems architecture is built on the basis of the business and functional needs. This involves outlining the functional coverage of the interactions between the applications and the data that they use and produce, as well as their interactions with users.

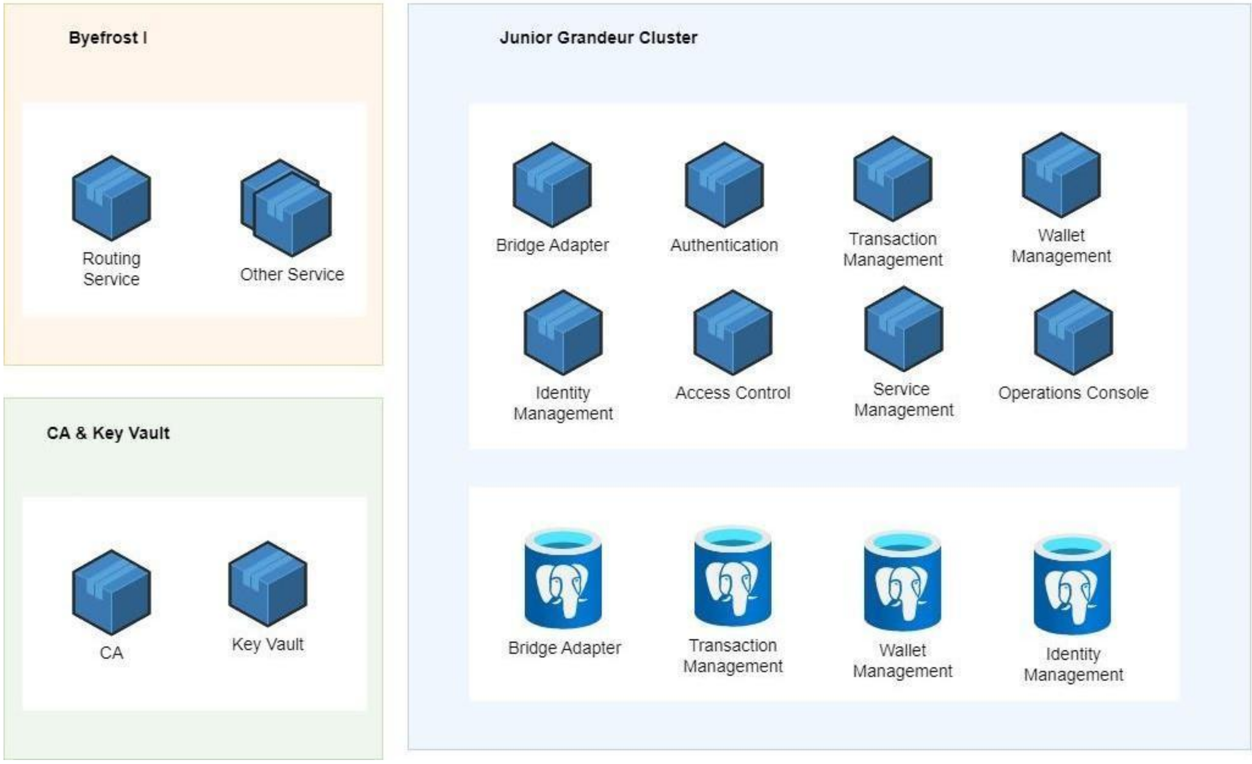


Figure 4. eNaira Information System Architecture

5.1 Application Services Components

The application service catalog identifies and maintain the list of all the applications service components in the system. It also contains the application component that shows micro services that represent an autonomous function of the Information System which can be configured and deployed independently.

Table 4 provides more information on the application service components.

S/No	Application Service Component	Description	**Data Types**
1.	Authentication	Responsible for verifying and authenticating the identity of users, devices, or services attempting to access the CBDC system. The application functions include but not limited: - User login, - Token generation, - Token validation, - Session management.	- User credentials, - Authentication tokens, - Biometric data (if used), - Access logs.
2.	Access Control	Managing user roles and permissions to control access to specific functionalities. The application functions include but not limited: - Role creation, - Privilege assignment, - Role-based access control	- Role definitions, - Privilege assignments, - User profiles.
3.	Service Account	Representing system entities or services with specific privileges to perform tasks on behalf of users or other services. The application functions include but not limited: - Service account creation, - permission management, - authentication.	- Service account credentials, - Permissions, - Associated user/ service information
4.	Bridge Adapter	Facilitating communication and data exchange between DCMS and JG infrastructure. The application functions include but not limited:	- Data format definitions, - Transaction records, - Adapter configurations

5.	Routing Logic	Determining the appropriate path for data and transactions within the system. The application functions include but not limited: - Route selection, - Load balancing, - Transaction routing	- Transaction details, - Network topology information
6.	Transaction Management	Handling the lifecycle of transactions within the JG system. The application functions include but not limited: - Transaction validation, - Balance enquiry - Processing, - confirmation, - balance eand - settlement	- Transaction details, - Status, - History, - Transaction IDs
7.	Wallet Management	Managing user wallets that hold eNaira balances and transaction records. The application functions include but not limited: - Wallet creation, - Balance updates, - Transaction recording	- Wallet IDs, - Balances, - Transaction history
8.	Sub Wallet Management	Allowing users or organizations to create sub-wallets within their main wallet for specific purposes. The application functions include but not limited: - Sub-wallet creation, - Balance updates, - transaction recording	- Sub-wallet IDs, - Balances, - Transaction history

9.	Identity Management	Managing user identities, including registration, verification, and authentication. The application functions include but not limited: - User registration, - Identity verification, - Account recovery	- User details, - Identity documents, - Verification status

10.	Wallet Tier Management	Categorizing wallets based on tier levels with different features and limits. The application functions include but not limited: - Wallet tier assignment, - Feature restrictions	- Wallet tier definitions, - Associated privileges
11.	Operation Console	Providing an administrative interface to manage and monitor the system. The application functions include but not limited: - Monitoring, - Configuration, - Performance analysis	- System status, - Logs, - Analytics data
12.	Multi-signature Wallet	Enabling wallets that require multiple signatures from authorized parties to authorize a transaction. The application functions include but not limited: - Multi-signature setup, - Transaction validation	- Wallet configuration, - Authorized signatories.

5.2 Entity Relationship Diagram

This shows a high-level representation of the data entities in eNaira and the relationships between them.

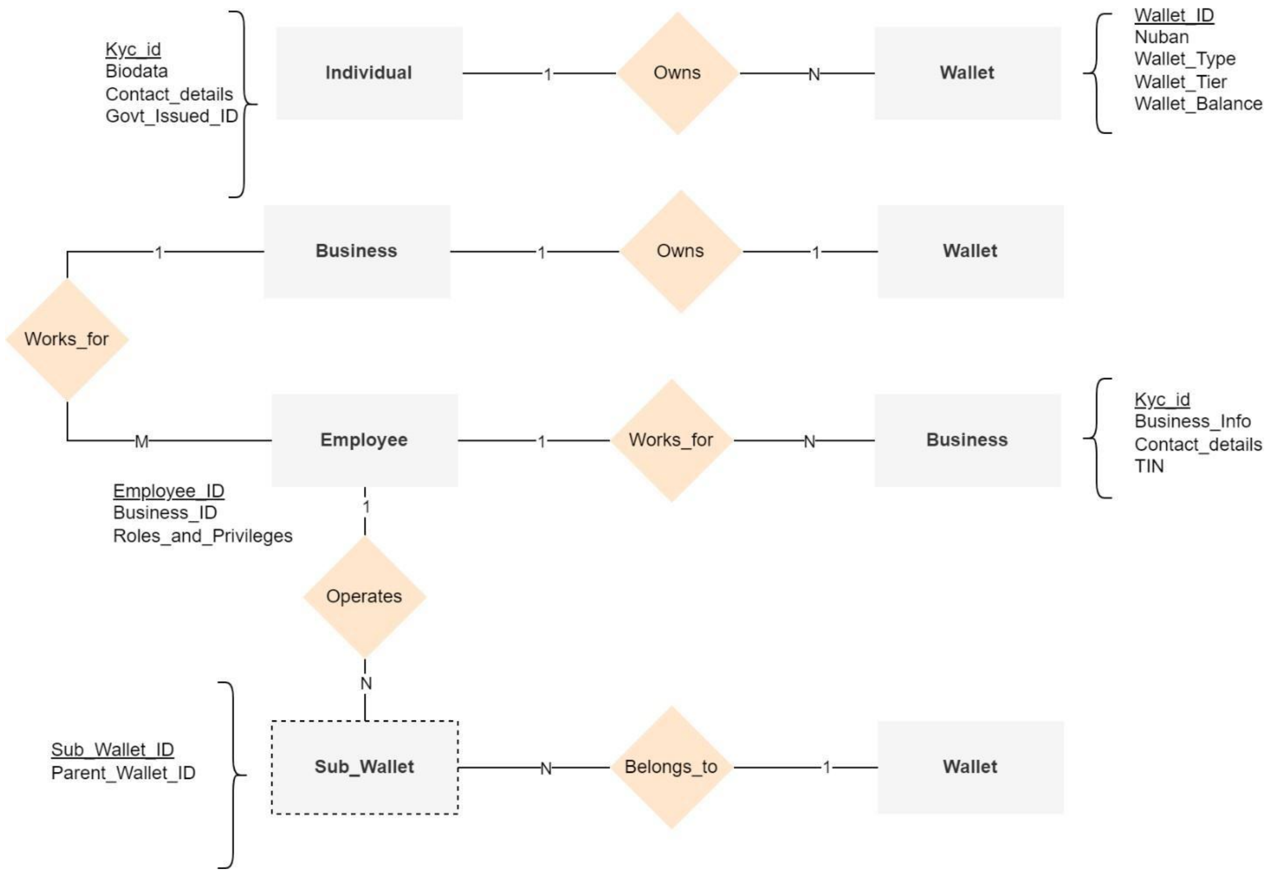
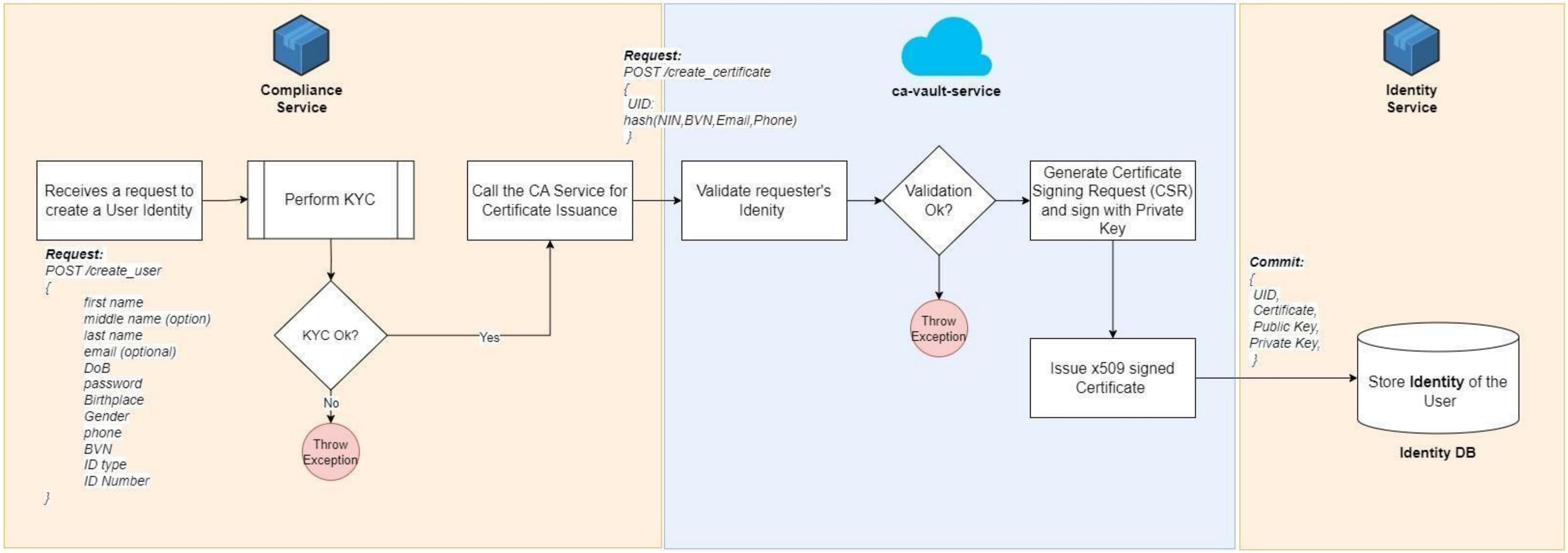


Figure 3. Entity Relationship Diagram

5.3 Process/System Realization

This section describes the sequence of events when multiple services are executing a business process. It shows the communication between the services.

JG Identity Creation & Certificate Issuance


Compliance Service
Receives a request to create a User Identity
Perform KYC
KYC Ok?
Yes
No
Throw Exception
Call the CA Service for Certificate Issuance
ca-vault-service
Validate requester's Idenity
Validation Ok?
Throw Exception
Generate Certificate Signing Request (CSR) and sign with Private Key
Issue x509 signed Certificate
Identity Service
Store **Identity** of the User
Identity DB

Request:

```
POST /create_user
{
        first name
        middle name (option)
        last name
        email (optional)
        DoB
        password
        Birthplace
        Gender
        phone
        BVN
        ID type
        ID Number
}
```

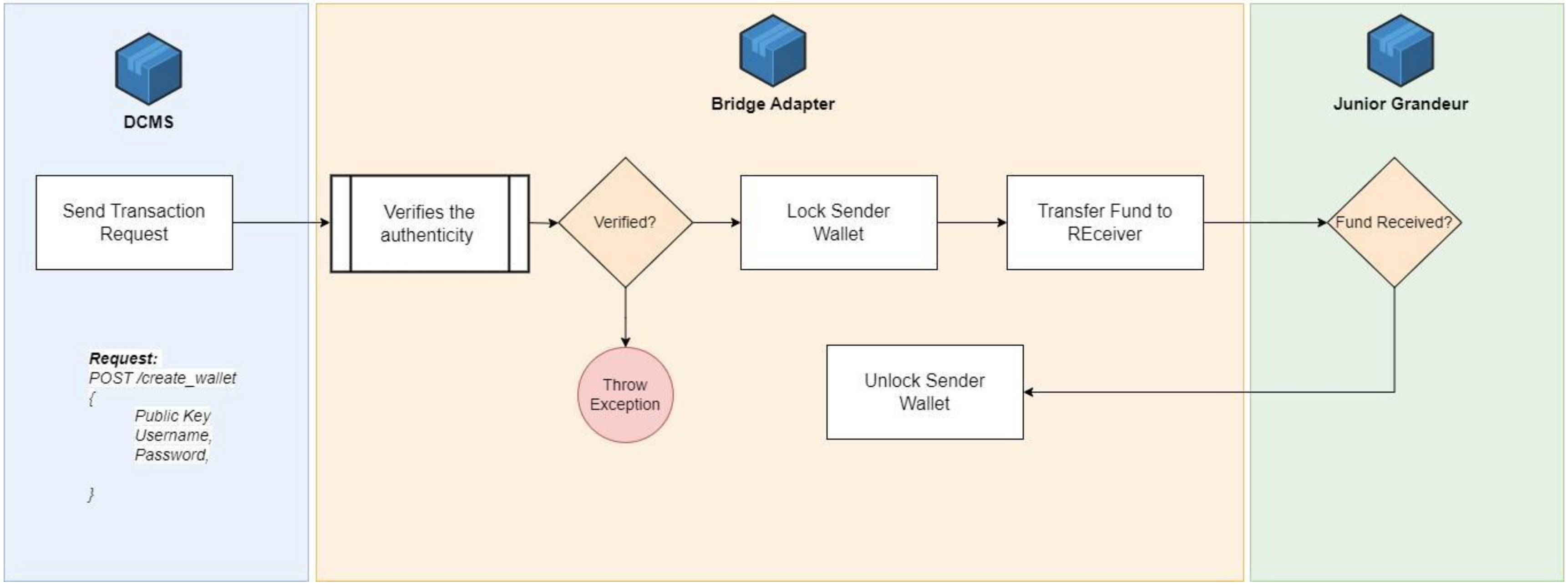
Request:

```
POST /create_certificate
{
 UID:
hash(NIN,BVN,Email,Phone)
}
```

Commit:

```
{
 UID,
 Certificate,
 Public Key,
Private Key,
 }
```

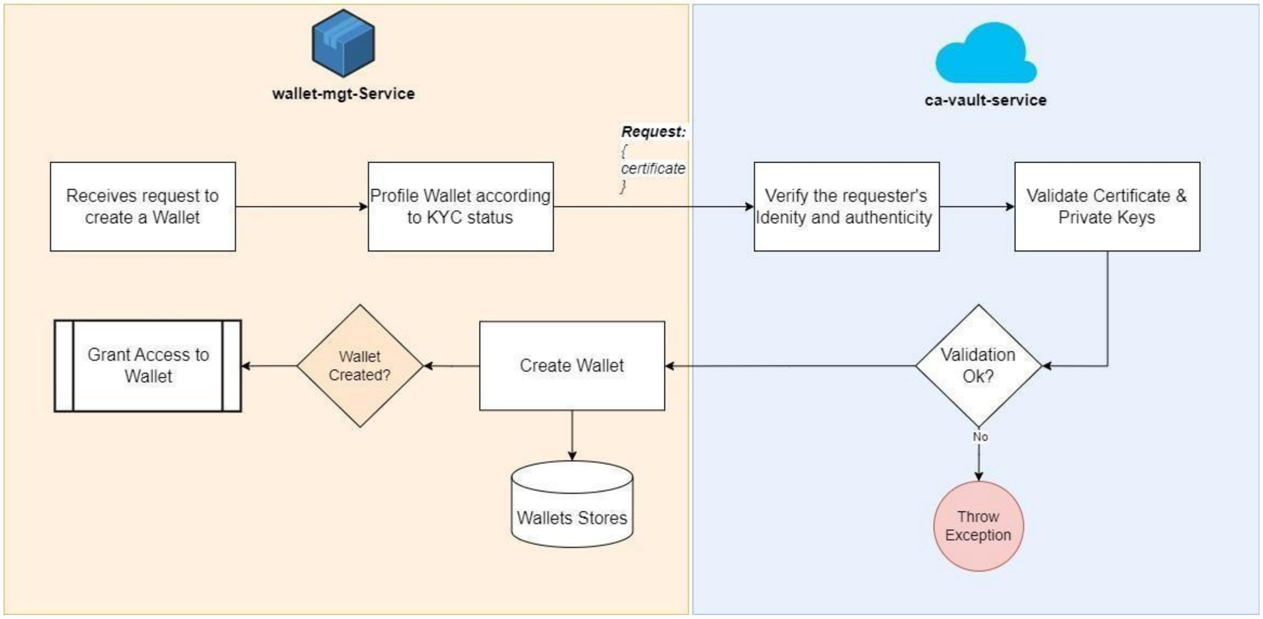
Bridge Adapter



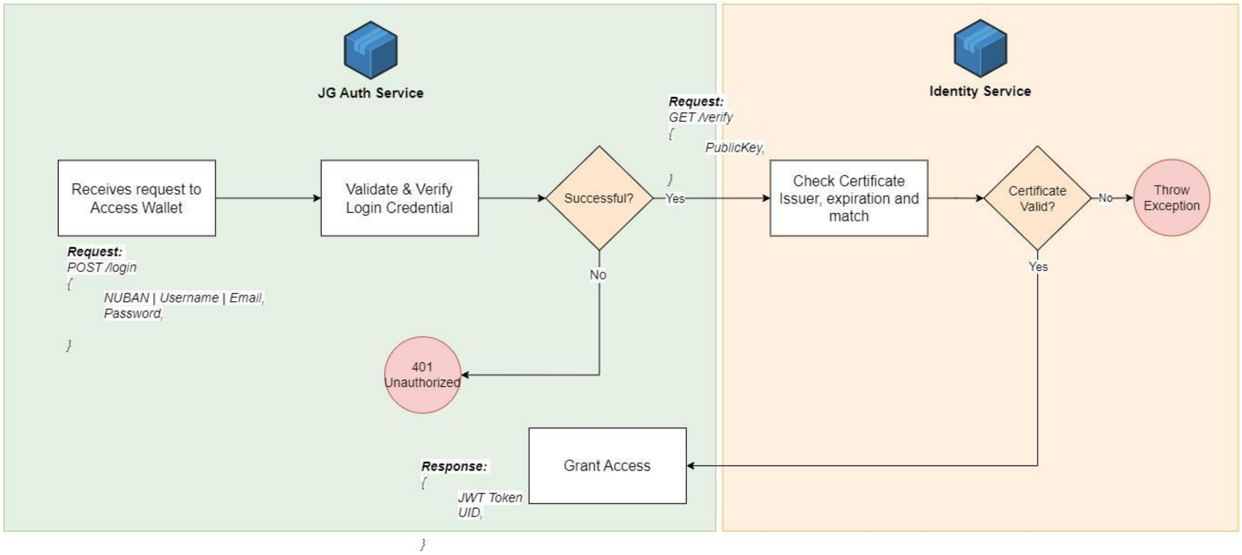
```
Request:
POST /create_wallet
{
        Public Key
        Username,
        Password,

}
```

JG Wallet Creation


wallet-mgt-Service
ca-vault-service
Receives request to create a Wallet
Profile Wallet according to KYC status
Request:
{
certificate
}
Verify the requester's Idenity and authenticity
Validate Certificate & Private Keys
Validation Ok?
No
Throw Exception
Create Wallet
Wallets Stores
Wallet Created?
Grant Access to Wallet

Access Management Process Flow



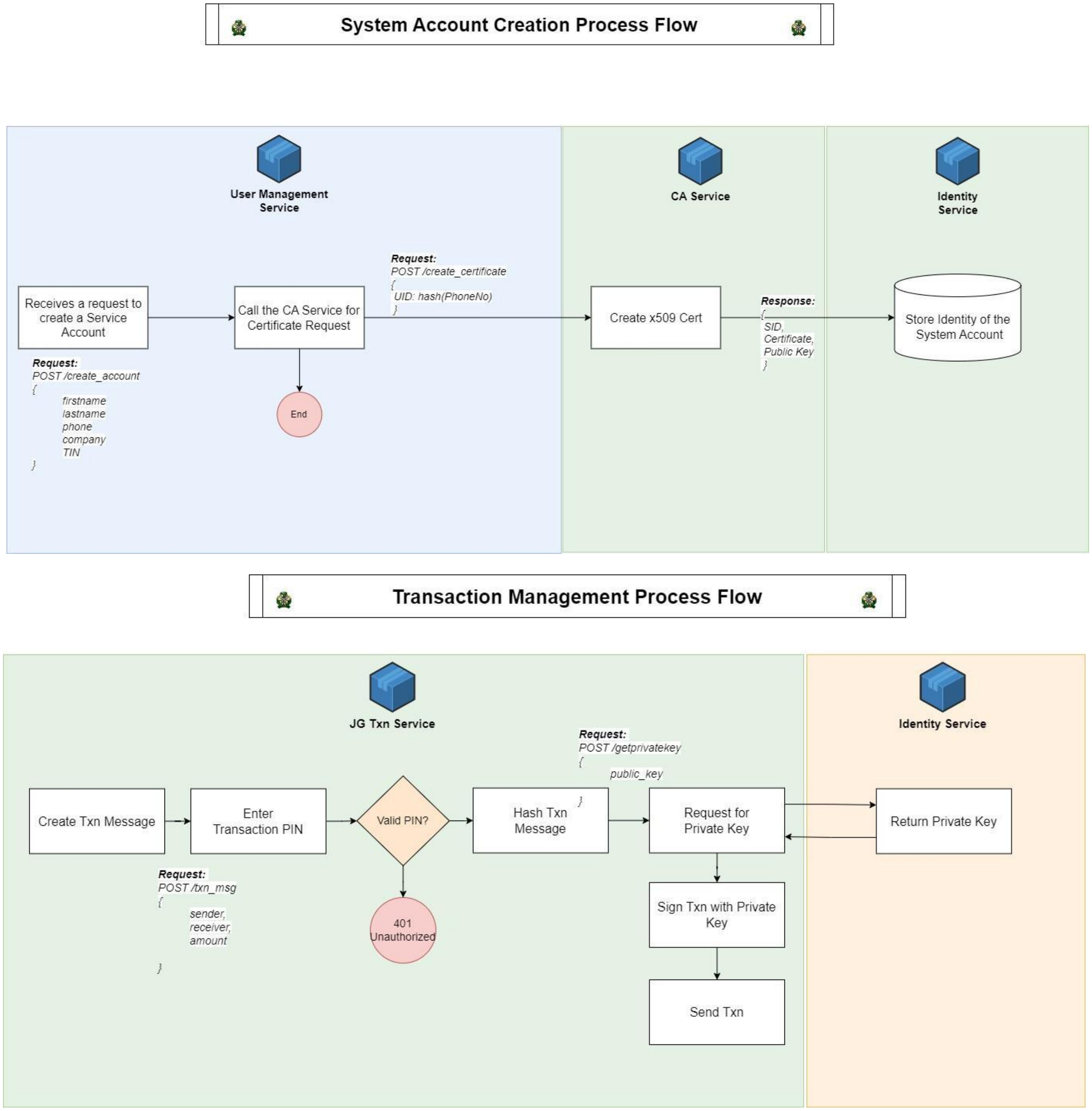


Figure 4. Process/System Realization for eNaira

5.4 Integration Architecture

The integration architecture shows the list of applications and technologies and how they communicate and share data.

Table 4 shows the Integration requirement for the system.

S/No	Requirement	Protocol	Source Format	Target Format	Connectivity	Frequency
1	The system shall be able to allow clients to request and obtain certificates from the CA securely (Certificate Enrollment)	Certificate Management Protocol (CMP)	JSON	X509	Uni-directional	Medium
2	The system shall be able to allow certificate enrollment for network device	Simple Certificate Enrollment Protocol (SCEP)	JSON	X509	Uni-directional	Low
3	The system shall be able to perform real-time checking of the revocation status of a certificate	Online Certificate Status Protocol (OCSP)	JSON		Uni-directional	Low
4	The system shall be able to provide secure communication between clients and the Key Vault service	HTTPS	JSON	JSON	Uni-directional	High
5	The system shall be integrated into the Security Information and Event Management (SIEM) of the Bank by exposing audit trail tables for analysis.	Sylog	Sylog Message Format	Sylog Message Format	Uni-directional	Medium
6	The system shall integrate with the Bank's Identity and Access Management System	Security Assertion Markup Language (SAML)	XML	XML	Uni-directional	Medium

6. Security Architecture

The security architecture highlights how the infrastructure, application and its data will be protected from unauthorized access from inside and outside of the Bank—either at rest (in database management systems or tape drives), or in motion (transiting through network/communication devices).

To meet the mentioned security requirements for eNaira, Table 5 shows the security implementations to be considered for the system.

S/No	Security Requirement	Implementation
1	Identity and Access Management	● Implement a robust identity and access management system to authenticate and authorize users. ● Utilize strong authentication methods such as two- factor authentication (2FA) for secure user verification. ● Integrate with the Bank's Identity and Access Management System for centralized user management.
2	Key Management	● Use a secure key management system to generate, store, and rotate cryptographic keys. ● Implement Elliptic Curve Cryptography (ECC) for key generation to ensure secure communication. ● Encrypt the private keys using node-RSA before storing them in a key vault. ● Use the user's email address or phone number as a universal unique identifier (UUID) to retrieve keys from the key vault. ● Provide a decryption system to decrypt the retrieved public/private key pairs.
3	Data Privacy	● Ensure sensitive data, including Personally Identifiable Information (PII), is encrypted at rest. ● Avoid storing passwords in plain text and hash them using a strong hashing algorithm. ● Implement secure session mechanisms to protect data in transit and prevent unauthorized access. ● Utilize secure end-to-end encryption (SSL) for exchanging sensitive transaction data over the internet
4	Secure Communication	● Support secure communication protocols such as Transport Layer Security (TLS) for authenticated connections. ● Authenticate connections to external systems involving sensitive information or functions. ● Avoid disclosing session IDs in URLs, error messages, or logs.
5	Smart Contract Security	● Perform thorough code reviews and security audits of smart contracts to identify and mitigate vulnerabilities.

		• Implement best practices for secure coding, including input validation and protection against common attacks (e.g., injection attacks)
6	Transaction Endorsement	• Incorporate mechanisms for detailed logging of transactions, including sequential numbers, timestamps, and parameterization changes. • Implement secure mechanisms to monitor, track, and restrict access to sensitive transaction data and critical resources. • Integrate with the Bank's Security Information and Event Management (SIEM) for audit trail analysis.
7	Certificate Authority (CA) System	• Implement a CA system to issue and rotate certificates for secure authentication and communication. • Store the certificates in a trusted key vault, such as the Open Source HashiCorp Vault.
8	Role-Based Access Control	• Implement role-based access control (RBAC) to ensure that users have appropriate access privileges based on their roles. • Define and enforce granular access control policies to protect critical resources.
9	Secure Integration	• Ensure secure integration of third-party components by validating their integrity and free from vulnerabilities. • Avoid storing secrets, API keys, and passwords in source code or online repositories.
10	Logging and Monitoring	• Establish a centralized logging mechanism to capture and analyze security-relevant events. • Set configurable log retention periods and integrate with SIEM for advanced analysis and monitoring.

6.1 Certificate Authority (CA) Service

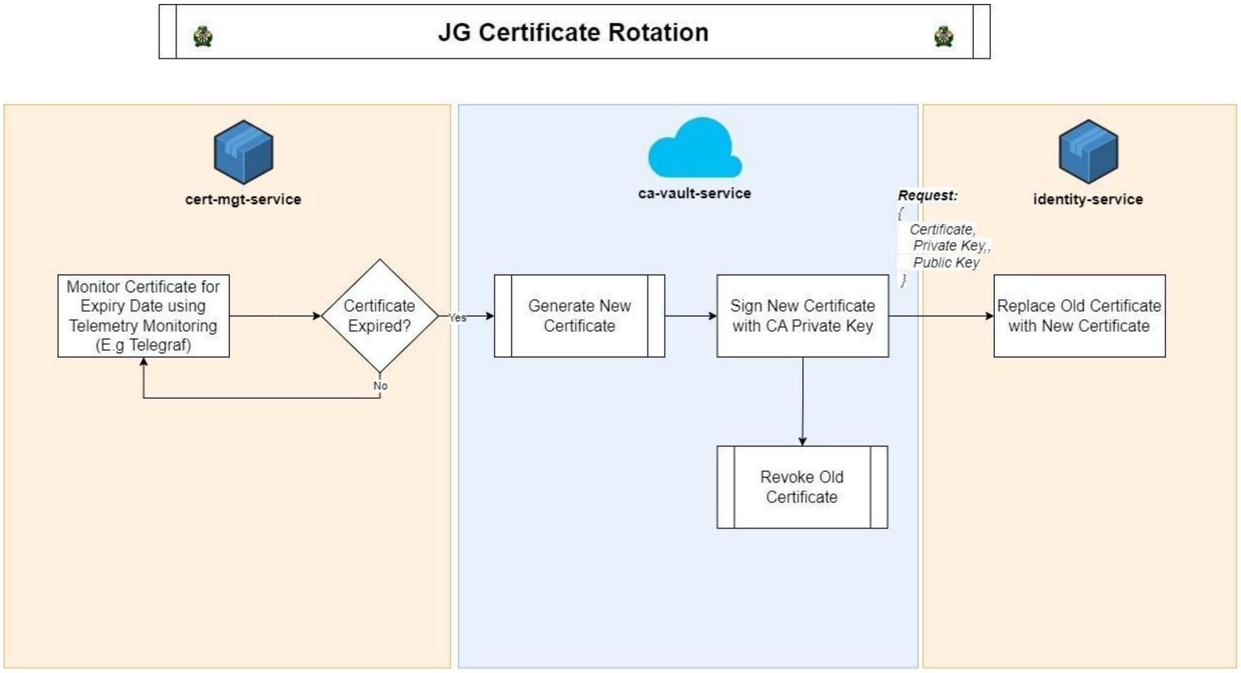
The Certificate Authority (CA) service is a crucial component of the system, responsible for issuing and managing digital certificates. The CA service plays a vital role in establishing trust, enabling secure communication, and verifying the identities of users and applications within the CBDC ecosystem.

The CA service will perform the following functions:

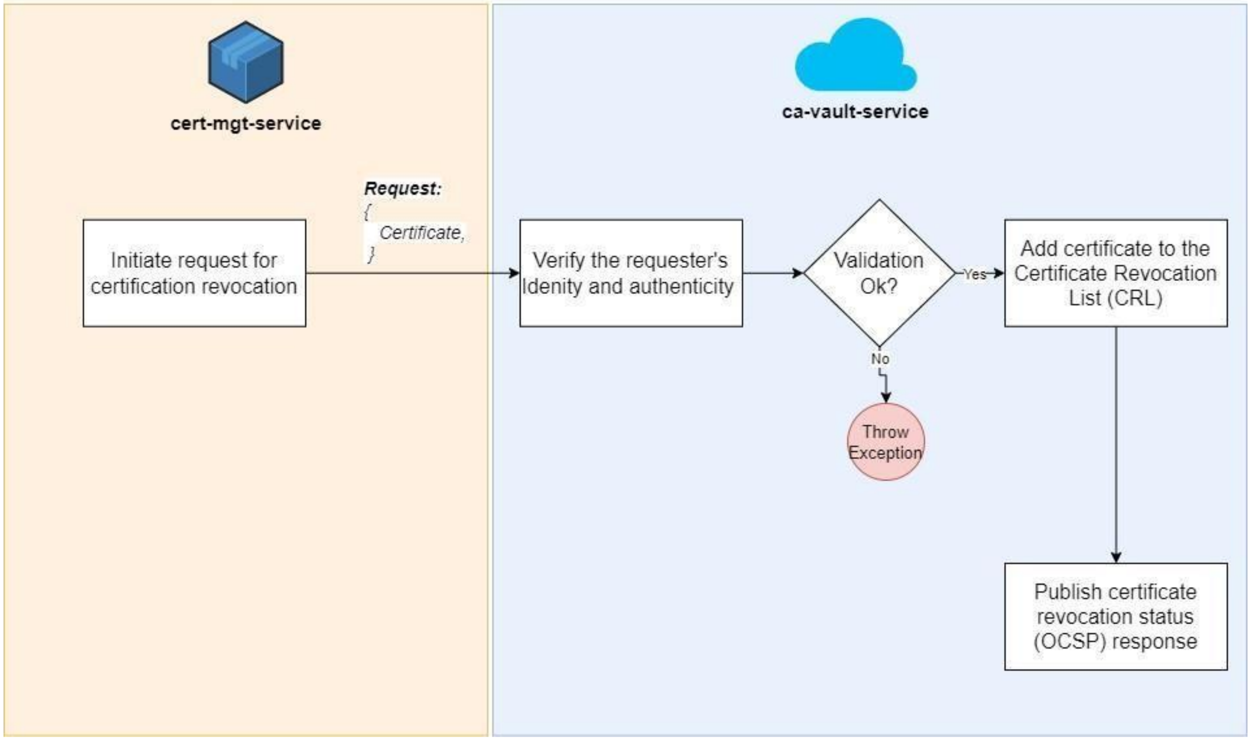
1. Certificate Issuance: The primary functionality of the CA service is the issuance of digital certificates. These certificates bind a user or application's identity to a public key, serving as a digital credential that can be used for authentication and secure communication. The CA service generates and signs these certificates, attesting to the validity and trustworthiness of the associated identities.
2. Public Key Infrastructure (PKI) Management: The CA service manages the Public Key Infrastructure, which includes the generation, storage, and revocation of cryptographic key pairs. It ensures that public keys are securely associated with

the corresponding identities and maintains a repository of revoked certificates to prevent the use of compromised or invalid credentials.

3. Certificate Revocation: In the event of a compromised private key or a change in user or application status, the CA service facilitates the revocation of digital certificates. Revocation ensures that any certificates associated with unauthorized or invalid entities are invalidated, preventing their misuse and maintaining the integrity of the CBDC system.
4. Certificate Validation and Trust Verification: The CA service provides mechanisms for validating and verifying the authenticity and integrity of digital certificates. It enables users and applications to verify the trustworthiness of certificates by validating the CA's digital signature and checking the revocation status against the repository of revoked certificates. This validation process ensures that only valid and trusted identities are accepted within the CBDC system.



JG Certificate Revocation


cert-mgt-service
ca-vault-service
Request:
{
Certificate,
}
Initiate request for certification revocation
Verify the requester's Idenity and authenticity
Validation Ok?
Yes
No
Add certificate to the Certificate Revocation List (CRL)
Throw Exception
Publish certificate revocation status (OCSP) response

6.2 Key Vault Service

The Key Vault is a critical component within the system that ensures the secure storage and management of cryptographic keys used for user and application authentication, encryption, and digital signatures.

The Key Vault is designed using a robust and secure architecture that prioritizes the confidentiality, integrity, and availability of cryptographic keys.

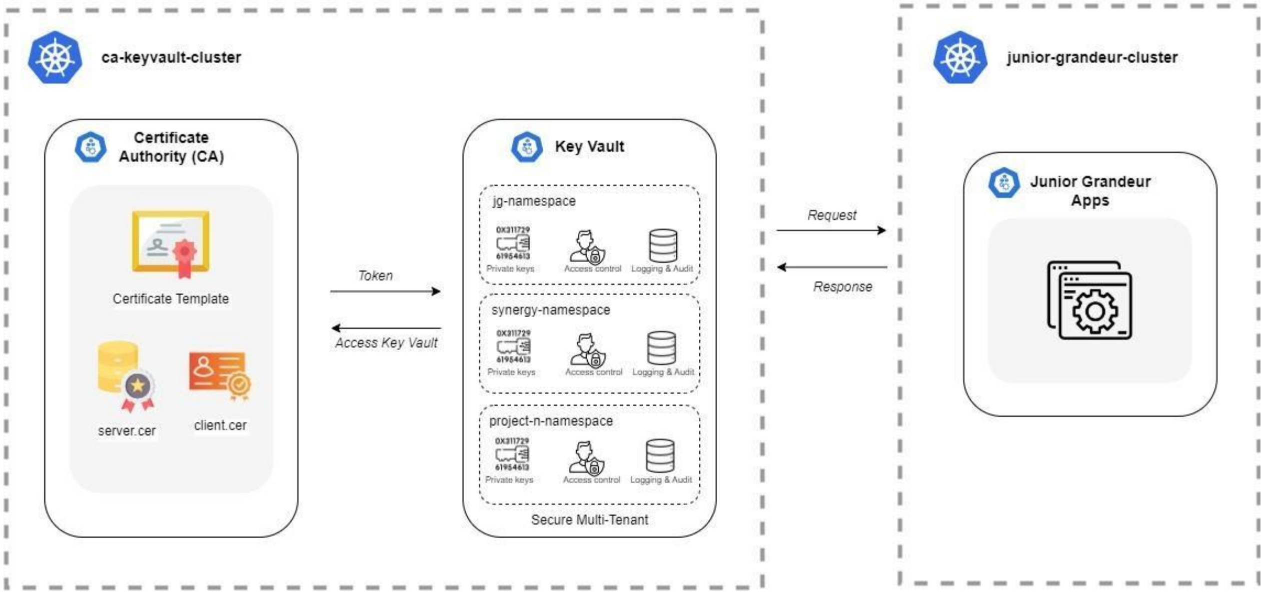


Figure 5. CA & key Vault Architecture for eNaira

Key Vault architecture include:

1. Backend Storage: The Key Vault relies on a highly secure and scalable backend storage system for storing cryptographic keys. This storage can be implemented using various technologies, such as hardware security modules (HSMs), secure key management systems, or cloud-based key management services. The choice of backend storage depends on the specific security requirements and infrastructure of the eNaira CBDC system.
2. Access Control: To ensure the protection of cryptographic keys, the Key Vault incorporates a robust access control mechanism. This mechanism employs authentication and authorization protocols to authenticate users or applications attempting to access the keys and enforce granular access permissions based on roles or privileges. It may utilize secure protocols like OAuth, PKI (Public Key Infrastructure), or other industry-standard authentication mechanisms.
3. Encryption and Key Wrapping: The Key Vault employs strong encryption techniques to safeguard the stored keys. Encryption ensures that even if unauthorized access is gained to the storage system, the keys remain protected. Additionally, key wrapping techniques are applied to securely manage and distribute keys within the Key Vault.
4. Logging and Auditing: Comprehensive logging and auditing mechanisms are integrated into the Key Vault to track and monitor key management activities. These mechanisms help detect any suspicious or unauthorized access attempts,

provide an audit trail for compliance purposes, and facilitate forensic analysis in the event of a security incident.

5. The architecture and implementation of the Key Vault can be extended to other projects requiring secure key management.

7. Technology Architecture

This chapter describes the development of the Technology Architecture that enables the Information System. The technology architecture of the system encompasses the layers, components, and technologies employed to support its functionality. Understanding the technology architecture is essential for developers, and stakeholders involved in system analysis, design, and optimization.

7.1 Technology Standards Catalog

This section documents the agreed standards for technology across the enterprise covering technologies, and versions, the technology lifecycles, and the refresh cycles for the technology.

Table 6 contains the URL to CBN IT Standards.

Title	**ITD Standards**
URL	IT Standards 2022 - Lists (sharepoint.com)

7.2 Cloud Architecture

The cloud architecture is designed with two Kubernetes clusters to support the deployment and management of services. One of the clusters is dedicated to running the "eNaira" services, which are hosted within a scale set consisting of multiple virtual machines (VMs). This setup allows for seamless scaling of the services based on demand. Additionally, the PostgreSQL database is integrated with the Kubernetes cluster using Pgpool, which serves as a connection pooler and load balancer to optimize database performance.

On the other Kubernetes cluster, the focus is on running the Certificate Authority (CA) services and Key Vault. This cluster specializes in handling security-related functions, such as issuing and managing digital certificates through the CA, and securely storing and accessing cryptographic keys and secrets using the Key Vault.

By employing this architecture, the system can ensure a scalable and resilient infrastructure for the services while maintaining a separate and dedicated cluster for critical security components like the CA and Key Vault. Kubernetes enables efficient container orchestration, and the cloud-native environment allows for easy scaling, monitoring, and management of resources to meet the dynamic requirements of the services and security components.

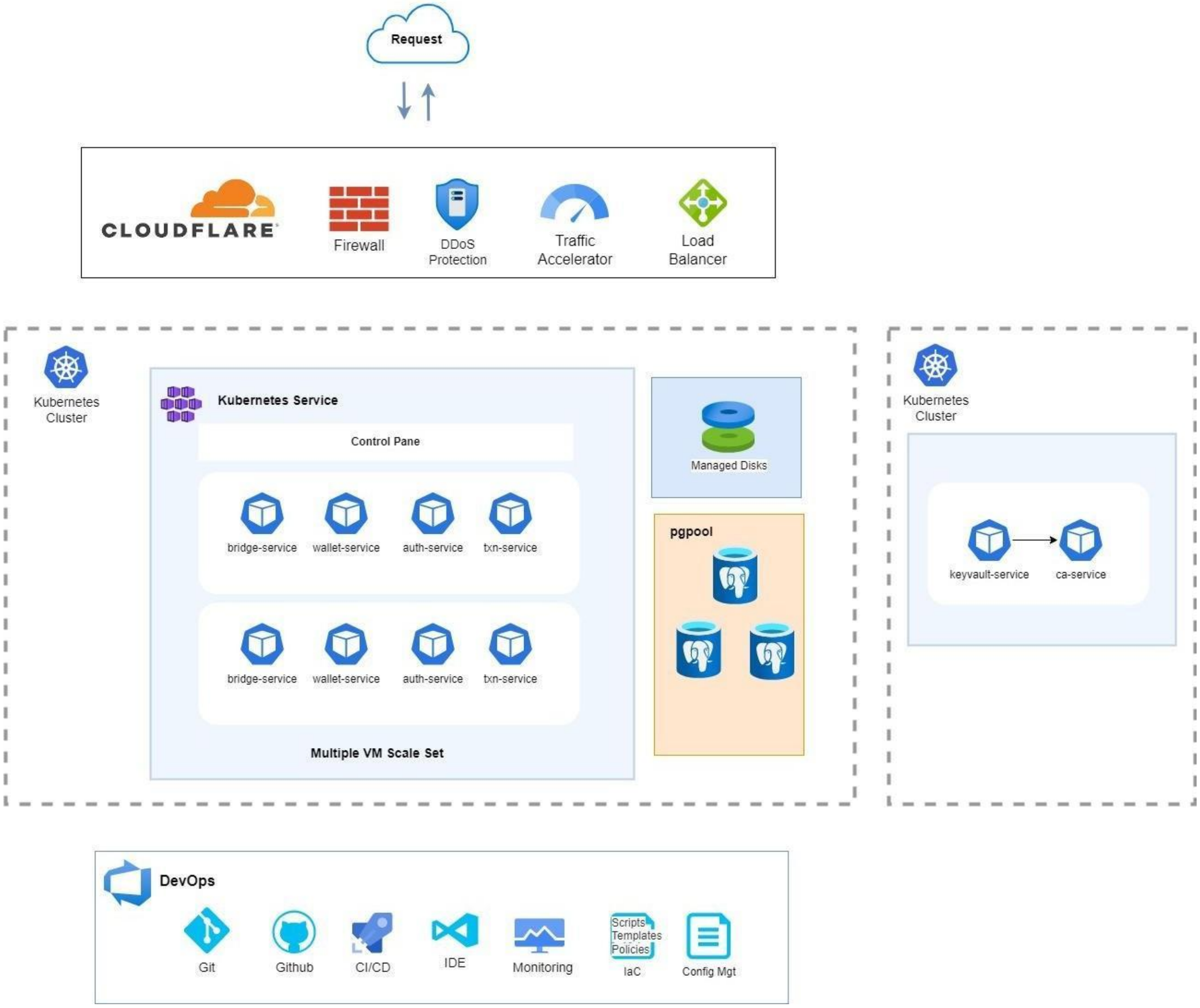


Figure 6. Cloud Architecture for eNaira

7.3 Infrastructure Automation

Infrastructure automation will reduce manual intervention, improve efficiency, increase scalability, and enhance the overall agility of the IT infrastructure. The components are listed below:

1. **Infrastructure as Code (IaC):** Infrastructure as Code is used to represent the desired infrastructure configuration in a declarative format using code. The code will be version-controlled, tested, and deployed like any other software.
2. **Configuration Management:** Configuration management tools automate the process of configuring and maintaining the state of servers and other infrastructure components. These tools help in managing software installations, applying security updates, and enforcing desired system configurations.

3. **Orchestration and Provisioning:** Orchestration tools enable the automation and coordination of multiple tasks and processes in a structured manner. They facilitate the provisioning of resources, such as virtual machines, containers, and storage, across different cloud providers or on-premises infrastructure.
4. **Continuous Integration and Continuous Deployment (CI/CD):** CI/CD practices automate the software build, test, and deployment processes. By integrating infrastructure automation with CI/CD pipelines, teams can automatically deploy applications to different environments, such as development, testing, staging, and production, while ensuring consistency and reliability.
5. **Monitoring and Logging:** Infrastructure automation include robust monitoring and logging capabilities to provide visibility into the health and performance of the infrastructure. Monitoring tools collect metrics and alerts on system resources, application performance, and security events. Logging tools centralize and analyze logs from various sources, enabling troubleshooting and performance analysis.

7.4 Deployment Environments

This section shows the configuration and setup where the system is deployed, tested, and accessed. It represents a distinct infrastructure and set of resources that facilitate the development, testing, and deployment of Byefrost.

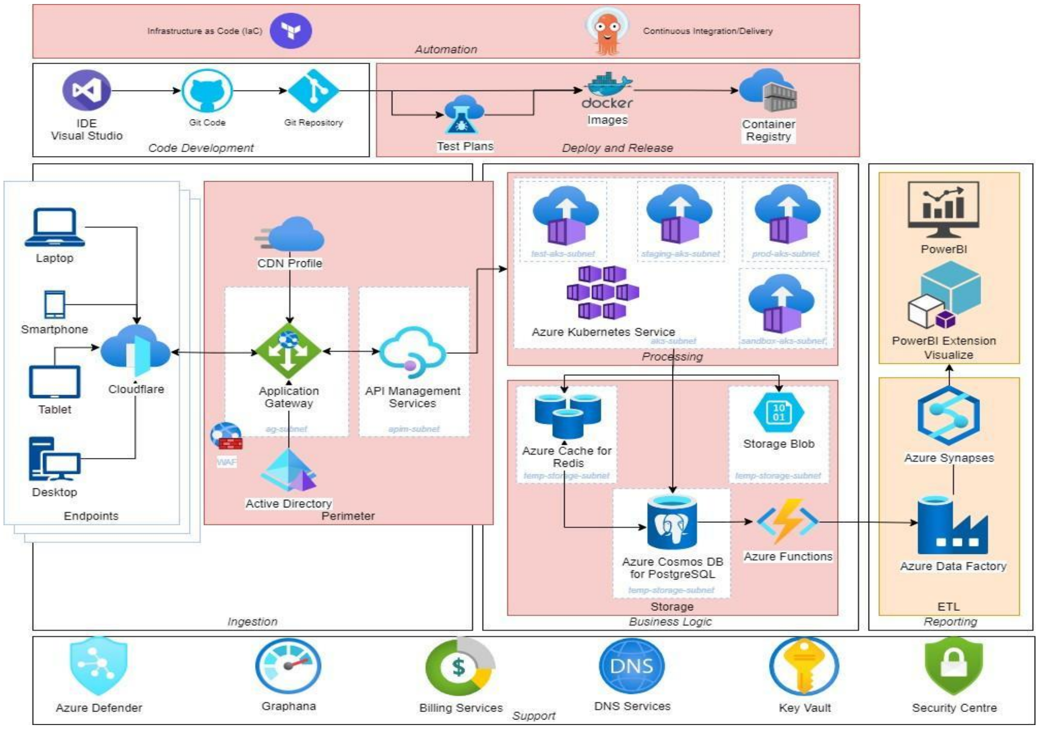
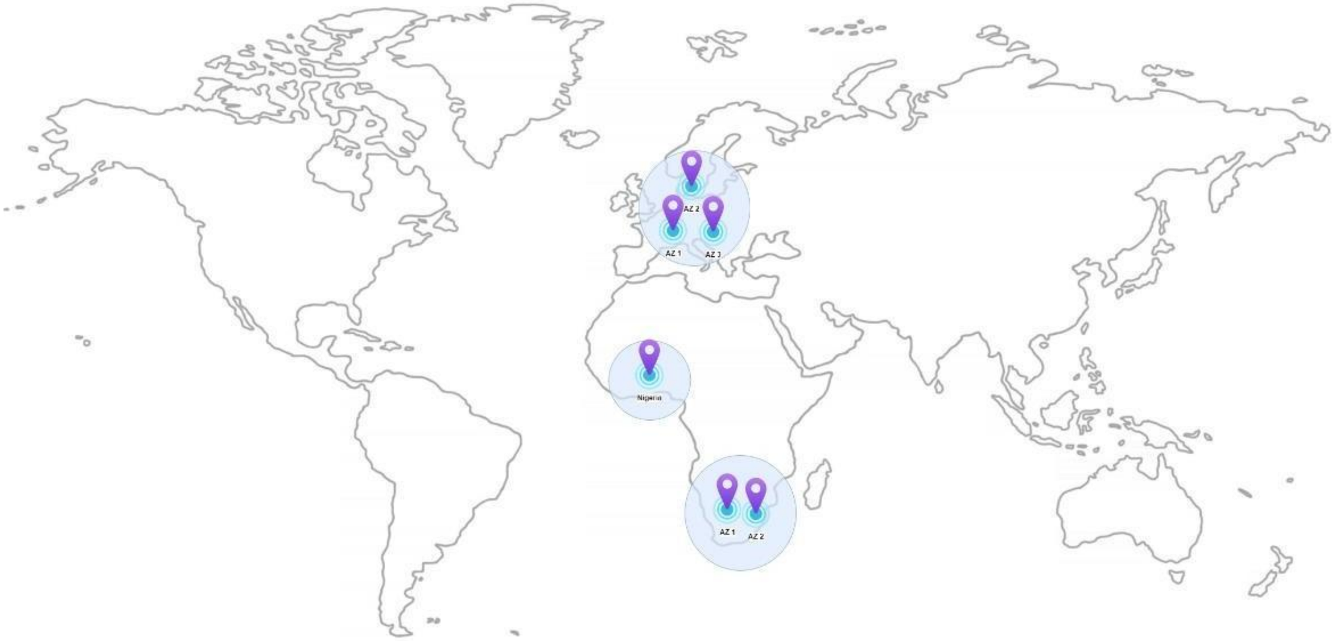


Figure 7. Deployment Environment Diagram

7.5 Environments and Locations Diagram

This shows the existence and location of different deployment environments. It depicts the various locations for redundancy.



Geo-replication;
Composite uptime: 99.999%

Figure 8. Environment & Location Diagram

8. Solution Options and Recommendations

The development of eNaira cannot be overemphasized. However, it is important that the decision is based on empirical evidence evaluating the potential benefits to efficiently and effectively carry out this task. We found two solution options to be the most appropriate path to follow in meeting the needs of the stakeholders.

8.1 Option 1 – In-house Development of the System

The application and infrastructure can be developed in-house. A wide array of benefits of reduced cost, increased time to delivery, application rationalization as it will be leveraging on an existing platform and in-house expertise. This will allow for easy scalability of the system to allow for future growth and functionality.

8.2 Option 2 – Co-development between CBN and Vendor

The Bank has the option to outsource the development of the application and infrastructure provisioning to third parties. However, for an easy handover, it is important to have a co-development between in-house developers and vendors.

8.3 Solution Options Risk Assessment

Table 8 shows the risk assessment table for the solution options

Solution Risk Assessment	**Option 1 (In-house Development of the System)**		**Option 2 (Co-development between CBN and Vendor)**	
	Probability	Impact	Probability	Impact
Risk 1 – Delivered without required functionalities	Low	High	Low	High
Risk 1 General Mitigation Strategy	Gather full functional requirements and design detail physical solution architecture. Get user involvement early in the project		Gather full functional requirements and provide matching evaluation criteria	
Risk 2 – Delivered outside of specified timeline	High	High	Low	High

Risk 2 General Mitigation Strategy	Get user involvement early in the project		Manage time taken to secure Management approval	
Risk 3 – Scalability and Responsiveness to change	High	High	Low	High
Risk 3 General Mitigation Strategy	Manage change request within stipulated timeline		Manage change request for additional module within timeline	
Risk 4 – Lack of adoption by users	Low	Low	Low	Low
Risk 4 General Mitigation Strategy	Effective change enablement		Effective change enablement	
Overall Risk Rating	**HIGH**		**LOW**	

8.4 Analysis of Sourcing Options

Based on the considerations made in the previous sections, a decision must be made on the best sourcing option for delivering the ARMS. Thus, a Decision Tree for analyzing "Build, Buy or outsource" option is established as shown in table 7 below. Scores of High
(H) = 3; Medium (M) = 2; Low (L) = 1 are assigned to relevant criteria.

Table 9 shows the decision tree for analyzing "Build, Buy or Outsource"

SN	DECISION CRITERIA	BUILD	BUY	OUT-SOURCE	REMARKS
1	Expertise	H=3	M=2	H=3	Build and out-source have the capabilities to meet the requirements (functional/nonfunctional)
2	Knowledge Area	H=3	H=3	H=3	Products in the market doesn't have capabilities to meet the functional requirements

3	Cost	H=3	L=1	M=2	In house development has a higher cost savings since we are reusing existing bank resources
4	Delivery Time	M=2	M=2	M=2	“Buy” will have the least time to deliver as the products are off the shelf
5	Technical/Overall Support	M=3	H=3	H=3	The three options will have very good support
6	Architectural Standardization	H=3	H=3	H=3	The three options will conform to the Bank’s standards
7	Reliability and High Availability	H=3	H=3	H=3	Any of the three option when delivered is expected to be “Reliable and highly available”
SCORE ANALYSIS					
	Total Score	20/21	17/21	19/21	From the score, the recommended Option is ***“BUILD & OUT-SOURCE”***

8.5 Recommendation

Based on the matrix on Solution Options in Table 8 above, the option to co-develop the eNaira system with a Vendor is hereby recommended. This is based on the benefits highlighted in the aforementioned table.

During the development phase, it is essential that the development team strictly adhere to the Solution Architecture:

1. Follow best practices to ensure the confidentiality, integrity, and availability of the system.
2. Adopt a modular and scalable design approach to facilitate future enhancements and accommodate increasing user demands. Use industry-standard design patterns and principles to ensure maintainability and extensibility of the system as defined in the IT Standards.
3. Implement performance optimization techniques, such as caching, load balancing, and efficient database management, to ensure optimal system performance even under high transaction volumes.
4. Implement a comprehensive testing and quality assurance process to identify and fix issues early in the development lifecycle. Perform unit testing, integration testing, and system testing to ensure the stability and reliability of eNaira.

5. Develop detailed technical documentation and provide comprehensive training materials to facilitate the onboarding of developers, system administrators, and end-users. This ensures a smooth transition and effective utilization of the eNaira system.
6. Establish a robust monitoring and maintenance process to proactively identify and address any issues or vulnerabilities.