

Developing a Self-Service Dashboard to Support Data Quality Monitoring Aligned with HESA Rules in Welsh Higher Education

Amina Meah | BSc (Hons) Computing (Data and Information Systems) | July 2026

1 Introduction and problem

HESA validation failures are used to quality-assure statutory student data returns. The existing process depended on a large current error extract plus separate assignments, notes, tolerance requests and target information. This made it difficult to identify priorities, retain operational history and move from headline totals to detailed evidence.

Project aim: create a self-service Power BI dashboard for monitoring and investigating HESA validation errors.

2 Aim and objectives

Aim:

To design, develop and evaluate a self-service Power BI dashboard that supports the monitoring and investigation of HESA validation errors.

Objectives:

- Analyse the existing HESA validation-error workflow and identify user requirements.
- Design a maintainable SQL Server and Power BI architecture.
- Develop overview, rule-level, record-level and student-code investigation pages.
- Preserve assignments, notes, tolerance requests, targets and archived error counts between data imports.
- Evaluate the dashboard through structured tasks, the System Usability Scale and qualitative user feedback.

3 Research basis

The design was informed by dashboard and BI literature. Few and Tufte emphasise clarity and reduced visual clutter [1], [2]. Howson describes self-service analytics through governed data [3]. Higher education BI research highlights integration, organisational readiness and data quality [4]. The evaluation used the System Usability Scale [6].

Key principle: organise evidence and support professional judgement, not replace it.

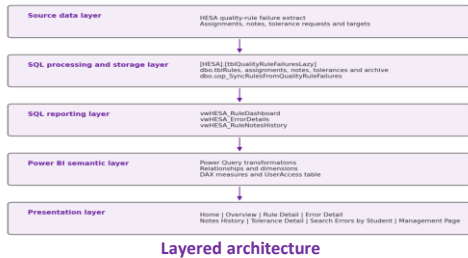
4 Methodology

A design science methodology was used because the project involved creating and evaluating a practical information-systems artefact. A pragmatic mixed-method approach combined technical testing, quantitative usability measures and qualitative feedback. Feature-Driven Development supported incremental delivery through small, testable features.

The project progressed from problem identification and requirements analysis to artefact design, construction, demonstration, testing and evaluation [7].

- Developer testing of SQL, Power Query, DAX, filters and navigation
- Ten structured dashboard tasks
- Five System Usability Scale questionnaire responses [6]
- Analysis of anonymised open-text feedback

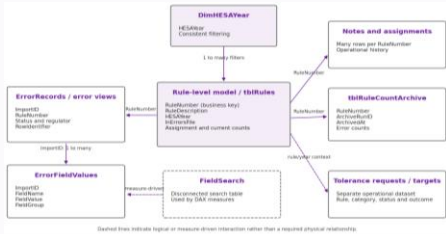
5 Technical design



Layered architecture

- Current snapshot:** the latest HESA validation-failure extract is imported into SQL Server.
- Persistent data:** assignments, notes, tolerances, targets and archived counts remain available between imports.
- Reporting layer:** SQL views, Power Query and DAX prepare governed datasets and interactive measures.

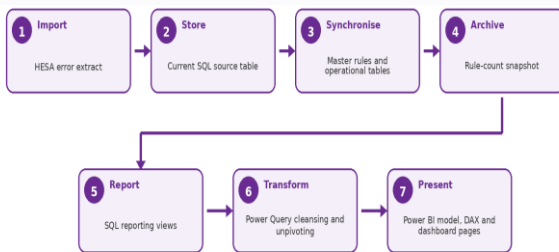
6 Data model decisions



Conceptual model and relationship paths

- RuleNumber used as the main business identifier
 - ImportID links ErrorRecords to unpivoted ErrorFieldValues
 - RowIdentifier retained for investigation but not treated as a student count
 - Student.OWNSTU used only on authorised search page
 - Archive grain = one row per rule, year and run
- These decisions avoided ambiguous relationships and prevented identifiers from being misinterpreted. Separating current errors from persistent operational data also ensured that notes and assignments were not lost when a failure disappeared from a later import.

7 Implementation flow



Dashboard workflow

- Import current HESA validation failures
- Synchronise rules and archive counts
- Preserve assignments, notes and tolerances
- Reshape populated error fields
- Refresh the Power BI model
- Monitor and investigate through dashboard pages

8 Dashboard artefact

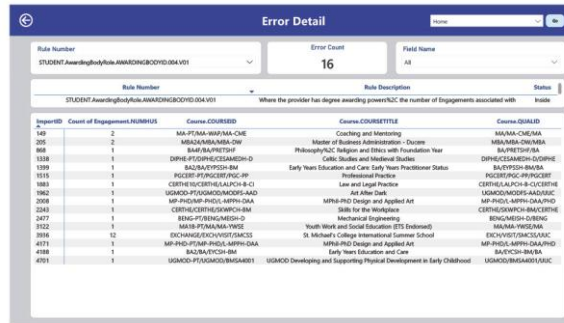
Eight task-based pages were developed. Key dashboard views are shown below.



Overview: KPIs, slicers and rule prioritisation

Consolidates 283,108 validation failures across 319 failing rules, with Inside/Outside breakdowns, assignment coverage, top rules and filters.

Overview: presents headline error totals, Inside and Outside status, assignment coverage, filters and the highest-volume failing rules.



Error Detail: populated fields for selected failures

Shows only populated HESA field values for selected error records, avoiding hundreds of blank source columns while retaining detail.

Error Detail: Displays only populated HESA fields for selected failures. ImportID provides a reliable technical key, while the searchable field selector reduces clutter and supports comparison across related records.



Management: targets, tolerances and archive monitoring

Summarises target rules, assigned workload, tolerance categories and archive-run trends for management-level oversight.

Management Page: Provides a strategic view of target-rule performance, assignment workload, tolerance categories and archived error counts. It separates management oversight from detailed operational investigation

9 Testing and results

88.5

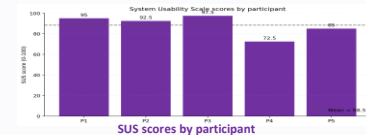
Mean SUS

4.6/5

Satisfaction

83.3%

Tasks without difficulty



Five questionnaire responses and four task sheets were returned; three task sheets contained complete ratings for all ten tasks.

10 Findings

- Users valued Overview and investigation pages
- Persistent tables protected notes and assignments
- Unpivoting made sparse source data usable
- Change icons required clearer explanation
- One genuine archive snapshot limited trend evidence

11 Technical contribution

Reusable SQL Server + Power BI pattern: separates the replaceable validation-error snapshot from persistent assignments, notes, tolerances, targets and archive counts, preserving operational context when resolved rules disappear from later imports This design is reusable because the same pattern could be adapted to other institutional data-quality processes that combine changing snapshots with persistent operational evidence.

12 Conclusion

The project substantially achieved its aim by producing a usable prototype linking current validation failures with persistent operational evidence. It supports movement from summary monitoring to rule, record, note, tolerance, student-search and management analysis.

Next steps: formal access control, clearer Change guidance, performance testing and additional archive snapshots.

13 References

- [1] Few, Information Dashboard Design, 2013.
 - [2] Tufte, The Visual Display of Quantitative Information, 2001.
 - [3] Howson, Successful Business Intelligence, 2013.
 - [4] Sequeira et al., Roadmap for implementing BI systems in HE, 2024.
 - [5] HESA, HE student data, 2026.
 - [6] Brooke, SUS usability scale, 1996.
 - [7] Peffers et al., Design Science Research Methodology, 2007.
- Tools: SQL Server, Power BI, Power Query, DAX and Microsoft Forms