**ROOT CAUSE ANALYSIS (RCA)**

**TRANSPORT NETWORK VEHICLE SYSTEM  
(ADMINISTRATIVE)**

**TRAILAD.CO**

**MV Campus**

**Quezon City, Philippines, 1118**

**10/2/2025**

**Table of Contents**

[Introduction 1](#_Toc289504777)

[Event Description 2](#_Toc289504778)

[Chronology of Events / Timeline 3](#_Toc289504779)

[Investigative Team and Method 4](#_Toc289504780)

[Findings and Root Cause 5](#_Toc289504781)

[Corrective Action 6](#_Toc289504782)

**Introduction**

The purpose of this Root Cause Analysis (RCA) is to identify and understand the main causes of a system issue that occurred during the development of the **TNVS Administrative System**. This analysis aims to find out what happened, how it happened, and why it happened, so that the development team can prevent similar problems in the future.

Root causes should be specific, identifiable, and manageable within the team’s control. This RCA helps improve the system’s reliability, ensures smooth functionality, and supports quality performance before deployment. The findings and corrective actions from this RCA will also help strengthen the project’s testing process as the team prepares for the final presentation

**Event Description**

On **October 2, 2025**, during the integration testing phase, the **Admin Dashboard** of the TNVS Administrative System experienced a major issue where it failed to display any data from the database. This issue appeared when the team was testing the connection between the **Admin Module** and the **Reservation Module**. The problem was first discovered by **Team Member Karylle M. Muyargas**, who noticed that all dashboard statistics and records were blank despite successful login.

The issue affected the entire TNVS Administrative testing process, as several key features—such as reservation tracking, driver management, and report viewing—could not be tested. This delayed other testing activities and required the project leader to immediately initiate an investigation. The problem had the potential to affect the project schedule, so identifying the root cause quickly was crucial to avoid further delays.

**Chronology of Events/Timeline**

* **October 2, 2025 (9:30 PM):** The development team began integration testing between the Admin and Passenger modules of the TNVS Administrative System.
* **October 2, 2025 (9:45 PM):** Team member **Karylle M.** observed that the Admin Dashboard was not displaying any data, even though the system logged in successfully.
* **October 2, 2025 (10:00 PM):** The team checked the database connection and confirmed that the connection file (db\_connect.php) was configured correctly.
* **October 2, 2025 (10:15 PM):** The project leader, **Chris John Rey T. Casimiro**, reviewed the code and found that one SQL query was returning empty results due to a missing table reference.
* **October 2, 2025 (10:45 PM):** The team leader corrected the SQL query by including the correct table name (“reservations”) and re-ran the test. The dashboard successfully displayed data.
* **October 2, 2025 (1:10 AM):** The issue was officially logged, and documentation was started for RCA purposes.
* **October 3, 2025 (2:00 AM):** Additional tests were conducted to verify that the fix worked and that other features were not affected.
* **October 4, 2025 (4:30 AM):** The team leader performed a full system test and confirmed that all dashboard features were stable and functioning properly.

**Investigative Team and Method**

The investigative team was formed by the project manager to analyze and identify the main cause of the system failure. The following members were involved in the investigation:

* **Chris John Rey T. Casimiro** – Project Manager / Team Leader
* **Karylle M. Muyargas** – Tester / Quality Assurance

The team gathered information by reviewing the system’s error logs, checking database connections, and inspecting the latest code changes. They recreated the issue in a controlled environment to observe the behavior of the Admin Dashboard. After analyzing the code, the team found that a missing SQL table reference prevented data from being displayed.

To confirm this, the team compared the current dashboard code with previous working versions and verified that the issue originated from a recent update to the SQL query. Interviews with the team members also confirmed that this specific code update had not been fully tested before integration.

**Findings and Root Cause**

After a detailed review, the team identified the following findings:

1. The SQL query in the Admin Dashboard lacked the correct table reference to the “reservations” table, causing data not to load.
2. The missing reference happened because of a small oversight during code editing before integration testing.
3. The issue was not detected earlier since the initial tests focused on module functionality, not the overall system connection.
4. The database and system connection were both functioning correctly, meaning the problem came from the code, not from the hardware or server.
5. There was no checklist to ensure cross-module database verification before integration testing.

Based on these findings, the **root cause** of the issue was determined to be a **missing database table reference in the SQL query**, caused by incomplete verification during the system integration phase.

**Corrective Action**

To prevent similar issues in the future, the project team has taken several corrective actions. First, a **code review checklist** was created to ensure that all SQL queries are verified before integration testing. Second, the team implemented **automated validation scripts** that check for missing or incorrect table references in the code.

Additionally, the team decided to perform **peer code reviews** before merging updates into the live system, ensuring that multiple members validate changes. The Quality Assurance (QA) team also added extra test cases to include cross-module data verification as part of the standard testing process.

Finally, all changes and findings from this RCA have been documented and communicated to the entire project team. These actions will help improve the accuracy, reliability, and performance of the TNVS Administrative System.