Codebase Changes Documentation

Power Pipeline Dashboard

Version Comparison Report

Date: February 4, 2026

Table of Contents

1. Summary Overview

This document outlines all changes made to the **power-pipeline-dashboard** codebase compared to the previous version (power-pipeline-dashboard-main-old).

1.1 Score Persistence Fix (HIGH PRIORITY)

> **Problem:** Calculated scores (Plant COD, Fuel, Capacity Size, etc.) were displayed in the preview but **never saved to the database** when creating a project.

Root Cause: handleAddSiteSubmit in DashboardContent.jsx only sent raw values, not the calculated scores from SCORE_MAPPINGS.

Files Modified:

File	Changes
src/DashboardContent.jsx	Import SCORE_MAPPINGS, calculate scores in handleAddSiteSubmit, add poi_voltage_kv
backend/models/projectModel.js	Make M&A Tier mapping case-insensitive
backend/migrations/002_add_score_columns.sql	NEW: Migration to add capacity_size and fuel_score columns

Scores Now Saved to Database:

Score Field	Calculated From	Rules
plant_cod	Legacy COD year	3 if <2000, 2 if 2000-2005, 1 if >2005
capacity_size	MW	1 if >50MW (individual) or >150MW (portfolio), else 0
fuel_score	Fuel Type	1 for Gas/Oil, 0 for Solar/Wind/Coal/BESS
capacity_factor	CF%	3 if <10%, 2 if 10-25%, 1 if 25-100%
markets	ISO/RTO	3=PJM/NYISO/ISO-NE, 2=MISO North/SERC, 1=SPP/MISO South, 0=ERCOT/WECC/CA
transactability_scores	Transactability	3=Bilateral developed, 2=Bilateral new/<10, 1=Competitive >10

1.2 Auto-Scoring Rules

Added real-time score calculation and preview to the Add New Project modal.

New Scoring Functions:

- **Capacity Size:** >50MW individual or >150MW portfolio = 1, else 0
- **Fuel Type:** Gas/Oil = 1, Solar/Wind/Coal/BESS = 0

Features Added:

- **Portfolio Checkbox:** Toggle between individual (>50MW) and portfolio (>150MW) thresholds
- **Real-Time Score Preview:** Shows 6 calculated scores as user fills form
- **Color-Coded Scores:** Green (high), yellow (medium), red (low), gray (N/A)

1.3 Files Changed Summary

New Files Added (7 files):

File Path	Description
backend/migrations/002_add_score_columns.sql	Migration for capacity_size and fuel_score columns
backend/scripts/checkTransactability.js	Script for validating transactability data
backend/scripts/importToSupabase.js	Script for importing data to Supabase
backend/scripts/runMigration.js	Database migration runner script
backend/utils/scoreCalculations.js	Backend score calculation utilities
src/utils/naValues.js	N/A value handling utilities
src/utils/scoreCalculations.js	Canonical score calculation functions

1.4 Major Implementation Changes

N/A Value Propagation System: When Excel data has missing values, the system now shows 'N/A' instead of '0'.

Expert Analysis Storage (Option B): Moved from separate expert_analysis table to storing current values in projects table with history in expert_analysis_history.

Edit Modal Data Population Fix: Added 8 missing fields (codename, acreage, fuel, etc.) to pipeline row object.

POI Voltage Management: Added max 5 limit, delete-then-insert pattern, and projectId-based fetch.

Score Calculation Fixes: Added null handling to prevent crashes when scores are null.

2. Backend Changes

2.1 M&A; Tier Case-Insensitivity Fix

File: backend/models/projectModel.js

> **Problem:** M&A; Tier dropdown values with different casing (e.g., 'Second Round' vs 'second round') weren't being mapped correctly to tier IDs.

Solution: Normalize M&A; Tier values to lowercase before mapping. Now 'Second Round', 'SECOND ROUND', and 'second round' all correctly map to ID 3.

2.2 New Database Migration

File: backend/migrations/002_add_score_columns.sql

Added **capacity_size** and **fuel_score** columns to store calculated scores.

```
ALTER TABLE pipeline_dashboard.projects ADD COLUMN IF NOT EXISTS capacity_size INTEGER; ALTER TABLE pipeline_dashboard.projects ADD COLUMN IF NOT EXISTS fuel_score INTEGER;
```

2.3 Backend Score Calculations

File: backend/utils/scoreCalculations.js

Added two new scoring functions:

- **calculateCapacitySizeScore(mw, isPortfolio):** Returns 1 if MW exceeds threshold, else 0
- **calculateFuelScore(fuel):** Returns 1 for Gas/Oil, 0 for other fuel types

2.4 Expert Analysis Storage (Option B)

File: backend/models/expertAnalysis.js

Aspect	Old Approach	New Approach
Storage Location	Separate expert_analysis table	projects table + expert_analysis_history
Query Pattern	Join from expert_analysis	Direct query from projects
Score Calculation	Frontend only	Server-side recalculation
History Tracking	None	Full audit trail in history table

2.5 Transmission Interconnection Changes

Delete-Then-Insert Pattern:

Changed from upsert (ON CONFLICT) to delete-all-then-insert to properly handle entry removals.

Max 5 Entries Enforcement:

Server-side validation now throws an error if more than 5 POI voltage entries are submitted.

Fetch by ProjectId:

Improved getTransmissionInterconnectionByProjectId with better type handling.

2.6 Controller Updates

File: backend/controllers/expertAnalysisController.js

Transmission fetch now supports both **?project=name** and **?projectId=123** query parameters, with projectId being preferred for reliability.

2.7 New Routes

File: backend/routes/expertAnalysisRoutes.js

Added edit history endpoints:

- GET /expert-analysis/history
- GET /expert-analysis/history/:historyId

3. Frontend Changes

3.1 Score Persistence in DashboardContent.jsx

Problem: Calculated scores were displayed in AddSiteModal preview but never saved to the database.

Changes Made:

- Added import for SCORE_MAPPINGS from constants
- Added poi_voltage_kv to initial state, reset state, and numericFields array
- Added score calculations in handleAddSiteSubmit before API call

Data Flow After Fix:

Step	Description
1	User fills form - raw values stored in newSiteData
2	handleAddSiteSubmit() builds cleanSiteData from raw values
3	SCORE_MAPPINGS calculates derived scores (e.g., 'Gas' -> fuel_score=1)
4	Both raw values AND calculated scores sent to backend
5	Database stores everything correctly

3.2 AddSiteModal Auto-Scoring Preview

File: src/components/Modals/AddSiteModal.jsx

New Features:

- **Portfolio Checkbox:** Toggle between individual (>50MW) and portfolio (>150MW) thresholds
- **ScoreItem Component:** Helper component for displaying color-coded scores
- **Real-Time Calculation:** useEffect hook recalculates scores on every form change
- **Preview Section:** Grid showing 6 calculated scores (Capacity Size, Fuel, Unit COD, CF, Markets, Transactability)

3.3 Edit Modal Data Population Fix

File: src/utils/calculations.js

Problem: Fields like codename, acreage, fuel were only in detailData, not at top level. The edit modal mapped row.codename which was always empty.

Solution: Added 8 missing fields to pipeline row object:

Field	Source Column
codename	projectCodenameCol
acreage	siteAcreageCol
fuel	fuelCol
markets	marketsCol
process	allColumns.processCol
gasReference	gasReferenceCol
colocateRepower	coLocateRepowerCol
contact	contactCol

3.4 Null Score Handling

File: src/utils/scoring.js

Problem: Calling .toFixed() on null crashed the function, causing Expert Analysis panel to show 'No Projects Found'.

Solution: Added null checks before formatting scores. If score is null, display '0.0' instead of crashing.

3.5 ExpertAnalysisModal Updates

File: src/components/Modals/ExpertAnalysisModal.jsx

Feature	Change
Max 5 POI Limit	Added check before adding new entries; shows alert if at limit
Add Button UI	Now shows count (e.g., '3/5 entries') and disables at max
Fetch by ProjectId	Changed from project name to projectId for reliable data retrieval
Always Save	Now saves even empty arrays to properly handle deletions

3.6 PipelineTable Field Mapping

File: src/components/Pipeline/PipelineTable.jsx

The edit button click handler mappings now work because the fields exist at the top level of the row object after the calculations.js fix.

4. Score Calculations

The score calculation system was completely refactored to use a single source of truth, properly handle N/A value propagation, and match the exact Excel formulas.

4.1 SCORE_MAPPINGS Functions

File: src/constants/index.jsx

capacitySize(mw, isPortfolio):

Type	Threshold	>Threshold	<=Threshold
Individual	50 MW	1	0
Portfolio	150 MW	1	0

fuelType(fuel):

Fuel Type	Score
Gas	1
Oil	1
Solar	0
Wind	0
Coal	0
BESS	0

transactability(type) - Fixed:

Now properly maps dropdown numeric values (1, 2, 3) to inverted scores:

Dropdown Value	Label	Score
1	Bilateral w/ developed relationship	3
2	Bilateral new/Process <10 bidders	2
3	Competitive >10 bidders	1

4.2 Complete Scoring Rules Reference

Field	Values	Score
Capacity Size	>50MW individual or >150MW portfolio	1
Capacity Size	<=50MW individual or <=150MW portfolio	0
Fuel	Gas, Oil	1
Fuel	Solar, Wind, Coal, BESS	0
Unit COD	<2000	3
Unit COD	2000-2005	2
Unit COD	>2005	1
Capacity Factor	<10%	3
Capacity Factor	10-25%	2
Capacity Factor	25-100%	1
Markets	PJM, NYISO, ISO-NE	3
Markets	MISO North, SERC	2
Markets	SPP, MISO South	1
Markets	ERCOT, WECC, CAISO	0
Transactability	Bilateral w/ developed relationship	3
Transactability	Bilateral new/Process <10 bidders	2
Transactability	Competitive Process >10 bidders	1
Thermal Optimization	Readily apparent value add	2
Thermal Optimization	No identifiable value add	1
Thermal Optimization	Yet to be saved	0

4.3 N/A Value Handling

File: src/utils/naValues.js

Utility Functions:

- **isNA(value):** Checks if value is N/A (null, undefined, '', '#N/A', '#VALUE!')

- **parseNullableNumber(value):** Returns null for missing values (not 0)
- **formatScoreDisplay(value):** Shows 'N/A' for null values
- **hasAnyNA(...values):** Checks if any value is N/A

4.4 Score Calculation Formulas

File: src/utils/scoreCalculations.js

Thermal Operating Score:

```
(COD x 0.20) + (Markets x 0.30) + (Transactability x 0.30) + (ThermalOpt x 0.05) + (Environmental x 0.15)
```

Redevelopment Score:

```
IF(any of Market/Infra/IX = 0, 0, (Market x 0.40 + Infra x 0.30 + IX x 0.30) x multiplier) where multiplier = 0.75 if "Repower", else 1
```

Overall Project Score:

```
Thermal Operating Score + Redevelopment Score
```

4.5 N/A Propagation Rules

Field	If Missing...
plant_cod	Thermal score = N/A
markets	Thermal score = N/A
transactability_scores	Thermal score = N/A
environmental_score	Thermal score = N/A
thermal_optimization	Defaults to 0 (exception)
market_score	Redevelopment score = N/A
infra	Redevelopment score = N/A
ix	Redevelopment score = N/A
co_locate_repower	Defaults to multiplier 1
Thermal score = N/A	Overall score = N/A
Redevelopment score = N/A	Overall score = N/A

4.6 Overall Rating Thresholds

Overall Score	Rating
>= 4.5	Strong
3.0 - 4.49	Moderate
< 3.0	Weak
N/A	N/A