|  |  |
| --- | --- |
| A blue sign with silver border  AI-generated content may be incorrect. | Product Requirements Document (PRD)SPC Station Health Charts |

**Version:** 1.4

**Date:** October 7, 2025

**Owner:** Technical Operations Analytics Team

**Status:** Delivered with Debug Tools

**Enhancement:** CSV Troubleshooting & Enhanced Error Reporting

## 1. Executive Summary

### Problem Statement

Airline technical operations teams need to monitor maintenance metrics across multiple stations to identify process variations and operational shifts. Current analysis methods involve manual Excel charting that:

* Lacks statistical rigor
* Doesn't detect process shifts consistently
* Requires extensive manual effort
* Cannot be easily shared or reproduced

### Solution

A lightweight, desktop-based Statistical Process Control (SPC) dashboard that:

* Runs locally without requiring web server infrastructure
* Implements industry-standard Wheeler's Rules for process control
* Auto-detects process phases and calculates control limits
* Exports publication-ready PNG charts
* Requires only Python (no admin rights needed)

### Success Metrics

* ✅ Deployment to all stations without IT support tickets
* ✅ Zero external dependencies (CDN, npm, web servers)
* ✅ Accurate phase detection using Wheeler's methodology
* ✅ Historical data coverage (Jan 2023 - present)
* ✅ Export capability for executive reporting

## 2. Business Requirements

### BR-01: Minimal Technical Barriers

**Priority:** P0 (Critical)  
**Rationale:** Users lack local admin rights and IT support is limited

**Requirements:**

* Must run with Python-only (version 3.7+)
* No npm, Node.js, or web server installation required
* Single-click launch via batch file
* Works on corporate locked-down Windows machines

**Acceptance Criteria:**

* User can extract zip and run without any additional installations
* No firewall/proxy issues (no CDN dependencies)
* Functions offline

### BR-02: Statistical Accuracy

**Priority:** P0 (Critical)  
**Rationale:** Decisions about maintenance process changes depend on accurate signal detection

**Requirements:**

* Implement Wheeler's XmR (Individuals and Moving Range) charts
* Use Natural Process Limits (2.66 sigma)
* Detect phases using Wheeler's Rules #1 and #4
* Calculate limits from baseline data, not entire dataset

**Acceptance Criteria:**

* Matches Wheeler's methodology exactly
* Detects realistic process shifts (not noise)
* Limits recalculate when true shifts occur
* Moving Range chart tracks process variation

### BR-03: Historical Data Coverage

**Priority:** P1 (High)  
**Rationale:** Need to analyze trends over 2+ years for pattern recognition

**Requirements:**

* Display data from January 2023 to current week
* Support 4 key maintenance metrics
* Track 3 major hub stations (AUS, DAL, HOU)

**Acceptance Criteria:**

* All charts show 140+ weeks of data
* Data updates weekly
* Historical patterns visible for comparison

### BR-04: Flexible Data Input

**Priority:** P1 (High)  
**Rationale:** Different data sources use different formats

**Requirements:**

* Auto-detect CSV format on upload
* Support Format A: timestamp,station,metric\_value (filename = measure)
* Support Format B: station,measure,date,value
* Map station names automatically (e.g., "Dallas" → "DAL")

**Acceptance Criteria:**

* Upload succeeds for both formats without user intervention
* Clear error messages if format is unsupported
* Filename becomes chart title for Format A

### BR-05: Chart Export

**Priority:** P1 (High)  
**Rationale:** Charts must be embedded in PowerPoint presentations and executive reports

**Requirements:**

* Save individual charts as PNG images
* High resolution suitable for printing
* Filename matches chart name
* One-click download per chart

**Acceptance Criteria:**

* PNG saves to local file system
* Image quality sufficient for 1920x1080 displays
* Transparent or white background

## 3. User Stories

### US-01: Maintenance Manager

**As a** maintenance station manager  
**I want to** quickly identify when my station's performance shifts  
**So that** I can investigate root causes immediately

**Acceptance Criteria:**

* Load latest data in < 5 seconds
* Visual indicators show phase boundaries
* Control limits update automatically when shifts occur

### US-02: Tech Ops Analyst

**As a** technical operations analyst  
**I want to** upload my own CSV data  
**So that** I can analyze custom metrics not in the standard set

**Acceptance Criteria:**

* Upload button accepts .csv files
* Auto-detects format
* Generates X and mR charts automatically

### US-03: Executive Reporting

**As a** senior operations director  
**I want to** export charts as images  
**So that** I can include them in monthly board presentations

**Acceptance Criteria:**

* Download button on each chart
* PNG format with clean styling
* Chart title included in image

### US-04: Field Technician (Limited IT Access)

**As a** field technician with no admin rights  
**I want to** run the dashboard on my corporate laptop  
**So that** I don't need to wait for IT to install software

**Acceptance Criteria:**

* Runs with only Python installed
* Double-click to launch
* No command-line expertise required

## 4. Functional Requirements

### FR-01: Data Loading

* **FR-01.1:** Load test data from /input folder via "Load Test Data" button
* **FR-01.2:** Upload individual CSV files via "Upload CSV" button
* **FR-01.3:** Display loading indicators during data processing
* **FR-01.4:** Show clear error messages if data format invalid

### FR-02: Chart Rendering

* **FR-02.1:** Display both X (Individuals) and mR (Moving Range) charts
* **FR-02.2:** Color-code data points (blue for normal, red for out-of-control)
* **FR-02.3:** Draw control limits (UCL, CL, LCL) as horizontal lines
* **FR-02.4:** Label phase boundaries visually
* **FR-02.5:** Auto-scale Y-axis based on data range

### FR-03: Phase Detection

* **FR-03.1:** Establish baseline with minimum 20 points
* **FR-03.2:** Calculate limits from baseline data only
* **FR-03.3:** Monitor for Rule #1: Point beyond control limits
* **FR-03.4:** Monitor for Rule #4: 8 consecutive points on one side of centerline
* **FR-03.5:** Start new phase when signal detected
* **FR-03.6:** Recalculate limits for new phase

### FR-04: Station Filtering

* **FR-04.1:** Dropdown to select individual station
* **FR-04.2:** "All Stations" option to view all
* **FR-04.3:** Charts update dynamically on selection change

### FR-05: Export

* **FR-05.1:** "Save PNG" button on each chart
* **FR-05.2:** Download triggers browser save dialog
* **FR-05.3:** Filename = sanitized chart title + .png

## 5. Non-Functional Requirements

|  |  |
| --- | --- |
| **Category** | **Requirement** |
| **Performance** | • Dashboard loads in < 3 seconds • Chart rendering completes in < 1 second per chart • CSV processing handles 10,000+ rows without lag |
| **Usability** | • No training required for basic usage • Instructions visible on welcome screen • Error messages actionable and clear |
| **Reliability** | • Handles missing data gracefully • Validates CSV format before processing • Doesn't crash on malformed input |
| **Maintainability** | • Pure Python + HTML/JavaScript (no frameworks) • Well-commented code • Modular architecture (separate concerns) |
| **Portability** | • Works on Windows, Mac, Linux • No hardcoded paths (relative paths only) • Python 3.7+ compatible |

## 6. Out of Scope (Explicitly Excluded)

### V1.0 Exclusions

* ❌ Database storage - CSV files only
* ❌ Multi-user collaboration - Single-user desktop tool
* ❌ Authentication/Authorization - Local tool, no security needed
* ❌ Real-time data updates - Manual refresh only
* ❌ Mobile support - Desktop browsers only
* ❌ Cloud deployment - Local only
* ❌ Advanced SPC rules - Only Wheeler's Rule #1 and #4
* ❌ Automated data collection - Manual CSV upload

### Future Considerations (V2.0+)

* 🔮 Additional Wheeler's Rules (#2: runs near limits, #3: trends)
* 🔮 Automated email reports
* 🔮 Comparison mode (station vs station)
* 🔮 Annotation/commenting on charts
* 🔮 Data export to Excel
* 🔮 Integration with airline ERP systems

## 7. Constraints & Assumptions

### Constraints

1. **No admin rights** - Users cannot install software
2. **Corporate firewall** - CDN access blocked (unpkg, jsdelivr, etc.)
3. **Python availability** - Python 3.7+ already installed enterprise-wide
4. **Weekly data granularity** - Daily not available from source systems

### Assumptions

1. Users have basic computer literacy (can extract zip files)
2. CSV data is reasonably clean (no extensive validation needed)
3. Python http.server module available (standard library)
4. Modern browser available (Chrome, Edge, Firefox - last 2 years)

## 8. Risks & Mitigations

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **Impact** | **Probability** | **Mitigation** |
| Python not installed | High | Low | Include Python check in launcher; provide install link |
| Port 8000 in use | Medium | Medium | Document how to change port in server.py |
| CSV format changes | Medium | Medium | Auto-detection logic; clear error messages |
| Browser compatibility | Medium | Low | Use standard HTML5 Canvas (widely supported) |
| Data quality issues | Medium | Medium | Input validation; graceful error handling |

## 9. Glossary

|  |  |
| --- | --- |
| **Term** | **Definition** |
| **SPC** | Statistical Process Control |
| **UCL/LCL** | Upper/Lower Control Limit |
| **CL** | Center Line (mean) |
| **XmR Chart** | Individuals (X) and Moving Range (mR) chart |
| **NPL** | Natural Process Limits (2.66 sigma) |
| **Wheeler's Rules** | Statistical rules for detecting special cause variation |
| **Phase** | Period of stable process performance with consistent limits |

## Approval

|  |  |  |  |
| --- | --- | --- | --- |
| **Role** | **Name** | **Date** | **Signature** |
| Product Owner | Tech Ops Manager | 2025-10-06 | ✅ Approved |
| Technical Lead | Development Team | 2025-10-06 | ✅ Delivered |
| End User Rep | Station Manager (AUS) | 2025-10-06 | ✅ Accepted |

**Document Version History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Changes** |
| 1.0 | 2025-10-06 | Development Team | Initial release |