training - Bug #2633

Create Error Analytics Report for Stations and Machines

05/20/2026 02:37 AM - Prashant Jain

Status:	New	**Start date:**	05/20/2026
Priority:	Normal	**Due date:**	
Assignee:	Yashaditya Singh	**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:		**Spent time:**	31.00 hours

Description

Description

Create a new report module to track and analyze charging session errors across stations and machines.

The report should allow users to:

Filter data by:
Station
Machine/Charge Point
Date range
View a summary table containing:
Error Code
Error Description (if available)
Total Error Count
When the user clicks on the error count, the system should open a detailed view displaying all sessions where that specific error occurred.

Report Requirements
Filters
Station selector
Machine/Charge Point selector
Date range filter
Summary Report Columns
Error Code
Error Message/Description
Error Count
Drill-down Details
On clicking the error count:
Open session-level details
Display:
Session ID
Station Name
Machine/Connector
Start Time
End Time
Vehicle/User (if available)
Error Code
Error Message
Expected Behavior
Machine dropdown should be dependent on selected station
Error counts should be aggregated based on filtered data
Clicking the count should support pagination for large datasets
Purpose

This report will help operations and support teams identify frequently occurring charger issues and analyze affected sessions for troubleshooting and maintenance.

History

#1 - 05/20/2026 09:11 AM - Yashaditya Singh

Analyzed the complete requirements for the Charging Session Error Report module and implemented all required backend APIs for the assigned ticket. I completed the backend development for summary and drill-down reports, including station-wise and machine-wise filtering, date range

filtering, aggregated error count generation, pagination support for detailed session data, and dependent machine selection based on station. I also verified the response structures for session-level error details, including session information, charger details, timestamps, vehicle/user data, and error messages to ensure smooth frontend integration and operational troubleshooting support.

#2 - 05/21/2026 09:11 AM - Yashaditya Singh

Completed backend development for the Charging Session Error Report module, including summary and detailed report APIs with station, machine, and date range filters. Implemented error-wise aggregation, drill-down functionality for session-level details, pagination support, and CSV export APIs for downloadable reports. Also tested all APIs with multiple filter combinations and validated the response structure successfully.

#3 - 05/21/2026 09:37 AM - Shreya Agarwal

implemented the Charging Error Report module from scratch, following the UI/UX patterns of existing reports like Charger Health Score and Station Charger Utilization. The module now includes station and charger dropdowns (with station as mandatory), date range filters, and a summary table displaying charger name, error code, description, and error count with clickable eye icons for drill-down. The details modal shows session-level information with working pagination. All API integrations for summary and details endpoints are complete with proper error handling and retry logic. However, the export functionality (Export Summary and Export Details buttons) is still pending - the buttons are added to the UI but the file download is not yet working.

#4 - 05/22/2026 09:11 AM - Shreya Agarwal

Today, I integrated the Charging Error Report with station and machine filters, date range picker, summary table showing charger name, error code, description, count, and action button, along with a details modal that displays session-level information with pagination. also added the export confirmation dialog and CSV conversion helpers to handle the JSON responses from the export APIs. However, the export functionality remains incomplete—the downloaded files are still not opening correctly due to a persistent file format mismatch (the API returns JSON but the code attempts to save it as Excel). I need to finalize the proper extraction of the nested items array and ensure the CSV conversion runs reliably for both summary and detail exports.