

CSV Data Quality & Analysis Tool (C++)

This document explains the design, components, and engineering mindset behind the CSV Data Quality & Analysis Tool built in C++. The goal of this project was learning real-world C++ practices rather than building a complex product.

Project Motivation

In real systems, data is rarely clean. Before analysis or processing, systems must validate input, handle malformed records, and ensure data integrity. This project simulates that workflow in a clean and understandable manner.

High-Level Flow

- 1 Verify file integrity using checksum
- 2 Read and parse CSV data
- 3 Validate and clean malformed rows
- 4 Analyze clean data (min, max, average)
- 5 Generate a readable report

Component Responsibilities

CsvReader: Reads CSV files, skips headers, parses raw rows, and assigns validation status.

DataCleaner: Filters invalid rows and converts valid rows into clean typed data.

CsvAnalyzer: Computes statistics such as min, max, and average from clean data.

FileHasher: Computes a checksum to detect file modifications or corruption.

main.cpp: Orchestrates execution flow and coordinates all components.

Engineering Mindset

- 1 Fail fast on invalid input
- 2 Separate raw, validated, and analyzed data
- 3 Keep logic modular and testable
- 4 Prefer clarity over cleverness

Future Extensions

The project is intentionally simple but extensible. Possible future improvements include support for Excel files, configurable validation rules, unit tests, and performance optimizations.