IT System Architecture and Components

1. Describe the overall architecture (include a diagram) of the IT system supporting the remittance services.

- Include details such as whether it is client-server, cloud-based, or a hybrid model.

Name of the IT system/software:

Version and date of deployment:

Is the system developed in-house or by a third party?

If third-party, provide vendor details:

1. Overall Architecture Overview

Model: cloud-based application.

Key Components:

Frontend (Mobile App):

- Built using Flutter, providing a cross-platform experience for Android and iOS.
- Manages user authentication, onboarding, transaction processing, and KYC submissions.
- Communicates with the backend via RESTful APIs and WebSockets for real-time updates.
- Integrates with third-party services like Volume (for payment) and Sumsub (for identity verification).

Backend:

- Developed in Laravel (PHP), serving as the core processing engine.
- Implements a modular service structure for handling payments, compliance, transaction workflows, and user management.
- Uses event-driven processing with AWS SQS to handle asynchronous tasks like transaction approvals, risk evaluations, and payout processing.
- Payment Processing: Volume Pay is integrated to process payments and settlements.

Database:

- PostgreSQL serves as the primary relational database, ensuring ACID-compliant transactions.
- Uses UUIDs for primary keys to ensure global uniqueness across all records.
- Redis is used for caching frequently accessed data to optimize performance.

Identity Verification & Compliance:

Sumsub handles KYC and AML checks, including:

- Document verification (passports, ID cards, bank statements).
- Face match and liveness detection.
- Risk scoring based on document authenticity and user history.

Rule-Based AML System:

- Configurable policies, limits, and risk thresholds based on transaction behavior.
- Automated transaction monitoring to flag suspicious activities.

Messaging & Orchestration:

- AWS SQS (Simple Queue Service) for event-driven processing, ensuring scalable and fault-tolerant transaction handling.
- AWS SNS (Simple Notification Service) for real-time notifications and alerts.
- AWS SES (Simple Email Service) for secure email communication, including transaction confirmations and regulatory notices.
-

Infrastructure & Deployment:

- Fully hosted on AWS, leveraging managed services for scalability and security.
- AWS Lambda is used for processing lightweight tasks asynchronously.
- S3 storage for document uploads (e.g., KYC documents, receipts, compliance reports).
- Laravel Vapor used for infrastructure automation and deployment management.

2. Deployment Information

- Name of the IT System/Software: Red Sea Money Transfer Application
- Version and Date of Deployment: Version 1.0.0, initially deployed in 2025
- Development: Fully in-house

2. What are the main software components and technologies used in the system?

- Specify databases, application servers, APIs, or other critical technologies.
- What are the key components (e.g., payment engine, risk engine, customer database).

- Network layout and hosting environment (on-premise / cloud)

1. Core Technologies & Software Stack

- Frontend: Flutter (mobile app for iOS and Android)
- Backend: Laravel (PHP framework)
- Database: PostgreSQL (relational database)
- Caching: Redis (for session management and faster queries)
- Messaging & Queues: AWS SQS, SNS, and SES
- Identity Verification: Sumsub (for KYC, liveness detection, and document verification)
- Payment Processing: Volume Pay (primary payment processor)
- Hosting & Infrastructure: AWS Cloud (with managed services)
- Storage: AWS S3 (for storing KYC documents, transaction receipts, etc.)

2. Key System Components

Transaction Engine:

- Manages money transfers, currency conversions, and payment processing via Volume Pay.
- Supports multi-currency transactions and exchange rate management.

Risk & Compliance Engine:

- Implements rule-based AML monitoring and fraud detection.
- Tracks transactions, customer behaviors, and flags suspicious activity.

KYC & Identity Verification:

- Integrates Sumsub to handle document verification, liveness detection, and compliance screening.

Access Control (RBAC + ABAC):

- Uses RBAC (Role-Based Access Control) for standard user roles (e.g., Admin, Agent, Customer).
- Implements ABAC (Attribute-Based Access Control) to restrict access based on transaction type, country, and risk level.

User Management & Authentication:

- Secure authentication and authorization using JWT tokens and session cookies.
- Customizable permissions per user role and attribute policies.

Notifications & Alerts:

- Uses AWS SNS for push notifications and alerts.
- AWS SES for email confirmations, receipts, and security notifications.

Customer Database & Ledger System:

- PostgreSQL stores customer profiles, transaction history, and account balances.
- A double-entry accounting system ensures financial consistency.

API Gateway & Integrations:

- Provides RESTful APIs for third-party integrations (e.g., banking partners, compliance services).
- Supports webhooks for real-time transaction updates.

3. Network Layout & Hosting Environment

Hosting: Fully cloud-based on AWS.

Application Layer:

- Laravel backend hosted on Lambda (Serverless) for scalability.

Database Layer:

- AWS RDS (PostgreSQL) for structured data storage.
- Redis used for caching frequently accessed data.

Storage Layer:

- AWS S3 for storing user documents, logs, and receipts.

Security & Access Control:

- Encrypted database connections using TLS 1.3.
- IAM roles and security groups for access control.
- RBAC and ABAC policies ensure granular access control.
- WAF (Web Application Firewall) and DDoS protection via AWS Shield.

3. How does the system integrate with external entities (e.g., banks, payment processors and payout partners)?

- Detail the methods and technologies used for integration.
- How are **authorised external connections** (e.g., APIs, partners) secured?

1. Integration with External Entities

The system integrates with various external entities such as banks, payment processors, and payout partners through secure API connections and event-driven workflows.

External Entity	Integration Method	Purpose
Payment Processors (i.e., Volume Pay)	RESTful APIs	Payment processing, settlements, refunds.
Banks & Payout Partners	Secure API & Webhooks	Bank transfers, deposits, and payout processing.
KYC & Compliance (e.g., Sumsub)	API integration with real-time verification	Identity checks, document validation, AML compliance.

2. Methods & Technologies Used for Integration

RESTful APIs & Webhooks:

- The system uses RESTful APIs for synchronous communication with payment processors and banks.
- Webhooks are used to receive real-time status updates (e.g., payment success, transaction failure).

Asynchronous Processing via AWS SQS & SNS:

- Transaction updates and reconciliation processes are managed using AWS SQS for event-driven processing.
- AWS SNS sends notifications for status changes to different system components.

Standard Protocols:

- For certain integrations proprietary API standards.

OAuth 2.0 & JWT for Authentication:

- Secure OAuth 2.0 authentication is used to connect with payment gateways and banking APIs.
- JWT tokens provide a secure, stateless authentication mechanism for API access.

PKI & Mutual TLS (mTLS) for Secure Connections:

- Mutual TLS (mTLS) ensures encrypted and authenticated communication with financial institutions.
- Public Key Infrastructure (PKI) is used for certificate-based authentication.

3. Security Measures for External Connections

API Authentication & Authorization:

- All external APIs are authenticated using OAuth 2.0, API keys, and JWT tokens.
- Role-based and attribute-based access control (RBAC + ABAC) ensures only authorized entities can access sensitive functions.

Data Encryption:

- All API requests and responses use TLS 1.3 encryption.
- Sensitive data (e.g., user credentials, transaction details) is encrypted using AES-256.

Rate Limiting & DDoS Protection:

- API rate limiting is enforced to prevent abuse.
- AWS WAF & AWS Shield provide DDoS protection against API attacks.

Logging & Monitoring:

- API requests and responses are logged using AWS CloudTrail and CloudWatch for audit trails.

- Anomaly detection mechanisms alert admins for any unusual API activity.

Remittance Transaction Processing

4. Provide a step-by-step description of how a remittance transaction is processed from initiation to completion.

- Include all stages, parties involved, and settlement arrangements.

1. Initiation of the Transaction

- Parties Involved: Customer (Sender), Red Sea Money Transfer (Service Provider), Payment Gateway (VolumePay), Red Sea Money Transfer backend services

Process:

- The sender initiates a remittance transaction via the front-end interface (mobile app).
- The sender provides necessary details like recipient information (name, address, bank details), amount to be transferred, and the selected payment method (pay with bank app).
- The system verifies if the sender is onboarded and compliant (e.g., KYC verification).

2. KYC/AML Verification

- Parties Involved: Red Sea Money Transfer Backend, Identity Verification System (Sumsub)

Process:

- Before the transaction is processed, the system checks if the sender is KYC-verified.
- If not verified, the system triggers a KYC process, possibly using liveness detection and document verification (Sumsub).
- If verification fails, the transaction is stopped, and the sender is notified.
- If verified, the transaction proceeds.

3. Transaction Authorization

- Parties Involved: Sender's Bank or Payment Method (VolumePay), Red Sea Money Transfer Backend, Payment Gateway

Process:

- With VolumePay, Red Sea Money Transfer interacts with VolumePay's API for authorization.

- VolumePay performs checks on the payment method, including fraud detection.
- The transaction is authorized or denied based on the available funds and fraud checks.

4. Transaction Processing and Status Updates

- Parties Involved: Red Sea Money Transfer Backend, Payment Gateway (VolumePay), Recipient's Bank or Payment Method

Process:

- Once authorized, Red Sea Money Transfer processes the transaction.
- The transaction status is updated in real-time using state machines.
- The funds are debited from the sender's account and forwarded to the recipient's designated account, either through bank transfer or cash pickup (depending on the payout method).
- If any payment or currency conversion is involved, the backend performs the necessary currency exchange using defined exchange rate rules.

5. Cross-Border Transaction Handling

- Parties Involved: Red Sea Money Transfer Backend, Cross-border Financial Networks, Partner Banks/Payment Systems

Process:

- For international remittance, the transaction may go through cross-border financial networks or local payment systems.
- Red Sea Money Transfer ensures compliance with local regulations and partners with local banks or payment systems.
- The backend keeps track of the transaction through its state machine and updates the status accordingly (e.g., "Processing", "In Transit", etc.).

6. Settlement and Delivery of Funds to Recipient

- Parties Involved: Red Sea Money Transfer Backend, Recipient's Bank or Payment Partner

Process:

- After successfully crossing borders, the payment system settles with the recipient's payment provider (e.g., recipient's bank, digital wallet, or remittance agent).
- The recipient receives the funds either in their bank account, wallet, or as cash, depending on the payout method.
- At this stage, the status of the transaction is updated to "Completed."

7. Final Notification to Sender and Recipient

- Parties Involved: Red Sea Money Transfer Backend, Sender, Recipient

Process:

- Once the transaction is complete, both the sender and recipient receive notifications (email/SMS/app notification) confirming the successful transfer.
- The sender receives a confirmation with transaction details (amount, fees, recipient's name, etc.).
- The recipient receives a notification regarding the credit or availability of funds.

8. Transaction Review and Reporting

- Parties Involved: Red Sea Money Transfer Backend, Customer Service, Regulatory Authorities

Process:

- If necessary, a transaction review is performed, especially for flagged or suspicious transactions (compliance, AML, etc.).
- The transaction details are logged for regulatory and audit purposes.
- If the transaction meets compliance criteria, it is archived.

9. Settlement Arrangements

- Parties Involved: Red Sea Money Transfer, Payment Gateway (VolumePay), Partner Banks, Cross-Border Financial Networks

Process:

- Settlement with partner financial institutions and payment gateways may occur at the end of each business day.

- Currency conversion and fees are accounted for as per the exchange rate rules and partner agreements.
- Reconciliation processes are triggered for tracking the transaction and ensuring that funds are correctly credited to the recipient's account.

5. What channels are available for customers to initiate remittance transactions?

- Examples include web platforms, mobile apps, or in-person services.

Since our remittance service operates exclusively through mobile apps built with Flutter, customers can initiate transactions via:

Available Channels:

Mobile App (iOS & Android)

- Customers can use the Flutter-based mobile app to send money, track transactions, and manage their accounts.
- The app supports payment initiation via linked bank accounts, debit/credit cards, or digital wallets (depending on the payment provider).
- Users can also perform identity verification (KYC) and access customer support through the app.

Unavailable Channels:

- Web Platform: Not available (all transactions are mobile-first).
- In-Person Services: Not offered (no physical branches or agent locations).

6. How does the system manage currency conversion and exchange rates for international transfers?

- Explain the process and any external data sources utilized.

Our system handles currency conversion and exchange rates manually, managed by the treasury team, without direct integration with external data sources like XE or Reuters. However, the software has built-in capabilities to support external data sources if needed in the future.

Process:

1. Manual Exchange Rate Management

- The treasury team manually sets and updates exchange rates within the system based on internal risk assessments, market trends, and business strategy.
- These rates are stored in the exchange_rate_rules table, which determines applicable rates for different corridors (e.g., GBP to ETB, GBP to AED).
- The system supports different rates for different customer segments if required.

2. Applying Exchange Rates During Transactions

- When a customer initiates a remittance, the system fetches the applicable exchange rate based on:
    - Sender's currency
    - Recipient's currency
    - Transaction corridor
    - Customer-specific rate rules (if applicable)

- The exchange rate is applied in real-time before confirming the transaction amount to the sender.
- The system calculates the converted amount, transaction fee, and total debited amount based on predefined rate rules.

3. Settlement and Reconciliation

- The treasury team ensures that internal funds are sufficient to cover payouts at the defined rates.
- At the settlement stage, exchange rates are validated to ensure consistency between expected payouts and actual payouts.
- Any necessary adjustments (e.g., rate fluctuations, reconciliation mismatches) are handled manually by the treasury team.

4. External Data Source Capability (Not Currently Used)

- While we do not currently fetch real-time rates from providers like XE, our system is designed to integrate with external exchange rate sources if required.
- This allows flexibility for future expansion into larger networks where automated exchange rate management is beneficial.

Security Measures

7. What encryption methods are used to protect sensitive payment data at rest and in transit?

- Specify algorithms or standards (e.g., AES-256, TLS).

Although VolumePay manages payment processing, our system ensures that all sensitive customer and transaction-related data is encrypted both at rest and in transit for security and compliance.

1. Data at Rest Encryption

- Algorithm Used: AES-256

Implementation:

- We store transaction metadata, such as transaction status, payment method type (pay by bank app), and payment provider reference IDs.
- These records are AES-256 encrypted before being stored in our PostgreSQL database.
- Personally identifiable information (PII), such as customer details (name, phone number, email), is also hashed or encrypted where necessary.

2. Data in Transit Encryption

- Algorithm Used: TLS 1.2+ (Transport Layer Security)

Implementation:

- All communication between mobile apps and the backend is encrypted using TLS 1.2 or TLS 1.3.
- API requests to VolumePay for payment authorization and processing are securely transmitted using HTTPS (SSL/TLS).
- Any webhook data received from VolumePay is also validated and processed over a secure channel.

3. Secure Storage & Compliance Measures

- We Do Not Store Sensitive Payment Data
- Payment transactions are tokenized by VolumePay, and we store only the transaction reference ID for reconciliation.

Environment Variable Security:

- API keys and encryption keys are securely stored using AWS Secrets Manager or environment variables with restricted access controls.

8. Describe the access control mechanisms in place for the system and sensitive data.

- Who is responsible for authorisation and restrictions to the system?

Our remittance system employs Role-Based Access Control (RBAC) and Attribute-Based Access Control (ABAC) to enforce strict authorization rules for accessing the system and data.

1. Access Control Mechanisms

A. Role-Based Access Control (RBAC)

Users are assigned predefined roles with specific permissions.

Examples of roles:

- Customer: Can initiate transactions, view history, and manage their profile.
- Customer Support Agent: Can view customer transactions but cannot modify sensitive data.
- Treasury Team: Can update exchange rates and manage liquidity.
- Compliance Officer: Can review flagged transactions and customer KYC data.
- System Administrator: Has access to infrastructure and system settings.

B. Attribute-Based Access Control (ABAC)

Access is further restricted based on attributes such as:

- Country restrictions: A compliance officer may only access transactions for a specific region.
- Transaction status: Agents can view transactions but cannot alter completed transactions.
- Customer segment: VIP customers might have different support access levels.

C. API Security & Authentication

- OAuth 2.0 & JWT (JSON Web Tokens) for secure authentication.
- Multi-Factor Authentication (MFA) required for admin and treasury users.
- Session timeouts and IP whitelisting for backend access.

D. Database Access Controls

- Role-based database access ensures only authorized users can query or update sensitive data.
- Column-level encryption for sensitive fields like PII (name, phone number).
- Audit logging to track access and modifications.

2. Who Manages Authorization & Restrictions?

- System Administrator : Manages user accounts, roles, and permissions at the system level.
- Compliance Team : Grants access to transaction monitoring tools and PII on a need-to-know basis.
- Treasury Team : Controls access to exchange rate configurations and liquidity management.
- Engineering Team : Enforces security policies, access rules, and infrastructure-level restrictions.

9. How is user authentication handled for both customers and internal staff?

- Detail methods such as multi-factor authentication or biometric verification.
- Who has access to sensitive data? List roles/functions.
- Are there tools or logs for **monitoring access and potential breaches**?

1. Authentication Methods

A. Customer Authentication

Customers authenticate using multiple layers of security, including:

- Password-based login (with strong password policies).
- Multi-Factor Authentication (MFA) via:
    - Biometric Authentication (FaceID, Fingerprint via device authentication).
    - mPIN (4–6 digit personal PIN for quick login and transaction approvals).
    - SMS OTP / Email OTP (One-time password sent to registered contact).

B. Internal Staff Authentication

Internal users (support agents, compliance officers, admins) authenticate using:

- Username & Password (stored securely using bcrypt hashing).
- Mandatory MFA:

    - TOTP via authenticator apps (Google Authenticator, Authy).
    - IP whitelisting for admin and backend system access.
    - Session expiration & auto-logout to prevent unauthorized access.

2. Access to Sensitive Data

Access to sensitive data is restricted based on roles using RBAC (Role-Based Access Control) and ABAC (Attribute-Based Access Control).

- Customer: Can access only their own data and transaction history.
- Customer Support Agent: Can view customer transactions but cannot modify sensitive data.
- Compliance Officer: Can access customer KYC details, flagged transactions, and risk reports.
- Treasury Team: Can update exchange rates and monitor liquidity but cannot see customer PII.
- System Administrator: Can manage user roles, system settings, and logs but cannot access financial transactions directly.

No one user has unrestricted access to all data; access is need-based and logged.

3. Monitoring & Breach Detection

We use advanced monitoring tools to track access, detect anomalies, and prevent breaches.

Access Logs & Monitoring

- All authentication attempts, failed logins, and access events are logged.
- Logs are stored securely and reviewed periodically.
- AWS CloudTrail & SIEM tools are used for detecting suspicious activity.

Real-time Breach Alerts

- Repeated failed login attempts trigger account lockout & alerts.
- Anomalous behavior detection (e.g., logins from unusual locations or devices).

Audit Logs & Forensics

- Admin and compliance officers have access to audit logs to track who accessed what data.
- Immutable logs ensure that access records cannot be tampered with.

10. What measures are in place to detect and prevent fraudulent transactions?

- Include tools or processes for fraud monitoring and mitigation.

Our remittance system employs rule-based fraud detection, real-time monitoring, and automated risk assessment to prevent fraudulent transactions. We use a multi-layered approach, combining velocity checks, sanction screening, identity verification, and behavioral analysis to mitigate risks effectively.

1. Fraud Monitoring & Risk Mitigation Strategies

A. Rule-Based Risk Engine

Our fraud detection system evaluates transactions using predefined risk rules. Each rule assesses specific patterns indicative of fraud, such as:

- Customer Behavior Analysis (e.g., multiple accounts, unusual transaction volumes)
- Geolocation & IP Monitoring (e.g., IP-country mismatch, high-risk countries)
- Payment Instrument Analysis (e.g., multiple payment accounts)
- Sanction List & Compliance Checks (e.g., customer/recipient name screening)
- Velocity & Volume Monitoring (e.g., rapid transactions to same recipient)

Each rule assigns a risk score, and transactions exceeding a risk threshold are:

- Flagged for manual review (Compliance Team)
- Automatically declined or held for additional verification

2. Key Fraud Prevention Measures

A. Customer & Payment Instrument Verification

- KYC (Know Your Customer): Customers undergo ID verification before initiating transactions.
- Biometric & MFA Authentication: Prevents unauthorized account access.

B. Real-Time Transaction Monitoring

- Velocity & Volume Checks: Detect rapid or repeated transactions.
- Linked Transaction Analysis: Identifies repeated patterns (e.g., same sender/receiver, split payments).
- IP & Device Fingerprinting: Blocks high-risk devices and suspicious geolocation patterns.

C. Sanction & Watchlist Screening

- Global Sanction List Checks: Ensures no customer, recipient, or account is flagged.

- Email & Name Consistency Checks: Detects fake or anonymized identities.

D. Adaptive Risk-Based Decisioning

- Dynamic Risk Thresholds: Adjusted based on transaction patterns.

3. Tools for Fraud Detection & Monitoring

A. Rule-Based Fraud Detection Engine

- Evaluates transactions against predefined risk rules.
- Automated decision-making (Approve, Hold, Decline).

B. Logging & Real-Time Alerts

- SIEM Tools & Cloud Monitoring (AWS CloudTrail, CloudWatch).
- Audit Logs track admin and compliance actions.

C. Manual Review & Case Management

- High-risk transactions are flagged for compliance review.
- Fraud analysts investigate disputed transactions.

Communication Channels

11. What protocols are used for secure communication between system components and external parties?

- Examples include HTTPS, SFTP, or VPNs.

Our remittance system follows industry-standard security protocols to ensure confidentiality, integrity, and authenticity of data exchanged between system components and external parties.

1. Communication Between System Components

- HTTPS (TLS 1.2/1.3): All internal and external API communications are encrypted using HTTPS with TLS 1.2+.
- mTLS (Mutual TLS): For third-party integrations.
- AWS SQS & SNS (Message Queues): Secure message-based communication between internal services.

2. Communication with External Payment Providers (e.g., VolumePay)

- RESTful APIs over HTTPS (TLS 1.2+): Payment processing requests and responses are securely transmitted using encrypted channels.
- API Key Authentication: External API requests require secure authentication mechanisms.

3. Secure Data Transfer for Compliance & Partners

- SFTP (Secure File Transfer Protocol): Used for batch data transfers with compliance authorities and partners.
- PGP/GPG Encryption: Sensitive files are encrypted before transfer over SFTP.

4. Remote Access & Admin Controls

- VPN (Virtual Private Network): Required for accessing internal services securely.
- Bastion Hosts & IAM Roles: Secure access to cloud resources via role-based permissions.

5. Monitoring & Security Logging

- AWS CloudTrail & CloudWatch: Logs all API requests, access, and system interactions.
- SIEM Tools (Security Information & Event Management): Tracks anomalies in communication patterns.

12. How is the integrity and confidentiality of transaction data ensured during transmission?

- Describe mechanisms like digital signatures or encryption.

To protect transaction data during transmission, we use encryption and digital signatures, depending on implementation constraints. These mechanisms ensure that data remains confidential, tamper-proof, and authenticated end-to-end.

1. Encryption for Data Confidentiality

We use strong encryption protocols to prevent unauthorized access:

- TLS 1.2 / TLS 1.3 (End-to-End Encryption)
  - All API and system communications use HTTPS with TLS 1.2+ to encrypt data in transit.
  - Prevents man-in-the-middle (MITM) attacks and data eavesdropping.
- AES-256 Encryption for Secure Data Exchange
  - For transactions requiring file-based exchanges (e.g., SFTP transfers), sensitive data is encrypted using AES-256 before transmission.

- PGP/GPG Encryption for File Transfers
    - Used when compliance or external partners require additional encryption for batch transaction reports.

2. Digital Signatures for Data Integrity & Authentication

To ensure transaction data has not been altered or tampered with, we use:

- HMAC (Hash-Based Message Authentication Code)
    - Used in API calls to validate that messages originate from a trusted source.
    - Example: HMAC-SHA256 ensures that no third party modifies request/response data.

- JWT (JSON Web Tokens) with Signature Validation
    - Used for API authentication and transaction integrity.
    - Ensures that request payloads are signed using a private key and verified with a public key.

3. Integrity Verification & Monitoring

- Checksum & Hashing (SHA-256/SHA-512)
    - Data integrity is verified by hashing transaction details before sending and comparing hashes after reception.

- SIEM Monitoring & Logging
    - All transaction logs are monitored for anomalies or unauthorized data modifications.

13. Are there any specific compliance standards that the communication channels adhere to?

- Examples include PCI DSS or ISO 27001.

Our communication channels comply with PSD2 (Revised Payment Services Directive) and diligently follow ISO 27001 security best practices, though we do not hold an official ISO 27001 certification.

PSD2 Compliance

- Strong Customer Authentication (SCA): Transactions require multi-factor authentication (MFA) via TOTP, biometric verification, or OTP-based validation.

- Secure API Communication: We use OAuth 2.0 and mutual TLS (mTLS) to ensure secure communication with banks and third-party providers.
- Transaction Risk Analysis (TRA): Continuous monitoring and fraud detection mechanisms help mitigate risks.

ISO 27001 Best Practices (Not Certified but Followed)

- Data Encryption: All communications utilize TLS 1.2+ for data in transit and AES-256 encryption for stored sensitive data.
- Access Controls: We enforce strict Role-Based Access Control (RBAC) and Attribute-Based Access Control (ABAC) policies.
- Security Audits & Logging: Our system incorporates regular security reviews, logging, and monitoring to detect unauthorized access or breaches.

14. What **communication channels** are used between the system and:

- Customers (e.g., email, SMS, mobile app)
- Partner institutions
- Regulatory bodies

Our system facilitates secure and efficient communication with different stakeholders through multiple channels:

1. Customers

- Mobile App (Flutter): The primary interface for transaction initiation, status tracking, and support.
- Push Notifications: Real-time transaction updates and security alerts.
- Email: Account-related notifications, transaction confirmations, and regulatory disclosures.
- SMS: OTPs for authentication and critical alerts (e.g., fraud warnings).

2. Partner Institutions (Banks, Payment Gateways, Payout Providers)

- API Integrations (REST/GraphQL) over HTTPS (TLS 1.2+): Secure real-time transaction processing and settlement updates.
- SFTP (Secure File Transfer Protocol): Used for bulk transaction reconciliation when required.
- VPN (if required by partners): Secure network connectivity for direct system integration.
- Webhooks: Instant notifications about transaction status updates.

3. Regulatory Bodies (Compliance Reporting & Audits)

- Email (PGP-encrypted where necessary): Communication regarding regulatory queries and compliance audits.
- Secure Portals: Some regulators require periodic uploads through their online platforms.

Each communication method is selected based on security, regulatory requirements, and operational efficiency to maintain a seamless and compliant remittance process.

Monitoring and Incident Response

15. What tools and processes are used to monitor system performance and security?

- Specify software or methodologies employed.

We use a combination of software tools, methodologies, and best practices to monitor system performance and security, ensuring optimal operation and compliance.

1. System Performance Monitoring

AWS CloudWatch:

- Monitors Lambda functions, RDS databases, and other AWS resources.
- Tracks system logs, triggers alerts for thresholds (e.g., CPU usage, disk space), and provides historical metrics.

Sentry:

- Error tracking and performance monitoring tool used to capture application errors, crashes, and performance bottlenecks in real-time.
- Helps in identifying bugs, tracking trends in error rates, and providing insights into how to fix critical issues.

Uptimerobot:

- Provides uptime monitoring for public-facing services and ensures the application is available and responsive to customers.
- Alerts the team if there are issues with the availability of key services.

Datadog (APM):

- Application Performance Monitoring (APM) tool used to monitor the performance of the entire application stack, including tracing API calls, database queries, and service interactions.
- Helps identify performance bottlenecks, slow transactions, and improves overall system efficiency.

2. Security Monitoring

AWS GuardDuty:

- Real-time threat detection for AWS infrastructure. Identifies suspicious activity like unauthorized API calls, potential data exfiltration etc.

CloudTrail (AWS):

- Logs and monitors AWS API calls, providing an audit trail of all user activity within the AWS environment to detect unauthorized access or abnormal behavior.

Security Information and Event Management (SIEM) Solutions

- Collects security logs, events, and alerts to provide real-time analysis of security-related incidents, such as data breaches or attempted attacks.

OWASP ZAP (Zed Attack Proxy):

- Used for regular security testing and vulnerability scanning of APIs and web applications to identify potential security flaws and prevent attacks.

Web Application Firewall (WAF):

- A cloud-based WAF is used to protect against common web exploits like SQL injection, cross-site scripting (XSS), and DDoS attacks.

3. Compliance and Audit Monitoring

AWS Config:

- Continuously monitors AWS resources for compliance with security policies, ensuring that no unauthorized changes are made to system configurations.

4. Incident Management and Response

PagerDuty:

- Provides automated incident management, ensuring that any security or performance alerts trigger the appropriate workflows, notifications, and actions.

Jira (Security Incident Tracking):

- Tracks security incidents from detection to resolution. Issues are escalated, resolved, and documented in Jira to ensure compliance with internal security policies.

Business Continuity and Disaster Recovery

16. What is the business continuity plan for the IT system in case of major disruptions?

- Include strategies for maintaining service availability.

In the event of major disruptions, our Business Continuity Plan (BCP) ensures that our IT system remains operational and service availability is maintained. This plan includes several strategies and protocols to ensure resilience, minimize downtime, and quickly restore service functionality. Below is an outline of the key components of our plan:

1. Disaster Recovery Strategy (DRS)

Cloud-Based Infrastructure (AWS):

Our infrastructure is fully hosted on AWS to take advantage of its built-in redundancy, scalability, and fault tolerance. Key elements include:

Multi-Region Deployment:

Critical services are replicated across multiple AWS regions to ensure availability in case of regional outages. This ensures that if one region experiences a disruption, services can failover to another region.

Automated Failover:

Services, databases, and APIs are designed for automated failover in the event of infrastructure failure, ensuring minimal service disruption.

Backups:

Daily backups of data and configurations are taken across all critical systems, including databases, application data, and server configurations. These backups are stored across different availability zones (AZs) for redundancy.

Snapshotting:

Snapshots of databases are taken regularly for quick restoration in case of failure.

2. High Availability (HA) Setup

Redundant Network Architecture:

Our network architecture is designed to avoid single points of failure. This includes multi-AZ deployment for all critical components (e.g., web servers, databases, queues, etc.) to ensure that traffic can be rerouted if one part of the system fails.

AWS Lambda for High Availability:

We use AWS Lambda to ensure high availability by leveraging its ability to run code in response to events across multiple AWS regions. Lambda functions can be triggered automatically to handle failures or disruptions, ensuring system continuity with minimal human intervention.

3. Incident Response and Communication Plan

24/7 Monitoring:

Our system is continuously monitored using AWS CloudWatch, Datadog, and Uptimerobot for performance issues, system alerts, and service outages. In case of any disruptions, alerts trigger predefined responses and notifications to the Incident Response Team (IRT), which operates 24/7.

On-call Response Team:

A designated on-call team is always ready to respond to issues, troubleshoot, and take corrective actions in the event of a system disruption.

Communication Protocols:

In the event of a major disruption, clear communication protocols are followed to inform both internal stakeholders and customers. Regular updates are provided on the status of the issue and expected resolution timeframes.

- Internal Communication:

  Teams use tools like Slack and Jira for real-time updates and coordination.

- Customer Communication:

  Customers are notified via email, SMS, and in-app notifications to keep them informed of service outages and restoration timelines.

4. Data Integrity and Backup

Real-Time Data Replication:

Real-time data replication between primary and backup systems ensures that we are always working with the most up-to-date information. This helps minimize the impact of data loss during service disruptions.

Backup Testing:

We regularly test data restoration from backups to ensure the integrity and reliability of our backup systems. This ensures that recovery times are minimized during actual incidents.

5. Post-Incident Review and Continuous Improvement

Root Cause Analysis (RCA):

After a major disruption or incident, we conduct a Root Cause Analysis to identify the cause of the disruption, evaluate the effectiveness of our response, and implement improvements to prevent recurrence.

Testing and Drills:

Regular disaster recovery drills and business continuity simulations are conducted to ensure that all team members are prepared to respond quickly and effectively to any disruption, minimizing downtime and service impact.

17. How frequently are backups performed, and what are the Recovery Time Objective (RTO) and Recovery Point Objective (RPO) for the system?

- Provide specific timeframes and backup policies.

We have a structured backup and disaster recovery policy to minimize data loss and downtime. Our strategy includes real-time replication, scheduled backups, and versioned storage to meet strict Recovery Time Objective (RTO) and Recovery Point Objective (RPO) standards.

Backup Frequency and Storage Policies

- Database Backups: Incremental backups every 5 minutes, retained for 30 days.
- Full Database Backups: Daily full backups, stored for 90 days in AWS S3 Glacier.
- Application Logs: Real-time streaming to AWS CloudWatch, retained for 30 days.
- Infrastructure Snapshots: Every 6 hours, stored for 7 days using AWS EBS snapshots.
- Configuration & Secrets: Version-controlled backups stored securely in AWS Secrets Manager & AWS S3.
- File Storage Backups: Hourly incremental and daily full backups, retained for 90 days with AWS S3 versioning.

Recovery Objectives

- Recovery Time Objective (RTO): ≤15 minutes – Maximum allowable time to restore the system after failure.
- Recovery Point Objective (RPO): ≤5 minutes – Maximum allowable data loss in case of failure.

Disaster Recovery and Failover Strategy

- Automated Failover: Multi-AZ database failover, auto-scaling, and load balancing ensure minimal downtime.
- Cross-Region Replication: Critical backups (database, files) are replicated to a secondary AWS region.

Backup Testing & Validation:

- Quarterly disaster recovery tests validate quick restoration.
- Automated integrity checks prevent corrupted backups.

Compliance and Regulatory Requirements

18. How does the system facilitate compliance with Anti-Money Laundering (AML) and Counter-Terrorist Financing (CTF) regulations?

- Describe integrated controls or processes.

Our system is designed to facilitate compliance with AML and CTF regulations by implementing a rule-based system with real-time monitoring, identity verification, and transaction screening. The following controls and processes ensure compliance:

1. Customer Due Diligence (CDD) & Know Your Customer (KYC)

- Identity Verification: Integration with Sumsub for ID document verification, biometric checks, and liveness detection.
- Risk-Based KYC: Dynamic risk profiling based on customer attributes and transaction behavior.
- Sanctions & Watchlist Screening: Automated screening of customers and beneficiaries against The UK Sanction List, OFAC, EU, UN, and other global watchlists.

2. Transaction Monitoring & Risk-Based Rules

- Real-Time Transaction Monitoring: Every transaction is assessed against predefined AML risk factors.
- Suspicious Activity Detection:
    - Velocity checks (frequency and amount of transactions).
    - Linked transaction analysis (multiple senders/recipients, structuring).
    - High-risk jurisdiction checks.
    - Unusual pattern detection based on historical behavior.
- Automated Risk Scoring: Transactions are assigned risk scores, and high-risk transactions trigger enhanced due diligence (EDD).
- Rule-Based Alerts & Case Management: Suspicious transactions are flagged, logged, and escalated to compliance officers.

3. Reporting & Regulatory Compliance

- Automated SAR/STR Reporting: Suspicious Activity Reports (SARs) and Suspicious Transaction Reports (STRs) can be generated for regulators.
- Threshold-Based Reporting:
    - Large transaction reports (CTR) for cash-based transactions exceeding regulatory thresholds.
    - Cross-border transaction monitoring for high-risk corridors.

- Audit Trails & Logging: Every compliance-related action is logged for auditability and regulatory reporting.

4. Enhanced Due Diligence (EDD) for High-Risk Customers.

- Manual Review & Approval: High-risk transactions require manual intervention before processing
- Ongoing Monitoring: Continuous reassessment of customer risk profiles.

19. What customer due diligence (CDD) processes are integrated into the system?

- Include verification and monitoring procedures.

19. Customer Due Diligence (CDD) Processes

Our system incorporates a multi-layered Customer Due Diligence (CDD) framework to ensure compliance with regulatory requirements and mitigate financial risks.

1. Customer Identity Verification (KYC)

- ID Document Verification: Integration with Sumsub to verify government-issued identity documents (passport, national ID, driver's license).
- Liveness Detection & Biometric Matching: Customers must perform a real-time liveness check and match their selfie with their ID document.
- Document Authenticity Checks: AI-powered analysis of ID documents to detect forgeries and tampering.
- Address Verification: Utility bills, bank statements, or geolocation checks to confirm customer residency.

2. Sanctions, Watchlist, and PEP Screening

- Global Sanctions List Screening: Customers are screened against OFAC, EU, UN, and other regulatory watchlists.
- Politically Exposed Persons (PEP) Screening: Identifies high-risk individuals requiring enhanced due diligence (EDD).

3. Risk-Based Profiling & Monitoring

- Automated Risk Scoring:
    - Customer risk is assessed based on transaction patterns, geographic location, and historical activity.
    - Customers from high-risk countries or industries are flagged for additional scrutiny.

- Behavioral Analysis: Continuous monitoring of transaction behavior to detect unusual patterns.
- Velocity & Volume Checks: Limits on transaction frequency and amount within specific timeframes.

4. Ongoing Monitoring & Enhanced Due Diligence (EDD)

- Continuous Risk Assessment: Customers are re-evaluated periodically based on their transaction activity and regulatory updates.
- Transaction Monitoring & Alerts:
    - Real-time fraud detection mechanisms flag suspicious activities.
    - Linked transaction analysis detects structuring or smurfing attempts.
- Manual Review & Escalation: High-risk transactions require manual intervention by compliance officers.

5. Record Keeping & Audit Compliance

- Retention of Customer Data: All customer onboarding and verification records are securely stored for regulatory audits.
- Audit Trails & Logging: Every compliance-related action is logged to ensure traceability.

Data Management and Privacy

20. How is sensitive payment data collected, stored, and processed within the system?

- Detail data flows and storage methods.

- Describe how the system:

- Files and stores sensitive payment data
- Tracks data access
- Monitors data usage

1. Payment Data Collection & Processing

- Our system does not directly collect, store, or process sensitive payment data such as card details or bank account credentials.
- All payment transactions are processed through VolumePay, which handles payment data securely via Open Banking APIs.
- Customers are redirected to VolumePay's secure interface for payment authorization, ensuring compliance without our system storing payment credentials.

2. Payment Data Storage

Since we do not store payment data, the only related data retained in our system includes:

- Payment metadata (e.g., transaction amount, currency, timestamp, status).
- Masked payment identifiers (e.g., last few digits of a bank account, payment reference numbers).
- Banking partner references (e.g., transaction IDs provided by VolumePay).
- Compliance-related records (e.g., audit logs for AML monitoring).

3. Access Control & Data Protection

- Strict Role-Based Access Control (RBAC) & Attribute-Based Access Control (ABAC) limit access to transaction metadata based on user roles.
- Sensitive metadata is encrypted at rest and in transit using AES-256 and TLS 1.2/1.3.
- No direct database access for non-administrative users—access is granted through secure API endpoints.

4. Data Access & Monitoring

Access Logging & Auditing:

- Every access or modification attempt is logged.
- Audit logs are monitored using AWS CloudWatch & Datadog.

Fraud & Compliance Monitoring:

- Unusual transaction behavior is flagged based on risk rules.
- Sanctions screening & AML checks prevent unauthorized payments.

Anomaly Detection:

- Real-time alerts for suspicious API requests or unauthorized access attempts.

5. Compliance & Regulatory Considerations

- Since we do not handle or store payment data directly, PCI DSS compliance does not apply to our system.
- Our system follows PSD2 and ISO 27001 security best practices to ensure transaction integrity and security.
- Data retention policies ensure metadata is stored securely for regulatory compliance while avoiding unnecessary exposure of sensitive information.

21. What measures are in place to restrict access to sensitive payment data to authorized personnel only?

- Include access right policies and monitoring tools.

1. Access Rights & Role-Based Restrictions

- Role-Based Access Control (RBAC) & Attribute-Based Access Control (ABAC) ensure only authorized personnel can access payment metadata.
- Principle of Least Privilege (PoLP): Users are granted the minimum necessary permissions.
- Separation of Duties (SoD): No single individual has full control over payment data processing and approval.
- Multi-Factor Authentication (MFA): Required for all internal personnel accessing transaction records.

2. Data Encryption & Protection

- Sensitive metadata is encrypted:
  - At rest using AES-256.
  - In transit using TLS 1.2/1.3.
- No direct database access for non-administrative users—access is restricted through secure API endpoints.

3. Access Monitoring & Auditing

Access logs & audit trails:

- Every access request is logged.
- Logs are monitored using AWS CloudWatch, Sentry, and Datadog.

Real-time anomaly detection:

- Alerts for unauthorized access attempts.
- Suspicious activity flagged for review.

Periodic access reviews:

- Regular audits to ensure compliance with access policies.
- Immediate revocation of access for inactive or departing employees.

4. Compliance & Policy Enforcement

- Strict internal security policies based on ISO 27001 best practices.

- PSD2-compliant access controls for secure handling of financial data.
- Regular security assessments & penetration testing to identify vulnerabilities.

System Testing and Auditing

22. Are regular security audits or penetration tests conducted on the system? If so, how often?

- Provide frequency and scope.

1. Frequency

- Quarterly internal security audits to assess system vulnerabilities.
- Annual external penetration testing performed by third-party security firms.
- Continuous automated vulnerability scans using industry-standard tools.

2. Scope of Security Audits

Application & Infrastructure Security:

- Web and mobile applications.
- Cloud infrastructure (AWS services).
- API endpoints and authentication mechanisms.

Access Control & Data Protection:

- User access rights and privilege escalations.
- Encryption policies for data at rest and in transit.

Regulatory & Compliance Reviews:

- Ensure adherence to ISO 27001 best practices and PSD2 security requirements.
- Verify logging, monitoring, and anomaly detection mechanisms.

3. Penetration Testing

Conducted annually by an independent cybersecurity firm.

Covers:

- OWASP Top 10 vulnerabilities.
- API security (including authentication & authorization).
- Network and infrastructure security.

- Findings are documented, and remediation plans are implemented promptly.

23. What is the process for addressing vulnerabilities or issues identified during testing?

- Include timelines and responsible teams.

1. Identification & Reporting

- Vulnerabilities are identified through security audits, penetration tests, automated scans, and real-time monitoring (Sentry, CloudWatch, Datadog).
- Findings are categorized based on severity: Critical, High, Medium, Low.
- Issues are logged in the internal security ticketing system and assigned to relevant teams.

2. Prioritization & Response Timelines

- Critical (0-day, high-risk exploits): Immediate response; patch within 24-48 hours.
- High: Fix within 5 business days.
- Medium: Addressed within 10 business days.
- Low: Fixed in the next scheduled release cycle.

3. Responsible Teams

- Security Team: Evaluates risks, provides recommendations, and verifies fixes.
- Development Team: Implements patches and ensures secure coding practices.
- DevOps Team: Deploys security updates and monitors system performance.

4. Validation & Deployment

- Fixes undergo internal testing and code reviews before deployment.
- Security patches are rolled out using CI/CD pipelines for quick implementation.
- Post-deployment validation ensures the issue is resolved without regressions.

5. Continuous Monitoring & Documentation

- Retesting is conducted to confirm fixes.
- Incident reports and remediation steps are documented for compliance and audit purposes.
- Lessons learned are integrated into future security training and protocols.

Statistical Data Collection

24. What data is collected regarding system performance, transactions, and fraud?

- Examples include transaction volume, error rates, or fraud incidents.

1. System Performance Metrics

- Uptime & Availability: Monitored using Uptimerobot, Datadog, AWS CloudWatch.
- Latency & Response Times: API response times, database query performance.
- Error Rates: Logged errors, failed API requests, and system exceptions (via Sentry & CloudWatch).
- Infrastructure Usage: CPU, memory, disk usage, and network traffic.
- Service Health Checks: Real-time monitoring of critical components (Lambda, databases, SQS).

2. Transaction Data Collected

- Transaction Volume: Number of transactions processed per period (hourly, daily, monthly).
- Transaction Statuses: Pending, Completed, Failed, Reversed, or Disputed transactions.
- Processing Time: Time taken for transactions to complete (initiation to payout).
- Payment Method Details: Type of payment instrument used (Bank Transfer, Open Banking).
- Currency Exchange Metrics: Exchange rate applied, fees charged, and conversion amounts.

3. Fraud & Risk Monitoring Data

- Fraudulent Transactions Detected: Transactions flagged by AML rules or manual review.
- Velocity Checks: Frequency of transactions per user, recipient, or payment method.
- Unusual Activity Alerts: Large, frequent, or geographically inconsistent transactions.
- Blocked/Blacklisted Users & Accounts: Customers flagged for fraud or regulatory concerns.
- Chargeback & Refund Incidents: Number and reasons for chargebacks and refunds.
- Risk Rule Triggers: Logs of transactions hitting risk rules (e.g., sanctions list, IP risk, velocity limits).

System Scalability and Performance

25. How does the system handle scalability and performance during peak transaction periods?

- Include capacity planning and load testing details.

1. Scalability with AWS Lambda

- Serverless Architecture: AWS Lambda enables automatic scaling based on transaction load. The system dynamically scales to handle peaks in transaction volume without manual intervention.
- Concurrency Management: AWS Lambda handles high concurrency efficiently, scaling functions in real-time to match the demand. The system is capable of processing large transaction volumes in parallel, leveraging AWS Lambda's burst scaling feature.
- Function Timeout & Memory Allocation: Lambda functions are optimized for low latency and high throughput by adjusting timeouts and memory allocation per function based on peak usage patterns.

2. Capacity Planning

- Historical Data Analysis: Transaction volumes are analyzed based on historical data, allowing for capacity estimates to accommodate expected peaks.
- Provisioned Concurrency: For predictable high-demand periods, AWS Lambda provisioned concurrency is used to pre-warm a set number of instances, ensuring that the system can handle sudden surges in traffic without delays.
- Auto-scaling Events: Integrated with AWS CloudWatch, auto-scaling rules are set to trigger based on metrics like CPU usage, memory consumption, or custom Lambda metrics (e.g., processing time). This ensures the system scales horizontally when needed.

3. Load Testing

- Stress Testing: The system undergoes stress testing to simulate peak load conditions and ensure AWS Lambda functions perform optimally during heavy transaction periods.
- AWS Lambda Metrics & Scaling Testing: Load testing includes simulating a large volume of simultaneous requests, tracking Lambda function performance metrics, and ensuring error-free execution under high loads.
- Third-Party Tools: Apache JMeter or is used for load testing to simulate high traffic and gauge Lambda's response to real-world transaction surges.

4. Monitoring & Optimization During Peaks

- Real-Time Monitoring: Performance metrics are continuously monitored using AWS CloudWatch and Datadog to track Lambda execution times, error rates, and memory usage.
- Dynamic Adjustment: Lambda function configurations (timeouts, memory settings) are adjusted dynamically based on real-time traffic and transaction processing requirements.
- Auto-scaling of Other Services: In addition to Lambda, services like SQS, SNS, and RDS are auto-scaled to handle the increased load during peak periods.

5. Handling Latency & Throughput

- Asynchronous Processing: Long-running tasks are handled asynchronously using SQS and SNS, ensuring that short tasks (like simple transfers) are not delayed by more complex tasks.
- Optimization: Lambda functions are optimized for minimized cold start latency and efficient memory usage, ensuring transactions are processed swiftly even during high-volume periods.

Transaction Integrity and Reconciliation

26. How does the system ensure the integrity and accuracy of transaction data?

- Detail validation and error-checking mechanisms.

1. Data Validation Mechanisms

Input Validation:

- All incoming transaction data is validated for format, completeness, and consistency. This includes verifying the payment details, recipient information, and transaction amounts.
- Cross-Referencing with External Sources:
    - Customer data (e.g., email, IP address, name) is cross-checked against internal reference lists and sanction

Transaction Consistency:

- The system ensures that transaction details such as sender, recipient, amount, and currency match across all records. If there are discrepancies, the transaction is flagged for review.
- Duplicate Transactions: The system prevents duplicate transactions by checking the transaction identifiers and ensuring no repeat entries are processed.

2. Error-Checking Mechanisms

Error Logging and Monitoring:

- Errors and failed transactions are logged using Sentry for real-time tracking and troubleshooting.
- Common errors, such as invalid card details or payment method failures, are identified and logged for further investigation.

Transaction State Machine:

- A state machine is used for each transaction to manage different stages (e.g., initiated, pending, completed). If a transaction fails at any point, it transitions to an error state, ensuring errors are captured and handled in a structured manner.
- Retries for Temporary Failures: If a transaction fails due to temporary issues (e.g., connectivity issues, service downtime), the system attempts to reprocess the transaction a set number of times before flagging it for manual intervention.

3. Data Integrity in Distributed Systems

- Atomic Transactions: All transactions are processed as atomic units, ensuring that any failure during processing results in a rollback, maintaining consistency and avoiding partial updates.
- Concurrency Handling: The system ensures that multiple transactions for the same account or recipient do not lead to conflicts or data corruption by enforcing transaction locks or optimistic concurrency controls.

4. Reconciliation and Auditing

- End-to-End Reconciliation: A reconciliation process is in place to ensure that transactions processed through VolumePay (or other payment providers) match the records stored in the system. This ensures that transaction amounts, fees, and balances are correct and consistent.
- Audit Trails: An audit trail is maintained for every transaction, capturing all changes, approvals, and updates to transaction data. This allows the system to trace data back to its source and ensures full traceability for all operations.
- Transaction Reporting and Alerts: Reports are generated regularly to monitor transaction data for anomalies. Alerts are triggered if discrepancies or unusual patterns are detected.

5. Data Quality Assurance

- Automated Testing: Automated unit and integration tests are run to verify that the transaction logic works correctly, ensuring that all edge cases are handled and that the system processes transactions as expected.
- Quality Control Checks: The system performs quality checks at each step of the transaction lifecycle to ensure that the final output is accurate and in line with expected results.

27. What reconciliation processes are in place to verify transaction completeness and accuracy?

- Include frequency and methods used.

1. Real-Time and Batch Reconciliation

Real-Time Reconciliation:

- Transactions are validated instantly against expected records.
- Any mismatches trigger alerts for immediate investigation.
- Payment processing status is synced with VolumePay to confirm successful debits and credits.

Batch Reconciliation:

- Conducted at scheduled intervals (e.g., daily, weekly, or monthly) to verify all transactions.
- Batches of transactions are compared against payment processor reports, bank statements, and internal records.

2. Key Reconciliation Methods

Transaction Matching:

- The system matches each transaction ID with corresponding entries in VolumePay (or other payment gateways) and the internal ledger.
- Checks for duplicate, missing, or incorrect entries.

Bank Statement Reconciliation:

- Periodic comparison of bank statement records with internal transaction logs to ensure all funds have been settled correctly.

Fee Reconciliation:

- Ensures that fees charged by payment providers match expected deductions.
- Verifies that transaction fees and currency conversion rates align with pre-configured rules.

Account Balances Check:

- Ensures that system balance updates correctly reflect incoming and outgoing transactions.
- Detects discrepancies due to failed transactions or system errors.

3. Discrepancy Handling & Reporting

Automated Discrepancy Detection:

- Any differences in expected vs. actual transactions trigger alerts.
- Mismatched transactions are flagged for manual review and resolution.

Dispute Resolution Workflow:

- If a transaction is missing, duplicated, or incorrect, the issue is escalated for correction.
- The reconciliation team collaborates with banks, payment providers, and internal teams to resolve disputes.

Audit Logs & Reporting:

- Detailed audit trails capture every transaction modification and reconciliation attempt.
- Reports are generated to track reconciliation performance, error rates, and unresolved mismatches.

Financial Reporting Support

28. How does the IT system support the recording and reporting of financial information for regulatory and compliance purposes?

- Describe integration with accounting procedures to collect information on revenue, expenses, client money, transaction volume.

Our system ensures accurate financial recording and reporting by integrating with internal accounting processes to track revenue, expenses, client funds, and transaction volumes.

1. Recording Financial Transactions

Double-Entry Accounting System

- Every financial transaction is recorded using debits and credits to ensure accuracy.
- Transactions are categorized into ledgers such as revenue, expenses, and client money.

Automated Transaction Logging

- Each remittance transaction is recorded in the system, capturing details like:
- Amount, currency, sender, recipient, fees, exchange rate, and status.System-generated records ensure accurate tracking and prevent manual entry errors.

Fee & Revenue Tracking

The system automatically calculates and records:

- Service fees, payment processor charges, and currency conversion margins.
- Ensures transparency in revenue generation for auditing and compliance.

2. Integration with Accounting & Reporting

Accounting Data Exports

- Financial records are structured for easy integration with accounting systems.
- Periodic data exports in formats compatible with standard accounting software (e.g., CSV, JSON, API-based reporting).

Reconciliation with Bank & Payment Processors

- The system reconciles internal records with bank statements, payment gateways (VolumePay), and client balances.
- Any discrepancies trigger alerts and require reconciliation.

Audit Logs for Compliance

- A complete audit trail is maintained for every financial transaction.
- Logs include: transaction approvals, modifications, system checks, and discrepancies.

3. Regulatory Reporting & Compliance

Transaction Volume & AML Reports

- Regular reporting for Anti-Money Laundering (AML) & Counter-Terrorist Financing (CTF) compliance.
- Generates reports on suspicious transactions, high-risk geographies, and large-value transactions.

Tax & Financial Compliance Reporting

- System supports reporting for regulatory bodies based on tax and financial laws.
- Generates revenue and tax-related reports for audits.

4. Secure Data Storage & Access

- Role-Based Access Control (RBAC & ABAC)
- Only authorized finance and compliance teams can access financial reports.
- Access is logged and monitored for security and audit purposes.

IT Team and Responsibilities

29. Can you describe the structure of the IT team responsible for developing, maintaining, and securing the remittance system?

- Include roles and responsibilities.

The IT team managing the remittance system consists of five key roles, each responsible for development, maintenance, security, and compliance.

1. Chief Engineer / Lead Architect

- Designs the overall system architecture and ensures it meets scalability, security, and compliance requirements.
- Oversees infrastructure decisions (AWS Lambda, databases, API security).
- Leads high-level technical strategy and ensures system resilience.

2. Backend Engineer

- Develops and maintains core transaction processing, payment gateway integrations, and APIs.
- Implements fraud detection, AML checks, and ledger accounting.
- Ensures high availability and reliability of backend services.

3. Frontend & Mobile Engineer

- Builds and maintains the customer-facing web and mobile applications.
- Implements secure authentication methods (MFA, FaceID, TOTP).
- Optimizes user experience and ensures seamless transaction flows.

4. DevOps & Security Engineer

- Manages AWS infrastructure, deployments, CI/CD pipelines, and high availability (HA) configurations.
- Ensures secure communication protocols (TLS, HTTPS, VPNs) and cloud security best practices.
- Monitors system performance using AWS CloudWatch, Sentry, Datadog, and Uptimerobot.
- Conducts security audits, vulnerability assessments, and incident response.

5. Compliance & Risk Analyst

- Ensures AML/CTF compliance and monitors fraud risk rules.
- Oversees transaction monitoring, sanction list checks, and KYC verification processes.
- Works with regulatory bodies and generates compliance reports for audits.

Outsourcing (if applicable)

30. Are any parts of the IT system or remittance processing outsourced to third parties? If so, how are these outsourced functions monitored and controlled?

- Detail oversight mechanisms and agreements.

Yes, certain parts of the IT system and remittance processing are outsourced to third-party providers. These are monitored and controlled through strict oversight mechanisms, contractual agreements, and compliance checks.

Outsourced Functions:

- Payment Processing → Handled by Volume Pay (Open Banking).
- Identity Verification & KYC → Managed by Sumsub.
- Monitoring & Logging → Managed via Sentry, AWS CloudWatch, Uptimerobot, and Datadog.
- Infrastructure Hosting & Security → Hosted on AWS (Lambda, S3, RDS, etc.).

Oversight & Control Mechanisms:

Contractual Agreements & SLAs

- Each third-party provider is bound by Service Level Agreements (SLAs) ensuring uptime, response times, and compliance with PSD2 and ISO 27001 practices.
- Regular vendor reviews ensure that providers maintain agreed service levels.

Compliance & Security Audits

- Identity verification (Sumsub) and payment processing (Volume Pay) are monitored for regulatory compliance (AML, CTF, KYC).
- Regular security audits and penetration testing ensure data protection.

Real-time Monitoring & Logging

- AWS CloudWatch, Sentry, and Datadog track system performance, error rates, and anomalies.
- Fraud detection mechanisms actively monitor transactions for suspicious patterns.

Incident Response & Escalation

- If a third-party service fails or introduces a security risk, a predefined escalation process is triggered.
- Failover mechanisms (e.g., backup payment providers) ensure continuity.

Regulatory & Data Privacy Compliance

- Third-party providers handling sensitive data must comply with GDPR, PSD2, and ISO 27001 best practices.
- No payment data is stored within the system, reducing PCI DSS requirements.