| **Power BI Dashboards Test Plan / DQ Strategy** |
| --- |
| DQ Strategy |



| REVISION HISTORY | | | | | |
| --- | --- | --- | --- | --- | --- |
| Ver. | Description of Change | Author | Date | Approved | |
| Name | Effective Date |
| 0.1 | Initial Draft | Gabriela Cretu | 12-09-2025 |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

|  | |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

| **Abbreviations and Acronyms** | |
| --- | --- |
| EDW | Enterprise Data Warehouse |
| ETL | Extract, Transform, Load |
| LND | Landing |
| MSTR | MicroStrategy Business Intelligence tool |
| LDAP | Lightweight Directory Access Protocol |
| BI | Business Intelligence |
| DQ | Data Quality |
| UAT | User Acceptance Test |
| GUI | Graphical User Interface |
| TBD | To Be Defined |
| SLA | Service-Level Agreement |
| DAX | Data analysis expressions |
| KPI | Key performance indicator |
| RLS | Row level security |
|  |  |
|  |  |

**Table Of Conetents**

[**INTRODUCTION**](#_heading=h.l74xl9tnqkzi) **5**

[Document Goal and Scope](#_heading=h.6o53wlz0vwk) 5

[Target audience](#_heading=h.r5265rmzc947) 5

[**OVERALL STRATEGY**](#_heading=h.a5zdpjwcqfj0) **5**

[**SCOPE OF WORK**](#_heading=) **6**

[Entry criteria](#_heading=h.ev0obbhlqoo) 7

[Risks](#_heading=h.s1j6y0hn1mas) 7

[**TEST APPROACH**](#_heading=h.x754dsdj133v) **8**

[Functional Testing](#_heading=h.q3394q139rrf) 8

[DATA VALIDATION](#_heading=h.2ns2ri0y6n7) 8

[Data Completeness Verification](#_heading=h.w6yyfosb1gg6) 8

[Data Transformation Verification](#_heading=h.f6gn1rkax9iy) 9

[Data Quality Verification](#_heading=h.ww047k1mtwic) 11

[REPORTS VERIFICATION](#_heading=h.mn3jukri86fj) 12

[Report Functionality Verification](#_heading=h.42y1h24a0ezd) 12

[Report Graphical User Interface (GUI) Verification](#_heading=h.tpl67h7ir8pc) 13

[Ad hoc Verification](#_heading=h.qx2yxi9v5fve) 14

[Non-Functional Testing](#_heading=h.56idgz80rv0j) 16

[SECURITY TESTING](#_heading=h.22hgdyl0wgus) 16

[PERFORMANCE TESTING](#_heading=h.unr1fvdwgv8x) 17

[User Acceptance Testing (UAT)](#_heading=h.780nchj5f8qg) 19

[Regression Testing](#_heading=h.gcizof3cta5n) 20

[\*Critical Regression Scenarios for Power BI Dashboards](#_heading=h.cbim38nv86p6) 22

[Sales Dashboard](#_heading=h.aar6t4vkgne9) 22

[Cost Dashboard](#_heading=h.4nmt0neh4r37) 22

[Stocks Dashboard](#_heading=h.kefu3urjh3ju) 23

[Test Levels](#_heading=h.m9d40kn7tboy) 23

[SMOKE TEST](#_heading=h.17mjd66p76bq) 23

[CRITICAL PATH TEST](#_heading=h.kpos92fcz24) 24

[EXTENDED TEST](#_heading=h.49w13imj2rxw) 24

[Defect Tracking](#_heading=h.5zs2kyvvbgol) 25

[ISSUE TYPES](#_heading=h.9c8yetk74iri) 25

[BUG SEVERITY DEFINITION](#_heading=h.kohdiu99r7do) 25

[BUG PRIORITY DEFINITION](#_heading=h.q4w52op2d7kl) 25

[ISSUES NAMING CONVENTION](#_heading=h.8ldhlpx51c9k) 26

[BUG WORKFLOW](#_heading=h.2axk55sezmb2) 26

# 

# 

# 

# INTRODUCTION

## Document Goal and Scope

**Goal:** The goal of this document is to define the testing strategy and approach for three Power BI dashboards: **Sales**, **Cost**, and **Stocks**. It provides guidance on what QA should test, how testing should be performed, what tools and acceptance criteria to use, and how risks will be managed.

**Scope:**

QA testing will focus on **dashboard-level validation**, including:

* **Data Validation:** Spot checks of totals, KPIs, and aggregations to ensure data correctness.
* **Functional Testing:** Filters, slicers, drilldowns, drill-throughs, exports, and interactivity.
* **GUI / Usability:** Layout, fonts, colors, responsiveness, tooltips, and labels.
* **Non-Functional Testing:** Performance (load times), security (role-based access), and compatibility.
* **User Acceptance Testing (UAT):** Business users validate that dashboards meet their needs.
* **Optional DAX Verification:** Limited checking of measures/formulas for key calculations where access is available.

**Out of Scope:**

* Backend ETL, database logic, or source system validation beyond dashboard spot checks.
* Full reconciliation of underlying data pipelines (unless spot-checked via DAX).

## Target audience

The document is to be used by:

* **QA Team** – guidelines for what to test, how to test, what tools to use, and acceptance criteria.
* **Customer Team** – ensures all essential testing requirements are addressed and strategies/techniques cover business needs.

# OVERALL STRATEGY

QA will follow a **primarily black-box testing approach** for the Power BI dashboards, focusing on **end-user validation** rather than ETL or backend code. Key principles:

* Validate dashboards as delivered to business users.
* Focus on **data correctness, functionality, usability, performance, and security**.
* Optional DAX verification for critical calculations if access is available.
* Testing is **risk-based and sequential**, prioritizing critical KPIs and key features first.
* Use **checklist-driven testing** with defined acceptance criteria for each dashboard.

# SCOPE OF WORK

The scope defines **what is included and excluded** from QA activities:

*Included:*

* **Sales Dashboard:** KPIs (total sales, sales by region, trends), filters (time, product, region), spot-check DAX calculations if available.
* **Cost Dashboard:** KPIs (total cost, cost breakdown, variance), filters (department, time), spot-check DAX measures.
* **Stocks Dashboard:** KPIs (inventory levels, stock movements, shortages), filters (warehouse, product), optional DAX checks.
* Functional elements: filters, slicers, drilldowns, drill-throughs, exports.
* GUI/UX: layout, formatting, responsiveness, tooltips, labels.
* Non-functional: performance (load times), security (role-based access), compatibility.
* UAT validation by business users.

*Excluded:*

* Backend ETL processes, database logic, or source system validation (except spot checks).
* Full reconciliation of underlying data pipelines (unless verified via DAX).
* Any development-level debugging or code changes.

## 

## 

## 

## 

## 

## Entry criteria

* Dashboards deployed in Power BI Test workspace
* Test EDW data available
* Mapping documents / reference values approved
* Test environment stable and accessible
* User roles and permissions configured

## Risks

| # | Risk | Severity | Description | Resolution |
| --- | --- | --- | --- | --- |
| 1 | Incomplete specifications | High | QA cannot confirm correctness without finalized KPIs | Require sign-off before testing |
| 2 | Test Data Mismatch | Medium | EDW test data inconsistent with expected results | Refresh EDW test data before execution |
| 3 | Slow performance | Medium | Dashboards may load slowly, affecting UAT; complex DAX can exacerbate load times | Define SLA ; escalate if exceeded |
| 4 | Incorrect role-based access | High | Users may see unauthorized data | Validate security roles early |
| 5 | Scope changes mid-sprint | High | Additional features or new DAX measures delay testing | Control via change requests/ reprioritization |
| 6 | DAX calculations errors | Medium | Incorrect DAX measures may lead to wrong KPI s | Spot-check critical measures; escalate to BI devs if discrepancies found |

# 

# TEST APPROACH

## Functional Testing

### DATA VALIDATION

Poorly implemented data processing from the Enterprise Data Warehouse (EDW) to Power BI may cause the dashboards to become ineffective in answering business questions.

To ensure high data quality, validation of the **Sales, Cost, and Stocks dashboards** will be carried out using:

* **Data Completeness Verification**
* **Data Transformation Verification**
* **Data Quality Verification**

Comparisons will be performed between **EDW SQL queries and Power BI dashboards**, with DAX formulas reviewed where accessible. If data discrepancies are found, additional checks will be done against LND–EDW mappings.

#### ***Data Completeness Verification***

**Objectives**  Confirm that all expected data from EDW is fully loaded into Power BI datasets, with no records or values missing.

**Scope**

* **Sales dashboard**: transactions, revenue, units, customers
* **Cost dashboard**: expenses, allocations, cost centers
* **Stocks dashboard**: stock levels, movements, inventory by product/location

**Test Case Techniques**

* Compare row counts between EDW fact/dimension tables and Power BI datasets
* Validate that all mandatory attributes (e.g., Product ID, Customer ID, Cost Center) are present
* Spot-check totals (e.g., monthly sales, total cost, inventory balance) between EDW queries and Power BI visuals
* Verify filter/prompt values (e.g., full list of fiscal years, regions, product categories)
* If mismatches arise, compare LND vs EDW data for missing/incorrect records

**Verification Technique**

* SQL queries in EDW
* Manual comparison of Power BI dashboards vs SQL output (Excel export where useful)

**Entry Criteria**

* EDW–Power BI mapping completed
* Data loaded into EDW test environment
* Power BI dashboards deployed to test workspace
* DQ checklist prepared
* Smoke Test passed

**Quality & Acceptance Criteria**

* Row counts and totals match between EDW and Power BI
* All filter/prompt values available in Power BI
* No missing records in Sales, Cost, or Stocks dashboards

#### ***Data Transformation Verification***

**Objectives** Ensure that transformations (aggregations, business rules, DAX measures) are applied correctly in Power BI for Sales, Cost, and Stocks dashboards.

**Scope**

* **Sales**: monthly totals, year-to-date revenue, average selling price, margin %
* **Cost**: cumulative cost, allocations, cost per unit
* **Stocks**: stock balances, inflows vs outflows, turnover ratios

**Test Case Techniques**

* Validate DAX measures by comparing with SQL calculations in EDW
* Check aggregations (e.g., Sales per month, Costs per cost center, Stock balances by warehouse)
* Spot-check cumulative values (e.g., YTD revenue vs EDW, running totals of costs)
* If available, compare outputs against legacy Cognos/Excel baselines

**Verification Technique**

* SQL queries against EDW
* DAX formula review (where access is available)
* Manual comparison of Power BI dashboards with reference values

**Entry Criteria**

* EDW–Power BI mapping finalized
* DQ checklist prepared
* Data loaded in EDW test environment
* Dashboards deployed in test workspace
* Smoke Test passed

**Quality & Acceptance Criteria**

* Power BI aggregations match EDW SQL outputs
* DAX formulas align with business rules in functional specs
* No discrepancies in YTD, cumulative, or ratio calculations

### 

### 

#### ***Data Quality Verification***

**Objectives** Ensure that Sales, Cost, and Stocks dashboards use high-quality input data (correct data types, valid values, no duplicates, expected boundaries).

**Scope**

* **Sales**: no duplicate transactions, valid customer IDs, sales dates in correct ranges
* **Cost**: no negative costs (unless business rule allows), valid cost center codes
* **Stocks**: no negative stock balances (unless allowed), product IDs and locations valid

**Test Case Techniques**

* Verify primary keys (e.g., Transaction ID, Product ID, Cost Center ID) are unique
* Check null handling (e.g., missing customer, missing product → flagged/handled properly)
* Validate data types (e.g., dates are valid, quantities are numeric, costs are currency)
* Test data boundaries (e.g., no sales before system go-live, no future-dated costs, no stock > defined max)

**Verification Technique**

* SQL queries in EDW
* Spot-check Power BI visuals
* DAX queries to validate measures handle nulls, errors, and invalid values

**Entry Criteria**

* EDW–Power BI mapping finalized
* Data loaded into EDW test environment
* Dashboards deployed in test workspace
* DQ checklist prepared
* Smoke Test passed

**Quality & Acceptance Criteria**

* No duplicate or invalid records in dashboards
* Nulls handled per business rules (ignored, defaulted, or flagged)
* All data values within expected ranges

### REPORTS VERIFICATION

Reports verification of the **Sales, Cost, and Stocks dashboards** will focus on ensuring that all functionality and user interface elements in Power BI operate as expected, based on business requirements and design specifications.  
 Validation will cover two main areas:

* **Report Functionality Verification**
* **Report User Interface Verification**

Additionally, **Ad hoc verification** will confirm that measures and attributes can be flexibly combined without errors, supporting exploratory analysis by business users.

#### ***Report Functionality Verification***

**Objectives** Verify that all required functionalities of the dashboards (filters, slicers, drilldowns, drill-throughs, sorting, export) work correctly and align with specifications.

**Scope**

* **Sales Dashboard:** filters by region, product, time; drilldowns from totals to product-level; export to Excel/PDF.
* **Cost Dashboard:** filters by department, cost center, and time; drill-throughs to expense details; ability to view variances.
* **Stocks Dashboard:** filters by warehouse, product, date; drilldowns into stock movements; export functionality.

**Test Case Techniques**

* Verify all slicers, filters, and drilldowns behave as expected
* Validate sorting options on tables and visuals
* Confirm drill-through navigation works correctly (e.g., summary → detail → chart)
* Verify export options (Excel, PDF) work and data is consistent
* Test time-based options (e.g., year, month, day) return correct subsets

**Verification Technique** Manual testing in Power BI dashboards; SQL queries in EDW for spot checks.

**Entry Criteria**

* Functional and non-functional requirements approved
* DQ checklist prepared
* Test environment ready and dashboards deployed in Power BI Test workspace
* Smoke Test passed

**Quality & Acceptance Criteria**

* All defined report functionalities are implemented and work as described in specifications
* No blockers in navigation, filtering, or exporting

#### ***Report Graphical User Interface (GUI) Verification***

**Objectives** Ensure dashboards meet design standards (layout, fonts, colors, responsiveness) and provide a clear, intuitive user experience across supported browsers and screen resolutions.

**Scope**

* **Sales, Cost, Stocks dashboards:** consistency of themes, chart colors, legends, tooltips, labels, headers/footers
* Layout verification across desktop browser resolutions (Chrome, Edge, Firefox)
* Responsiveness checks for resizing and different screen widths

**Test Case Techniques**

* Validate dashboards display correctly in supported browsers/resolutions
* Check use of corporate color palette, fonts, headers/footers
* Verify correct alignment of visuals (charts, tables, slicers)
* Confirm tooltips and labels provide correct information
* Check that slicers, filters, and buttons are visible and usable without overlap/clipping

**Verification Technique** Manual verification in Power BI Service (with browser testing).

**Entry Criteria**

* Design requirements defined and approved
* Supported browsers/resolutions list finalized
* Test environment and dashboards available

**Baseline**

* Design specification (style guide, dashboard mockups)

**Quality & Acceptance Criteria**

* Dashboards match design specifications (colors, fonts, layout)
* No rendering issues in supported browsers/resolutions

#### ***Ad hoc Verification***

**Objectives** Confirm that measures, attributes, and hierarchies can be combined flexibly without errors in Power BI, supporting exploratory analysis and confirming model integrity.

**Scope**

* **Sales Dashboard:** measures (e.g., revenue, units, margin %) combined with attributes (e.g., region, product, customer)
* **Cost Dashboard:** measures (e.g., total cost, variance, allocation %) combined with cost centers, departments, periods
* **Stocks Dashboard:** measures (e.g., stock levels, movements, turnover ratio) combined with product, warehouse, date hierarchies

**Test Case Techniques**

* Test combining each measure with multiple attributes (e.g., revenue by region, by product, by month)
* Validate measures aggregate correctly across different hierarchies (e.g., sales YTD by region → total = global YTD)
* Check attributes from different dimensions interact correctly with measures (e.g., cost center vs period vs allocation %)
* Validate hierarchies (e.g., Date: Year → Quarter → Month → Day) drill correctly
* Spot-check results against EDW SQL outputs for complex joins

**Verification Technique** Manual ad hoc queries in Power BI (table visual testing, DAX Studio if available); SQL for validation.

**Entry Criteria**

* EDW–Power BI mapping finalized
* Data model (bus matrix / star schema) confirmed
* Dashboards deployed in test workspace
* DQ checklist prepared
* Smoke Test passed

**Quality & Acceptance Criteria**

* All measures calculate correctly when combined with attributes
* No errors when drilling across hierarchies or joining dimensions
* Data displayed matches EDW queries

## Non-Functional Testing

Non-functional testing is the verification of quality characteristics of the reporting solution. It refers to aspects that are not tied to specific user actions but affect the overall usability and reliability of the dashboards, such as **security, performance, usability, and compatibility**.

For the **Sales, Cost, and Stocks Power BI dashboards**, **security** and **performance** testing will be conducted as part of non-functional testing.

### SECURITY TESTING

**Objectives** Security testing ensures that only authorized users can access the dashboards and that data is restricted according to user roles. It also confirms that the dashboards are protected against unauthorized access and follow organizational security standards.

**Security Testing Requirements**

1. Only defined users should have access to the Power BI dashboards (user list maintained by admins).
2. Users are grouped by business roles (e.g., Sales team, Finance/Cost team, Supply Chain/Stocks team). Each group should only access their relevant dashboards or restricted data views.
3. All users have “read-only” access (view and interact with dashboards), not edit rights, unless explicitly authorized.
4. Row-level security (RLS) must ensure that users only see data relevant to their group (e.g., a regional sales manager sees only their region).
5. Sessions should respect Power BI Service security controls (e.g., authentication via Azure AD/LDAP, timeout settings).

**Scope**

* **Access level verification** (dashboard/workspace access, RLS validation).
* **Browser/session security verification** (encryption, session timeout, unauthorized access attempts).

**Test Case Techniques**

* Verify only authorized users can open dashboards.
* Confirm user groups only access their allowed dashboards (e.g., Finance cannot access Sales dashboard).
* Validate RLS restrictions (e.g., user in EU sales region cannot view APAC sales data).
* Attempt to access dashboards with unauthorized accounts → access denied.
* Test session timeout behavior.
* Confirm password/session encryption via browser developer tools.

**Verification Technique** Manual testing in Power BI Service with different test accounts/roles.

**Entry Criteria**

* Security requirements (user roles, RLS definitions) approved.
* DQ Checklist finalized.
* Dashboards deployed in Power BI Test workspace.
* User accounts and roles configured.

**Quality & Acceptance Criteria**

* Only defined users/groups can access dashboards.
* Unauthorized users cannot access dashboards or restricted data.
* Role-based restrictions (RLS) work as expected.

## PERFORMANCE TESTING

**Objectives** Performance testing ensures that dashboards load and refresh within acceptable timeframes, providing a usable experience for business users.

**Scope**

* **Sales dashboard:** test with commonly used filters (e.g., time, region) and full data volume.
* **Cost dashboard:** test with allocations, variances, and department-level filtering.
* **Stocks dashboard:** test with large product/location combinations and stock movement queries.

**Assumptions**

* No formal SLAs yet; testing will establish baseline performance metrics.
* Industry best practice: dashboards should load in **<10 seconds for typical queries**, **<30 seconds for maximum data volumes**.

**Test Case Techniques**

* Measure load time with typical filter combinations (e.g., monthly sales by region).
* Measure load time under maximum data volume (e.g., all products, full history).
* Validate drilldowns and drill-through performance (e.g., from Sales summary → product detail).
* Test exports (Excel/PDF) for responsiveness and data completeness.

**Verification Technique**

* Use Power BI Service performance analyzer.
* Browser developer tools to measure load/render times.
* SQL queries to confirm heavy loads return expected data volume.

**Entry Criteria**

* Performance requirements (baseline or SLA) documented.
* Dashboards deployed in Power BI Test workspace.
* DQ checklist prepared.

**Quality & Acceptance Criteria**

* Dashboards load within agreed thresholds (baseline <10 sec, heavy load <30 sec).
* Drilldowns, filters, and exports perform without errors or timeouts.
* Performance consistent across supported browsers.

## User Acceptance Testing (UAT)

The **Sales, Cost, and Stocks dashboards** are built on top of the Enterprise Data Warehouse (EDW) and are intended to provide business users with accurate, timely, and actionable insights. Since business users are the ultimate stakeholders, **User Acceptance Testing (UAT)** ensures that dashboards meet their expectations and can be trusted for decision-making.

**Objectives** UAT will confirm that the dashboards:

* Provide **correct and reliable data** (KPIs, metrics, aggregations).
* Offer **user-friendly and intuitive interfaces** (filters, slicers, drilldowns, exports).
* Meet **delivery expectations** in terms of performance, accessibility, and usability.

**Scope**  UAT will cover all three dashboards individually:

* **Sales Dashboard:** validation of revenue, sales trends, regional/product breakdowns.
* **Cost Dashboard:** validation of cost allocations, variances, and department-level analysis.
* **Stocks Dashboard:** validation of stock balances, movements, shortages, and warehouse-level breakdowns.

**What UAT Will Check**

* **Correctness of data:** spot-check results against EDW queries or reference reports.
* **Graphical usability:** layout, navigation, filters, slicers, drilldowns, tooltips, and exports.
* **Performance and delivery:** dashboards load in acceptable timeframes, and exports (Excel/PDF) are accurate.

**Execution Approach**

* UAT will be conducted by business users who are subject matter experts in Sales, Finance (Costs), and Supply Chain (Stocks).
* QA will provide UAT test checklists and facilitate walkthrough sessions.
* Each dashboard area will enter UAT **only after functional and non-functional testing is completed** and accepted by QA.
* UAT execution will follow the **approved project schedule** and be prioritized based on deliverables.

**Entry Criteria**

* Dashboards deployed to UAT workspace in Power BI Service.
* QA testing (functional + non-functional) passed.
* UAT checklist finalized.
* Business users trained (if required).

**Exit Criteria**

* All UAT test cases executed.
* No **high/critical** issues remain open.
* Business stakeholders provide formal sign-off.

**Quality & Acceptance Criteria**

* Business users confirm dashboards meet their needs (data accuracy, usability, performance).
* No blocking issues prevent dashboards from being used in production.

## Regression Testing

The BI ecosystem is subject to continuous changes such as:

* new data loads,
* changes in ETL pipelines,
* bug fixes or logic adjustments,
* software upgrades to the database, ETL engine, or BI platform.

Such changes can introduce unintended side effects that may corrupt or alter pre-existing information in the **EDW** or **Power BI dashboards (Sales, Cost, Stocks)**.

**Objectives** The goal of regression testing is to ensure that modifications at the EDW, ETL, or Power BI model level do not negatively affect existing dashboards or their functionalities.

**Scope** Regression testing will cover:

* The most frequently used datasets (filtered views commonly applied by users).
* SQL queries and transformations applied during data aggregations.
* Report components: filters, slicers, drilldowns, grids, graphs, and KPIs.
* Export functionality (Excel, PDF).
* Dashboard performance (load and refresh times).

**Test Case Techniques**

* Re-run critical test cases for each dashboard (Sales, Cost, Stocks) to validate **no data discrepancies** are introduced.
* Compare results from Power BI dashboards against SQL queries executed on the EDW (baseline validation).
* Validate that all modifications requested (new features or change requests) are deployed without breaking existing reports.
* Run confirmation testing for bug fixes to ensure previously reported defects are resolved without introducing regressions.
* Conduct smoke testing in the test environment after deployments to validate data availability, filters, and visuals.

**Verification Techniques**

* SQL test scripts against EDW for data validation.
* Power BI export comparison (Excel, PDF).
* Manual verification of visuals, filters, drilldowns.
* Automated dataset validation scripts (if available).

**Entry Criteria**

* Dashboards deployed to the test environment.
* Data quality checks for new/modified datasets passed.
* Change requests, bug fixes, or feature enhancements deployed to the test workspace.

**Quality & Acceptance Criteria**

* Comparison between baseline results (SQL queries, approved dashboards) and updated dashboards shows no unintended discrepancies.
* All high-priority user journeys (critical KPIs, filters, drilldowns, exports) function as before.
* No degradation in performance after deployment.

## ***\*Critical Regression Scenarios for Power BI Dashboards***

### Sales Dashboard

* Verify **Total Sales** at different aggregation levels (daily, monthly, yearly).
* Verify **Revenue by Product Category** and compare with SQL baseline.
* Validate **Sales by Region / Country / Branch** (with filters applied).
* Confirm **Top N Customers** and **Top N Products** visualizations.
* Drilldown functionality: e.g., from **Total Sales → Region → Customer → Invoice**.
* Exported data (Excel/PDF) matches the visualized data.

### Cost Dashboard

* Verify **Total Costs** by department, cost center, and time period.
* Validate **OPEX vs CAPEX** breakdown accuracy.
* Check **Unit Cost per Product / Service** against baseline SQL.
* Confirm **Cost allocation across departments** works with filters.
* Drillthrough from **Summary Costs** into **Detailed Transactions**.
* Export functionality maintains consistent totals and breakdowns.

### Stocks Dashboard

* Validate **Stock on Hand** per product, warehouse, and time period.
* Verify **Stock In vs Stock Out movements** match EDW records.
* Confirm **Stock Valuation** (cost × quantity) is accurate.
* Check **Aging Inventory** calculations (e.g., items >90 days).
* Drillthrough from **Total Stock** into **Warehouse-level** and **Product-level**.
* Exported stock balances (Excel/PDF) reconcile with SQL results.

**Acceptance Rule for All Dashboards**

* Numbers must match SQL queries executed on EDW within tolerance (e.g., rounding rules).
* Visuals, filters, slicers, drilldowns, and exports work as defined in specs.
* No new performance degradation (load/refresh times remain within limits).

## Test Levels

### SMOKE TEST

**Purpose:** Quickly assess whether the dashboards are ready for deeper testing.  
 **Scope:**

* Verify user access and permissions for dashboards.
* Check major functionalities of Sales, Cost, and Stocks dashboards, including:  
  + Filters, slicers, and drilldowns
  + Export to Excel/PDF
  + KPI totals and charts for the most commonly used datasets
* Basic GUI verification: default browser, screen resolution, layout, colors.
* Data consistency verification for key metrics using previously saved baseline datasets.

**Outcome:** If any major functionality or data discrepancy fails, QA notifies the team and testing is paused until a corrected version is available.

### CRITICAL PATH TEST

**Purpose:** Identify bugs affecting the most important functionality or datasets for end users.  
 **Scope:**

* Focus on core KPIs and metrics for each dashboard:  
  + **Sales:** Total sales, revenue by product/category, top customers/products
  + **Cost:** Total cost, department-level allocations, CAPEX/OPEX breakdown
  + **Stocks:** Stock on hand, stock movement, inventory valuation, aging items
* Verify key interactions: drilldowns, drill-throughs, filters, and summary calculations.
* Critical path tests are based on the defined **DQ Checklist** for each dashboard.

**Execution:** Manual testing according to the checklist, ensuring high-priority data and functionality behave correctly.

#### EXTENDED TEST

**Purpose:** Detect issues in non-typical, edge-case, or less frequent scenarios.  
 **Scope:**

* Invalid or unusual filter combinations.
* Boundary value testing (e.g., zero sales, negative adjustments, empty stock).
* Ad hoc joins between attributes and measures.
* Complex drilldowns or nested filters.
* Data and visual validation under unusual screen resolutions or browsers.

**Execution:** Conducted after Smoke and Critical Path tests, using ad hoc testing and business domain knowledge.

* Also serves as acceptance tests for the release candidate build.

## 

## Defect Tracking

A **defect (bug)** is any anomaly observed between the requirements specification/mapping document and the behavior of the Power BI dashboards (Sales, Cost, Stocks). All defects will be logged and tracked using **Jira**, and metrics/statistics will be included in QA reports.

### ISSUE TYPES

All defects in this project will be categorized as follows:

* **Bugs (BG):** Issues identified during testing of dashboards (data mismatches, visual errors, functional problems).
* **Issues (IS):** Problems that might block testing or dashboard usage. Typically higher priority than bugs.
* **Change Requests (CR):** Requests to modify or enhance existing dashboard functionality (e.g., adding a new slicer or measure).
* **New Features (NF):** Requests for entirely new dashboards, visualizations, or reports.

### BUG SEVERITY DEFINITION

Severity indicates the impact of the defect on dashboard reliability and usability:

* **Blocker:** Prevents testing or dashboard usage entirely.
* **Critical:** Major dashboard component (Sales, Cost, or Stocks) is inaccessible or produces completely incorrect data.
* **Major:** Core functionality/data is incorrect; no workaround available.
* **Minor:** Non-critical functionality issue; workaround exists.
* **Trivial:** Cosmetic/visual issue (colors, fonts, formatting) that does not impact functionality.

### BUG PRIORITY DEFINITION

Priority indicates how quickly the defect should be fixed, based on business needs:

* **Critical:** Must be fixed ASAP.
* **Major:** Important functionality; fix within a short timeframe.
* **Medium:** Fix after Critical and Major issues.
* **Minor:** Low-priority; fix after all other issues.

### ISSUES NAMING CONVENTION

Each issue name should reflect the dashboard and area affected:

**Structure:**

[Issue Type]: [Component]: [Dashboard/Area] (optional): [Item Description]

* **[Issue Type]** – Bug, Issue, CR, NF
* **[Component]** – EDW (for data load/ETL issues) or BI (for dashboards and visualizations)
* **[Dashboard/Area]** – Sales, Cost, Stocks, or specific report/visual within dashboard

**Examples:**

* BG: BI: Sales Dashboard: Total revenue calculation mismatch
* [CR] BI: Cost Dashboard: Add department-level filter

### BUG WORKFLOW

The standard Jira workflow for defects will be followed:

1. **TO DO:** Issue is logged in Jira.
2. **In Progress:** Development or BI team works on the fix.
3. **IN REVIEW:** Fixed issue is deployed to test environment.
4. **DONE:** Issue is resolved and no further action is needed.

This ensures all **Power BI dashboard issues** (Sales, Cost, Stocks) are tracked consistently, prioritized appropriately, and addressed in a structured manner.