# Introduction:

Requirement Analysis

The Automated Payment Fraud Tracking and Dispute Resolution System for Brillian Bank aims to revolutionize the bank’s fraud detection and dispute management processes. This system will automate the reading of digital application forms and handle complaints received via IVR and Dispute raising. For each potentially fraudulent transaction, the system will verify the payment with the relevant card network's fraud detection system, ensuring the validity of payments and merchant refunds. Additionally, it will generate detailed fraud reports and facilitate manual approvals for refund transactions.

The system will support dispute claims and allow customers to submit and track disputes. The system ensures efficient handling of a high volume of customer disputes. With robust role-based access control (RBAC) implemented, the system ensures secure viewing of reports, and customers will be able to view the status of their disputes in real-time.

Designed to serve Brillian Bank customer base across Europe and North America, the system will also manage disputes for international payments, providing a robust and scalable solution to enhance the bank’s fraud detection and dispute resolution capabilities.

**User Requirement Analysis**

### Functional Requirements:

**FR1:** Customers can securely log in to the system using their credentials.

**FR2:** Customers can reset their passwords via email, ensuring a secure password change process.

**FR3:** Customer credentials are securely stored to protect personal data, adhering to best practices in data security. **FR4:** Customers can file complaints via an Interactive Voice Response (IVR) system, with the IVR behavior mocked using Postman.

**FR5:** Customers can raise disputes using online forms.

**FR6:** Customers receive summaries of their dispute resolutions and ticket updates via email.

**FR7:** Customers can track their disputes with real-time status updates.

**FR8:** Customers can generate and download reports, and simultaneously get the reports via email.

**FR9:** Customers can perform international payments supporting multi-currency transactions.

**FR10:** Customers can translate the page into their preferred language.

**Admin Requirement Analysis**

### Functional Requirements:

**FR1:** Admins can securely log in to the portal and access admin-specific data.

**FR2:** Admins can reset their passwords via email.

**FR3:** Admins ensure that all user credentials are encrypted to maintain security.

**FR4:** Admins can access and manage complaints and disputes, particularly those involving refund amounts that meet or exceed a specified threshold value.

**FR5:** Admins have the privilege to manually approve or reject requests.

**FR6:** Admins will receive notifications regarding new complaints and updates from vendor.

**FR7:** Admins manage integrations with card networks and oversee fraud disputes related to these networks.

**FR8:** Admins can generate and download reports, and simultaneously get the reports via email. Using these reports, admins can analyze fraudulent incidents.

**FR9:** Admins manage the integration module for external data access by card networks.

## Vendor Requirement Analysis

### Functional Requirements:

**FR1:** Vendors can securely log in and access vendor-specific data.

**FR2:** Vendors can reset their passwords via email.

**FR3:** Vendors credentials are encrypted to prevent unauthorized access.

**FR4:** Vendors will receive notifications regarding new and existing disputes from bank.

**FR5:** Vendors can view and respond to complaints and disputes related to their services.

**FR6:** Vendors can generate an API access key to retrieve information related to their card network.

**FR7:** Vendors can generate, download reports, and simultaneously get the reports via email.

# Non-Functional Requirements:

1. Strong encryption and secure password reset.
2. Restricted access and robust error handling.
3. User-friendly design and quick loading forms.
4. Resilience to failures and adherence to security standards.
5. Efficient handling of high volumes and scalability.
6. Data protection, compliance, and integrity.
7. International dispute processing and security measures.
8. Integration reliability and scalability.
9. User access control and maintainability.

# In-Scope:

1. **User Authentication and Authorization**: Implement secure authentication and authorization mechanisms for users, admin, vendor, including password reset via email.
2. **Encrypted Credentials Storage**: Store user credentials using strong encryption methods to prevent unauthorized access, adhering to industry best practices.
3. **Complaint Management**: Enable customers to file complaints via IVR, dispute forms, with mock data transmission tested via Postman.
4. **Integration with Card Networks and Fraud Systems**: Integrate with card networks and fraud detection systems for transaction verification and resolution and raising refunds.
5. **Reporting and Fraud Analysis**: Provide role-based reports and in-depth analysis of fraudulent incidents, trends, and patterns.
6. **Manual Payment Approvals**: Enable manual approvals for high-value refund payments with flexible routing, notifications to designated approvers, and actions by approvers.
7. **Dispute Management**: Allow customers to submit and track disputes with real-time status updates, email notifications, and a detailed dashboard displaying all dispute information.
8. **Responsive Web Interface and Data Security**: Ensure the web interface is fully responsive for laptops, tablets, and mobile devices; store data securely with encryption and comply with international data protection regulations.
9. **Support for International Transactions**: Facilitate domestic and international payments with real-time forex values, support for multiple currencies, automatic currency conversion, and fraud detection for international transactions.
10. **Seamless Data Exchange**: Ensure seamless data exchange and integrity with external systems via APIs, supporting compatibility with diverse protocols and data formats.

# Out-Scope:

1. Capturing handwritten application forms: The project will not involve the development of functionality to capture handwritten application forms or convert them into digital format.
2. Handling a high volume of complaints (5000 per hour): The project will not include optimizing the system to handle such a high volume of complaints within the specified time frame.

# Architecture Constraints:

1. Design Constraint: The architecture must follow a microservices pattern to allow independent deployment and scaling of components.
2. Security Constraint: Password will be stored after encryption and the data will be stored securely.
3. Process Constraint: The development process must follow the Agile methodology with regular sprints and continuous integration. There must be automated testing in place to ensure at least 80% coverage in test cases.
4. Error Handling: Any transient faults in the system will be handled using Resilience4j.
5. Coding Standards and Best Practices Constraint: Proper variable names must be used to enhance code readability and maintainability. Static data must be stored in external configuration files (e.g., JSON) and not hardcoded within the application code. Consistent code formatting and styling should be enforced across the codebase. Comprehensive code documentation and comments should be provided for complex logic and algorithms.
6. Notification Constraint: Users will receive notifications about actions taken on their disputes.
7. Email Constraint: Users will receive emails for account accessing, transaction, reset password and report download.