Accutraxx – Architecture Diagram

Version 1.0

Table of contents

1. Introducción

Accutraxx is an issuing system where a user can perform transactions. Since WA and Accutraxx are directly related, a user must have an active WA account. WA is a product implemented over Accutraxx. It processes the transaction of users. Accutraxx is a business application used mainly for online banking transactions.
This architecture document aims to illustrate the design and architectural structure of Acccutraxx for a clear understanding of developers, Project managers, Technical Users, and stakeholders.

2. Description of Solution

The software Architecture design of Accutraxx aids in understanding the architectural flow of the product. It shows how different products are connected with Accutraxx and the type of payment Accutraxx processes.

The internal system of Accutraxx automatically resolves the problems enocuntered. These are reoslved through making use of automated systems built in.

Following are a few examples of sections which are automated.

- An email notification system is set. An email is automatically sent to the network administrator if the server is down or any other issue is identified.
- Remote Access is available. The router, switch, or servers can be automatically adjusted remotely.
- The networking issues are resolved through Datto-RMM and Level-5 applications which are automated.

3. Specification of architectural requirements

Functional Requirements

Following are the few functional requirements Accutraxx fulfils:

- It processes the transaction for the IFARHU-CA scholarship programs.
- To safely secure the database on Level 5.
- Two databases: SQL and MongoDB.
- The user can only sign up with a CedulaD.
- A user's transactions are only processed if they have an active bank account.

Non-Functional Requirements

The following are the non-functional requirements:

- Improve the usability of the system.
- Enhance the reliability and security of the system.

Architectural Constraints
• Accutraxx is an issuing system that processes transactions done by the user. • The current transactions are done view Accutraxx is of IFARHU-CA. • A user must have a valid bank account before processing a transaction via Accutraxx. • The WA (Mobile app) and Solidarity are integrated with Accutraxx. • To process a transaction via Accutraxx, a user must have an active WA account. • Accutraxx only processes transactions done through a bank account, virtual or physical card.

4. Design of Reference Architecture

The general reference architecture of Accutraxx is provided. The modules previously developed and connected are taken into account. The architecture map shows how in general a user's trace is created on Accutraxx and further how a new user account is created on Accutraxx from backend prespective.

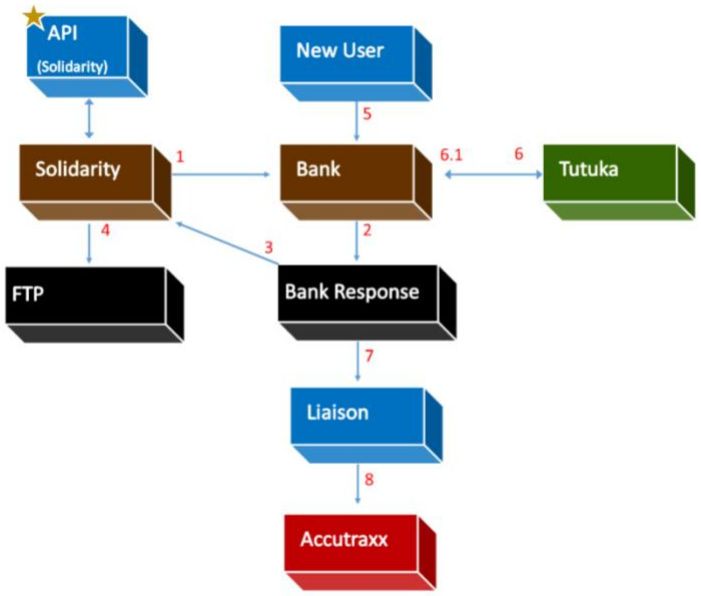

Illustration 1. Reference diagram

The above image initiates where a reload file is sent to the bank from Solidarity. The bank shares back a file with Solidarity. The bank's response is also imported on Liaison, and the trace is saved on Accutraxx.

5. Logical View

The following diagrams shows the logical flow of Accutraxx. It elaborates the logical view of all different components and how they interact. The main purpose of logical view is to display a brief graphical representation of how different components are connected and interact at different levels.

1) Component diagram of Accutraxx Architecture

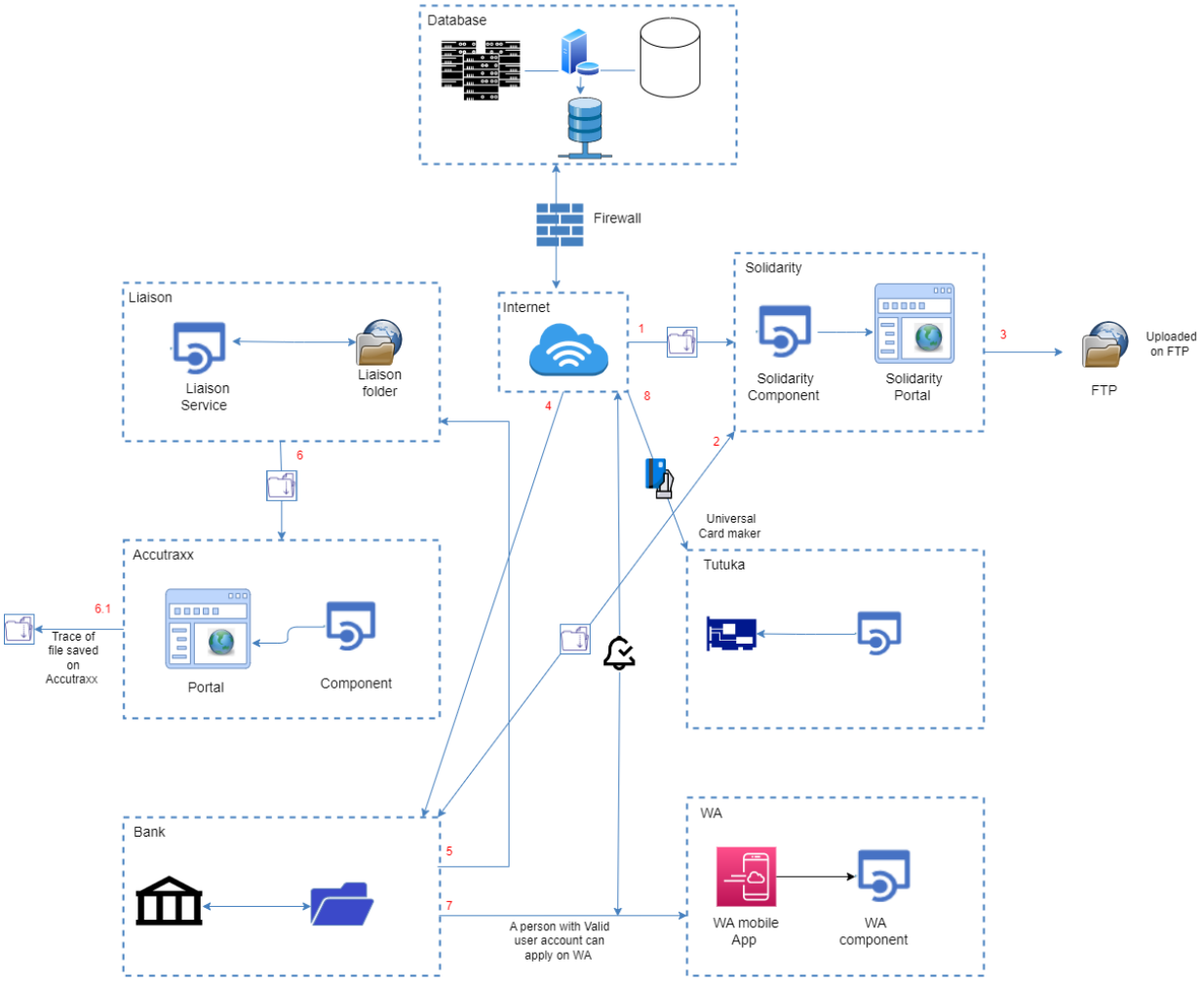

Illustration 2. Component diagram

1. File Solidarity (reload file)
2. Generate a file and send it to the bank (Solidarity). Imports the bank response. Generate a new file of applied customers and send it to the bank. Bank send us a new file (final file)
3. Uploads ON FTP.
4. Now a person can apply for a bank account.
5. Liaison service imports that file

6. The final file is sent to Accutraxx
7. A trace of final file is also saved on Accutraxx.
8. Users with a bank account can sign in to WA
9. A user with a valid bank account applies for the card on Tutuka.

2) Data flow of Accutraxx Architecture

The data flow will focus on showing the static view of Accutraxx. Following are the five systems that are related to Accutraxx (portal). All these five systems interact before a client can make virtual or physical payments using Accutraxx issuing system.

- Solidarity (Reloads)
- Liaison Service (File Processing Service)
- WA (Mobile)
- Bank (CA API)
- Tutuka (Paymentology)

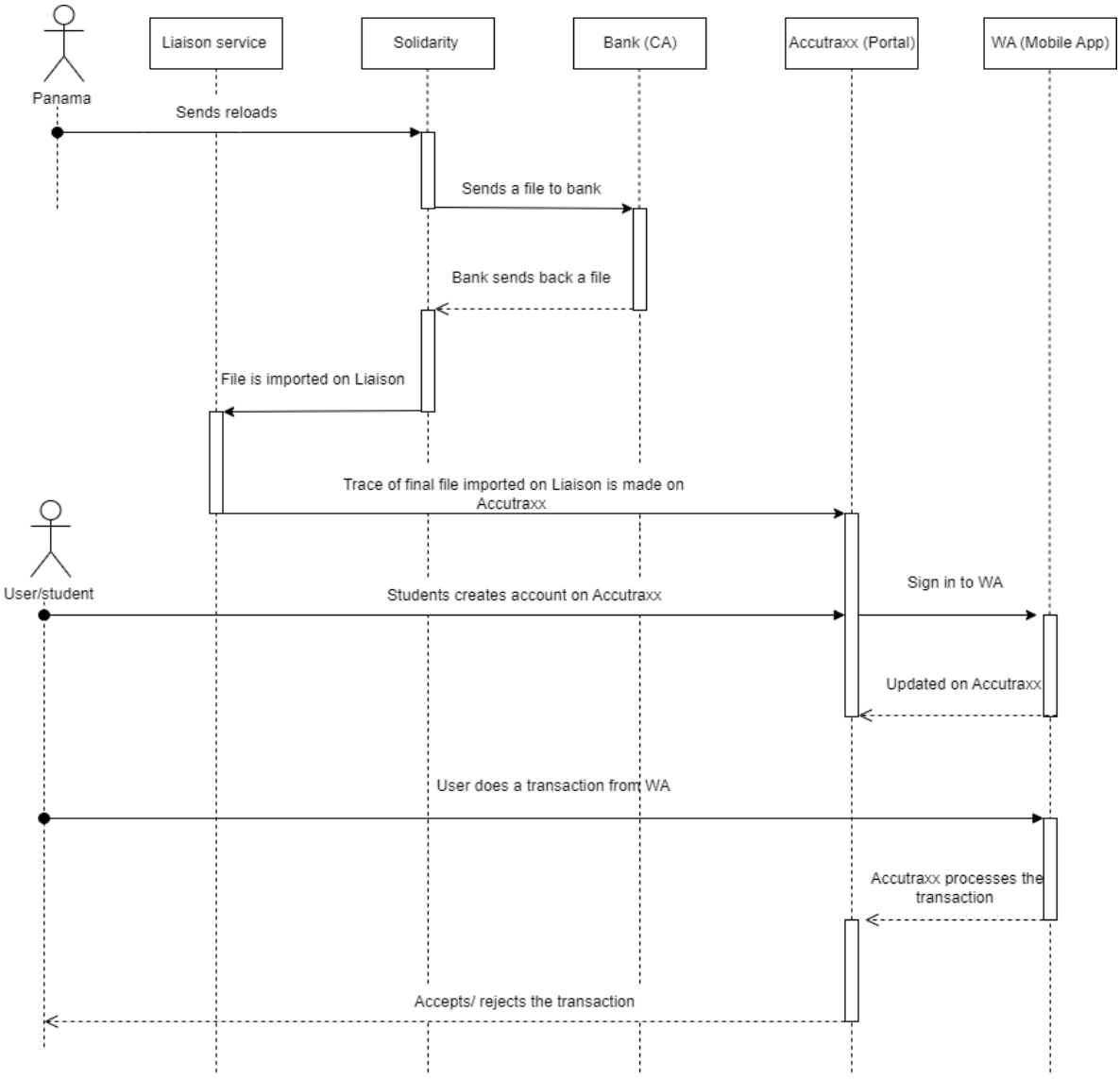

Illustration 3. Data flow diagram

6. Process View

A continuación, se presenta un diagrama donde se puede observar el proceso general que se busca completar con la integración de Mantenimiento Américas.

1) Client creating account on Accutraxx

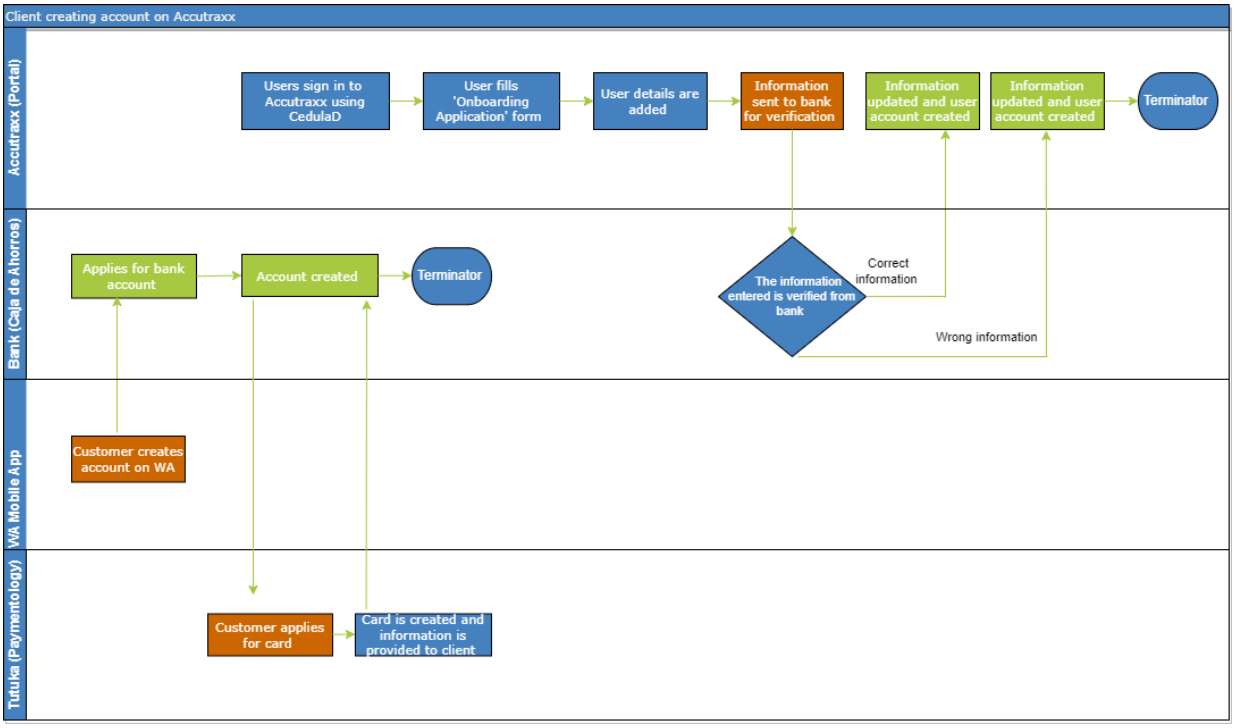

Illustration 4. Client creating account

2) Performing a transaction

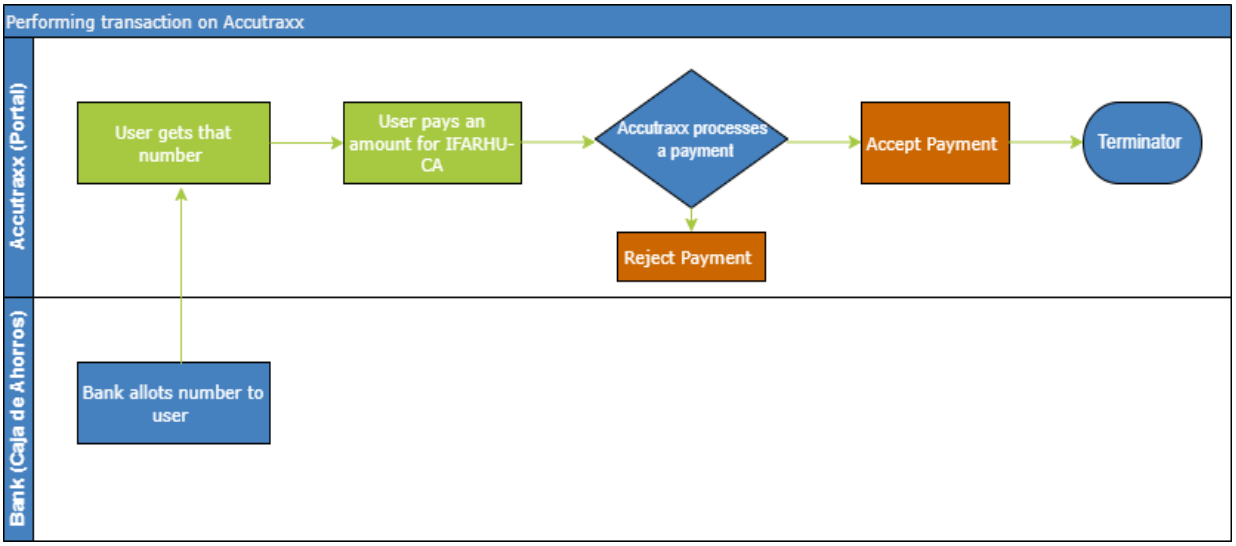

Illustration 5. Performing a transaction

3) Network Security of Accutraxx

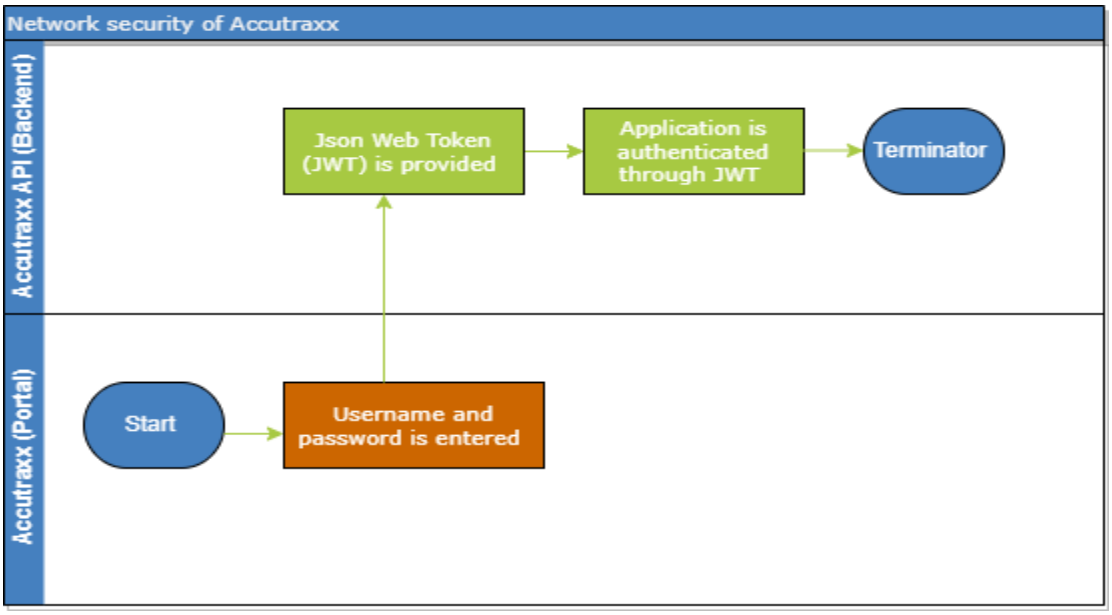

Illustration 6. Network security of Accutraxx

7. Architecture of Resilience

Two servers manage Accutraxx. These two servers are managed by Level 5.

Following are the two servers:

- PCI: It is responsible for managing production. This server is secure. All the development done is stored on this server.
- Non-PCI: It manages all the tests and other minor activities. This server is as secure as all the testing is done through this server.

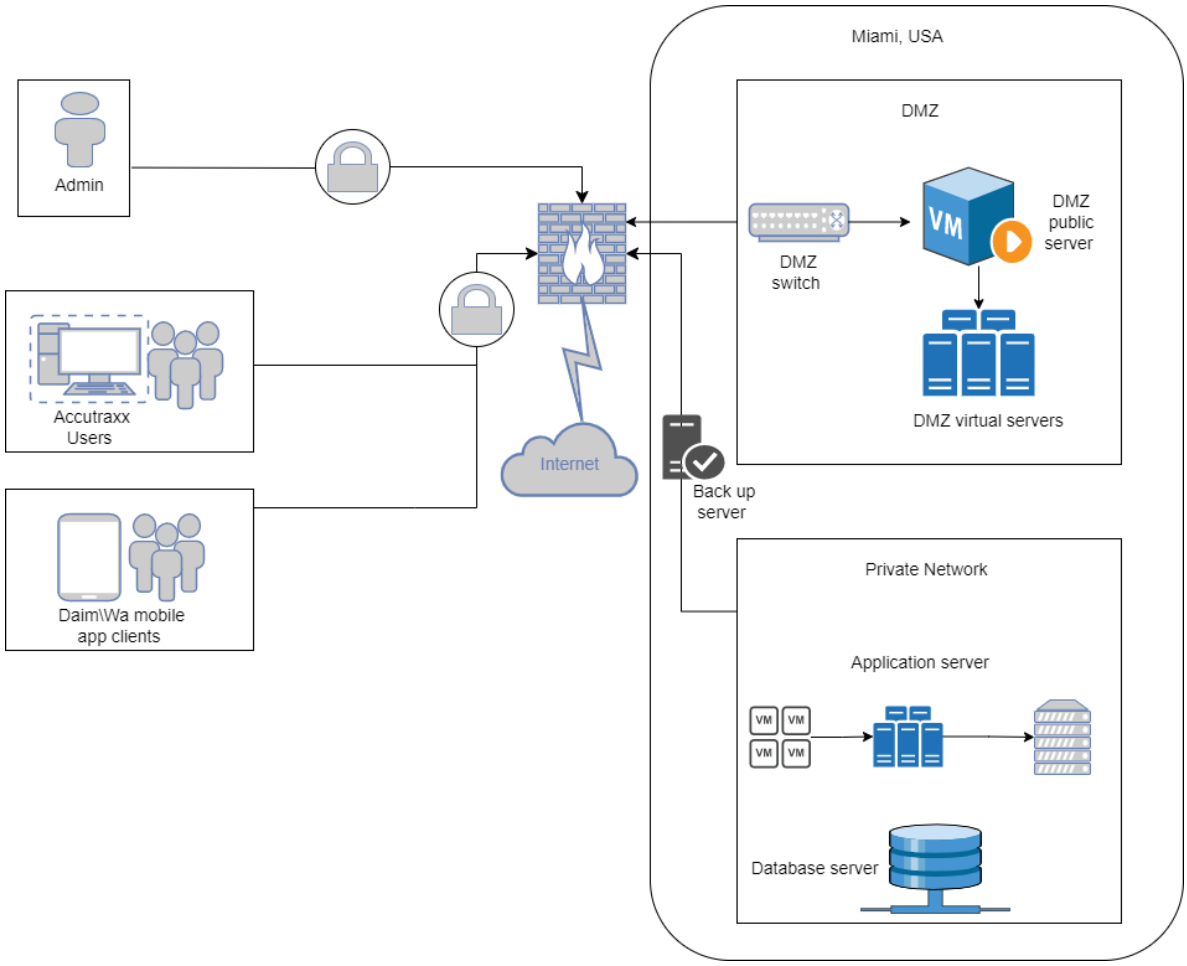

Illustration 7. Architecture of Resilience

8. Layer Diagram

The Layer diagram of Accutraxx is shown in 3 tier architecture designs.

It is based upon three layers:

- Presentation Layer: Client/customer
- Application Layer: Level-5, which securely contains all the data.
    - PCI: It is secured and holds all the production data.
    - Non-PCI: It is less secure and has all the test cases.
- Database Layer: SQL and Mongo server.

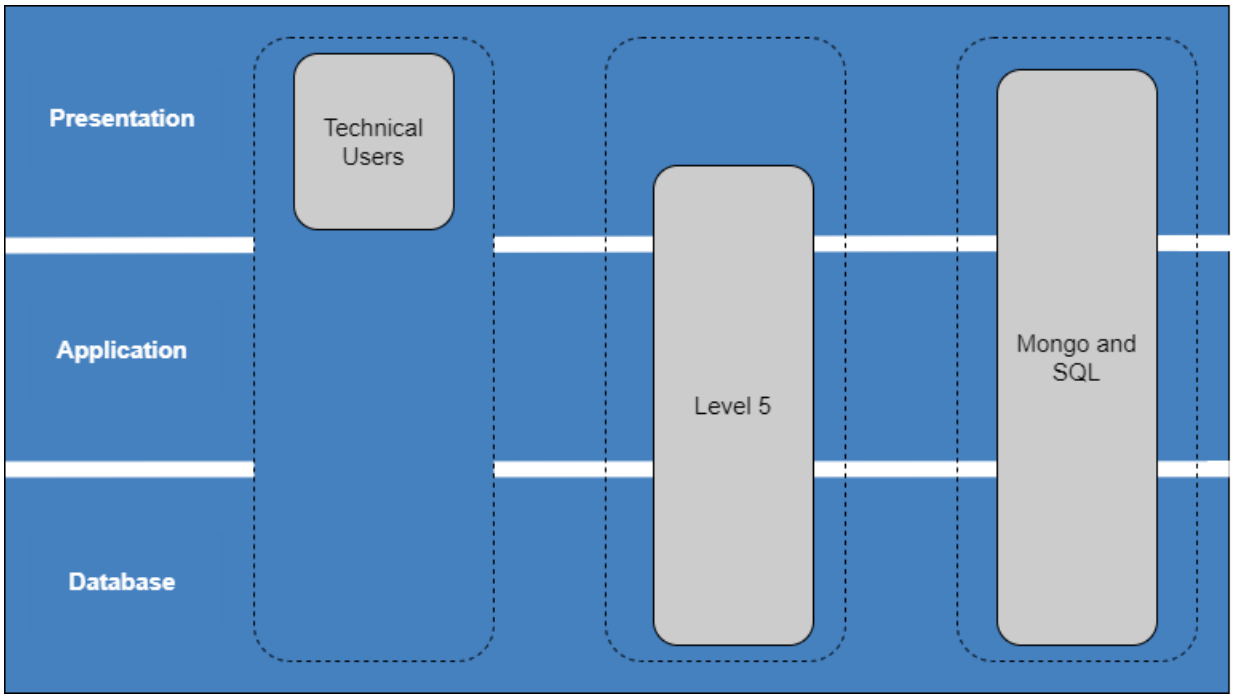

Illustration 8. Layer diagram

Explanation
The high-level architecture design represents the 3-tier architecture design of Accutraxx. The presentation layer includes all the Technical users who might code, Test, deploy or review the product.
The application layer holds the Level-5 company, where all the data is secured. PCI is for secure data where production is done. Non-PCI is for testing phases and is not secure.
Database Layer contains the databases on which Accutraxx performs. Currently SQL and Mongo is used.

9. Connectivity

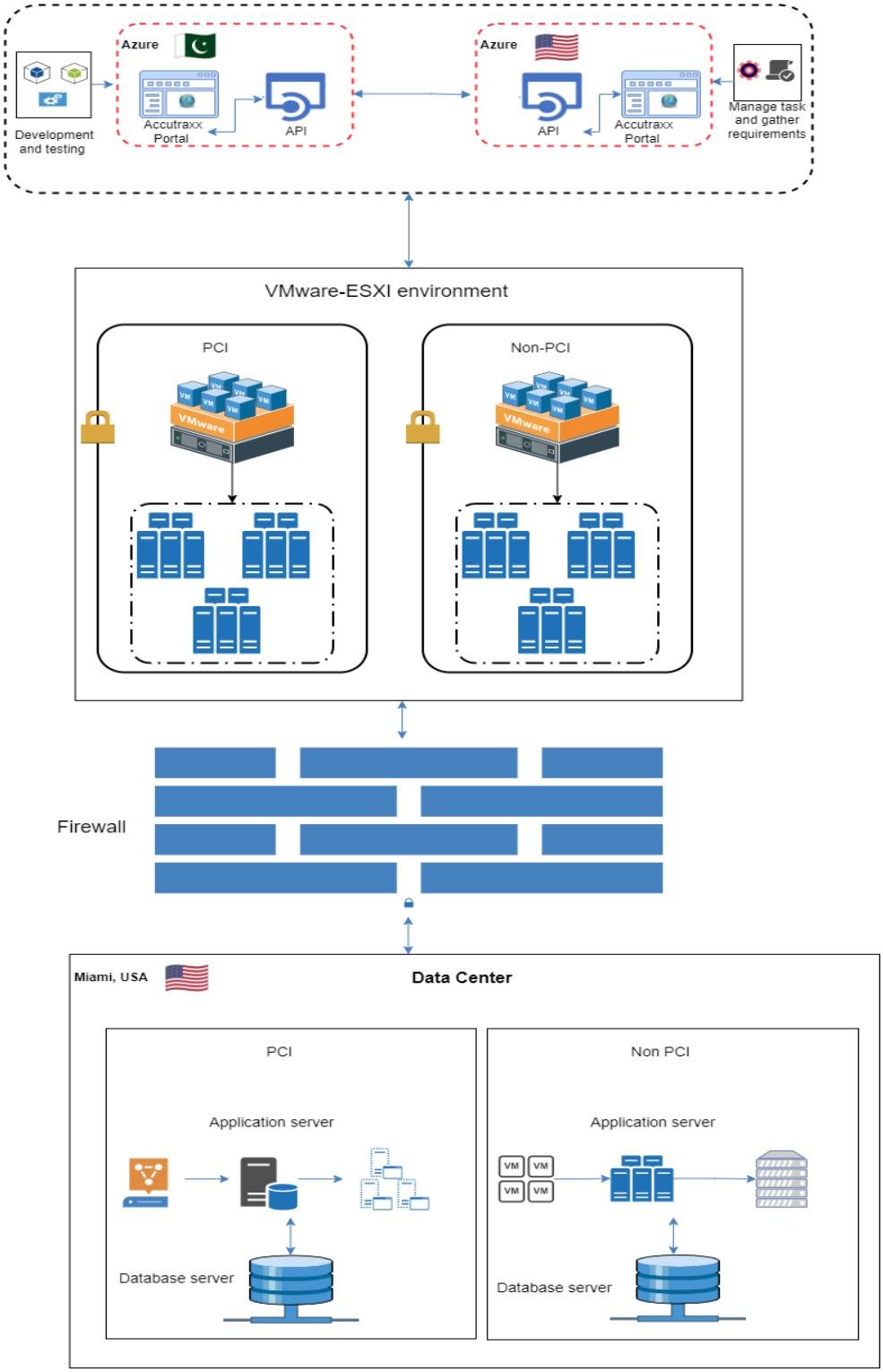

Illustration 9. Connectivity diagram

10. Roles

Role	Description	User
Software Architect	Maintenance	Plan and maintain information for the development of the current platform.
System Administration	Information	Know and understand the different components, connections, and capabilities for troubleshooting and business continuity.
Product Development	Information	Understand base components, connections, and capabilities to plan for new features and functionalities.
Tech Support	Information	Understand base components for troubleshooting purposes.

Tabla 1. Role users.