SecureComply

An Explainable GDPR Compliance Auditing Tool for SME's

Enhancing GDPR compliance through explainable security assessment and data-driven insights

Final Year Project

Student Name: Cormac Casey

Student ID: C00283808

Course: Cybercrime & IT Security

Institution: South East Technological University

Supervisor: Omer Ali

1. Introduction

1.1 Project Overview

SecureComply is a command-line based GDPR compliance auditing tool developed to support Small and Medium Enterprises (SMEs) in assessing their data protection posture in a structured, accessible, and explainable manner. The system is designed to transform organisational input data into a quantified compliance score, accompanied by detailed insights and actionable recommendations.

At its core, SecureComply operates as a modular audit pipeline. The system accepts structured GDPR-related input data in JSON format, representing organisational practices across key areas such as security controls, transparency obligations, and internal governance processes. This data is processed through a series of stages, including ingestion, validation, compliance scoring, and report generation. Each stage performs a distinct function, ensuring that data is consistently prepared, verified, and evaluated before results are produced.

A key feature of the system is its deterministic rule-based scoring engine, which evaluates each input against predefined GDPR-aligned controls. This approach ensures that all outputs are consistent, reproducible, and fully explainable. Rather than relying on opaque or probabilistic methods, SecureComply provides control-level scoring with clear justifications, allowing users to understand exactly how their compliance score is derived.

The system generates a structured HTML report that presents the overall compliance score, category-level breakdowns, and prioritised recommendations for improvement. This report is designed to be accessible to non-technical users, translating complex compliance data into clear and actionable insights.

In addition to core functionality, SecureComply incorporates several advanced features to enhance realism and usability. These include optional host telemetry integration, which allows real system-level security indicators to be included as contextual information, and a benchmarking mechanism that compares results against a synthetic dataset of SME profiles. An optional AI-driven narrative module is also included, providing a high-level executive summary of the audit results without influencing the underlying scoring logic.

The overall design of SecureComply emphasises modularity, explainability, and real-world applicability. By combining structured data processing, deterministic evaluation, and user-focused reporting, the system provides SMEs with a practical and repeatable method for understanding and improving their GDPR compliance posture.

2. Research Context and Design Rationale

2.1 Compliance Assessment Approaches

Organisations use a range of methods to assess GDPR compliance, including manual auditing, automated rule-based systems, and more recently AI-assisted approaches. Each method differs in terms of cost, scalability, and level of insight. Understanding these approaches informed the design of SecureComply, particularly the decision to adopt a rule-based scoring model with optional AI support.

2.1.1 Manual Auditing

Manual auditing is the traditional approach to GDPR compliance assessment and involves human experts reviewing organisational practices against regulatory requirements. This typically includes analysing policies, reviewing security controls, and assessing how personal data is processed within the organisation.

The main strength of manual auditing is its ability to provide context-aware and detailed analysis. Human auditors can interpret complex scenarios and apply judgement where regulatory requirements are not strictly defined. This makes manual auditing highly effective in identifying nuanced compliance issues.

However, this approach presents several limitations, particularly for SMEs. Manual audits are time-consuming, costly, and require specialist expertise, making them less accessible to smaller organisations. In addition, they are not easily repeatable, meaning compliance is often assessed at a single point in time rather than continuously.

Another key issue is consistency and scalability. Different auditors may reach different conclusions, and the process does not scale well across multiple organisations or frequent assessments.

These limitations influenced the design of SecureComply, where a deterministic, rule-based approach was adopted to provide consistent and repeatable assessments. By automating key aspects of compliance evaluation, the system aims to reduce reliance on manual processes while still maintaining transparency in how scores are calculated.

2.1.2 Automated Rule-Based Systems

Automated rule-based systems provide a structured approach to assessing GDPR compliance by applying predefined rules to input data. These systems evaluate specific controls, such as encryption, access management, and data handling practices, and assign scores or classifications based on how well the input aligns with expected standards.

The primary advantage of rule-based systems is their consistency and repeatability. Unlike manual auditing, the same input will always produce the same output, ensuring that assessments are objective and not influenced by human interpretation. This makes them particularly suitable for SMEs, where access to expert auditors may be limited.

Rule-based systems are also scalable and efficient, allowing multiple assessments to be performed quickly with minimal resources. This supports more frequent evaluations, enabling organisations to monitor compliance over time rather than relying on one-off audits.

However, this approach has limitations. Rule-based systems rely on predefined logic and therefore may oversimplify complex regulatory requirements. GDPR often involves context-specific interpretation, which cannot always be fully captured through static rules. Additionally, the accuracy of results is dependent on the quality of input data, meaning incomplete or incorrect inputs can lead to misleading outcomes.

These characteristics directly influenced the design of SecureComply. The system uses a deterministic scoring engine, where each GDPR-related control is mapped to a defined rule and

associated score. This allows for transparent and explainable outputs, where users can clearly see how each input contributes to the overall compliance score.

To address the limitations of rule-based systems, SecureComply incorporates several enhancements. A validation layer ensures that inputs conform to expected formats and values, reducing the risk of invalid data affecting results. Additionally, the introduction of a data quality score highlights the completeness of input data, allowing users to understand the reliability of the assessment. Finally, an optional AI-generated narrative is included to provide higher-level interpretation, complementing the structured scoring approach.

Overall, the use of an automated rule-based system enables SecureComply to deliver consistent, scalable, and explainable GDPR assessments, while incorporating mechanisms to mitigate the inherent limitations of this approach.

2.1.3 AI-Assisted Analysis

AI-assisted analysis introduces the use of machine learning or large language models to enhance compliance assessments by generating insights, summaries, or recommendations based on structured input data. Unlike rule-based systems, which rely on predefined logic, AI systems can provide more contextual and human-readable interpretations of results.

The primary advantage of AI-assisted approaches is their ability to improve usability and communication of complex outputs. In the context of GDPR, where technical and legal concepts can be difficult to interpret, AI can translate structured audit results into clear, executive-level summaries. This is particularly useful for SMEs, where decision-makers may not have deep technical or regulatory expertise.

However, AI-assisted analysis also introduces challenges. Outputs may vary depending on prompts and model behaviour, leading to reduced consistency and explainability compared to deterministic systems. Additionally, reliance on external AI services introduces considerations such as cost, availability, and potential failure scenarios.

These factors directly influenced how AI was integrated into SecureComply. Rather than using AI for core compliance scoring, the system adopts a hybrid approach, where a deterministic rule-based engine performs all calculations, and AI is used only as an optional enhancement for generating a CISO-style executive summary.

This design ensures that the core audit remains fully transparent, repeatable, and explainable, while still benefiting from AI's ability to improve readability and provide higher-level insights. Importantly, SecureComply includes a fallback mechanism, where a deterministic summary is generated if the AI service is unavailable or an error occurs. This guarantees that the system remains functional and reliable under all conditions.

By limiting AI to a supporting role, SecureComply balances the strengths of AI-assisted analysis with the need for consistency and trust in compliance assessments. This approach enhances the usability of the tool without compromising the integrity of the underlying scoring model.

2.2 Design Considerations

The design of SecureComply was guided by a number of key considerations identified through both research and practical limitations observed in existing GDPR compliance solutions. These considerations focused on balancing accuracy, usability, explainability, and real-world applicability, particularly for SMEs.

A primary consideration was explainability. Many existing compliance tools provide high-level outputs without clearly demonstrating how results are derived. To address this, SecureComply was designed around a deterministic rule-based scoring engine, where each GDPR-related control is mapped to a clearly defined rule and associated score. This ensures that all outputs are transparent, allowing users to understand how individual inputs contribute to the overall compliance assessment. The inclusion of control-level justifications further reinforces this transparency.

Another key consideration was handling imperfect and incomplete data, which is a common issue for SMEs. As identified in earlier sections, organisations may not always have full visibility over their data processing activities. To address this, the system incorporates a data ingestion and normalisation process, which standardises inputs and assigns default values for missing data. This is supported by a validation engine, ensuring that only valid inputs are processed. Additionally, a data quality scoring mechanism was introduced to explicitly indicate the completeness of input data, allowing users to assess the reliability of the audit results. This directly addresses a major limitation in existing solutions, which often assume fully accurate input data.

Usability was also a critical design factor. Many enterprise-level tools are complex and require significant expertise to operate. SecureComply was designed as a lightweight command-line tool with clear help and usage options, enabling ease of use for both technical and non-technical users. The generation of a structured HTML report with visual elements, such as score breakdowns and prioritised recommendations, further enhances accessibility and supports decision-making.

The need for consistency and repeatability also influenced the design. Manual auditing approaches, while detailed, are not easily repeatable and may produce inconsistent results. By implementing a rule-based system, SecureComply ensures that the same input will always produce the same output, enabling reliable and repeatable compliance assessments. This is particularly beneficial for organisations aiming to monitor compliance over time.

Another important consideration was the balance between automation and contextual insight. While rule-based systems provide consistency, they may lack the ability to communicate results effectively to non-expert users. To address this, SecureComply adopts a hybrid approach, where AI is used as an optional enhancement to generate a CISO-style executive summary. Importantly, this AI component does not influence the core scoring logic and includes a fallback mechanism to ensure system reliability in the event of API failures or quota limitations.

Modularity and extensibility were also considered in the system design. The project was structured into distinct components, including ingestion, validation, scoring, and reporting. This modular approach allows for easier maintenance and future expansion, such as integrating additional regulatory frameworks or real-world telemetry data.

Finally, real-world applicability was a central consideration throughout the design process. The system was developed to reflect realistic SME scenarios, including limited data availability,

varying levels of security maturity, and the need for clear, actionable outputs. The inclusion of features such as benchmarking and prioritised remediation planning further enhances the practical value of the tool.

In summary, the design of SecureComply was shaped by the need to address the limitations of existing solutions while providing a practical, explainable, and accessible compliance assessment tool. By integrating deterministic scoring, data validation, usability-focused reporting, and optional AI support, the system achieves a balanced approach that aligns with both technical and user-centric requirements.

2.2.1 Explainability vs Automation

A key design consideration in the development of SecureComply was the balance between automation and explainability. While automated systems enable fast and scalable compliance assessments, they can often reduce transparency if the underlying logic is not clearly visible to the user. In the context of GDPR, where accountability and justification are critical, this lack of transparency can limit trust in the results.

Fully automated approaches, particularly those driven by AI, can generate high-level outputs quickly but may not clearly demonstrate how conclusions were reached. This creates challenges for organisations that need to understand why they are non-compliant and what specific actions are required to improve. For SMEs, this is especially important, as decision-makers may rely directly on the tool's output without access to external expertise.

To address this, SecureComply adopts a deterministic rule-based approach for all compliance scoring. Each GDPR-related control is evaluated using predefined rules, with an associated score, justification, and maximum value. This ensures that every result is fully traceable, allowing users to see exactly how their inputs influence the final compliance score. The inclusion of control-level outputs and prioritised recommendations further strengthens this explainability.

At the same time, automation remains essential for usability and scalability. SecureComply automates the entire audit pipeline, including data ingestion, validation, scoring, and report generation. This enables users to perform repeatable assessments quickly without requiring manual intervention.

To balance these two aspects, a hybrid design was implemented. Automation is used for all core processing, while explainability is preserved through transparent scoring logic and structured outputs. AI is incorporated only as an optional enhancement, generating a high-level executive summary without influencing the underlying results. This ensures that the system benefits from improved readability while maintaining full control over how scores are calculated.

This design directly addresses the limitations identified in existing solutions, where either manual approaches lack scalability or automated tools lack transparency. By combining automation with explainable scoring, SecureComply provides a system that is both efficient and trustworthy, aligning with the needs of SMEs and the accountability requirements of GDPR.

2.2.2 Data Quality Challenges

Data quality was a significant consideration in the development of SecureComply, as the accuracy and completeness of input data directly impact the reliability of compliance assessments. In real-world scenarios, particularly within SMEs, organisations often lack complete visibility over their data processing activities, leading to incomplete or inconsistent inputs.

During the development of this project, attempts were made to obtain real-world datasets to support more realistic evaluation. Access to institutional data within the college environment was explored however, this was not permitted due to data protection and confidentiality constraints, which is expected given the sensitive nature of personal and organisational data under GDPR. Similarly, efforts to source real SME GDPR-related datasets were unsuccessful, largely due to the absence of publicly available datasets containing detailed compliance information. This reflects a broader industry challenge, where GDPR-related data is inherently restricted and not openly shared.

As a result, the project adopted the use of synthetic SME datasets, generated to simulate realistic organisational scenarios. This approach allowed for controlled testing of different compliance levels while ensuring no sensitive data was exposed. However, the use of synthetic data introduced its own limitations, particularly in ensuring that generated data accurately reflects real-world complexity.

To enhance realism and reduce reliance on purely synthetic inputs, SecureComply incorporates additional host-based telemetry data. A host scanning component collects technical security indicators, such as HTTPS configuration and encryption status, which are then merged with the SME dataset. This allows certain fields to be automatically populated from real system data, reducing manual input requirements and improving data accuracy for specific controls. The integration of host-derived data demonstrates how the system can combine structured user input with externally collected signals to produce a more informed assessment.

To address both real-world and synthetic data challenges, SecureComply was designed with mechanisms to handle imperfect and incomplete inputs. During ingestion, missing values are normalised using a consistent placeholder, allowing the system to continue processing without failure. A validation layer ensures that all inputs conform to expected formats and allowed values, reducing the risk of invalid data influencing results.

A key enhancement introduced in the final version of the system is the data quality scoring mechanism, which evaluates the completeness of input data based on the proportion of missing or “empty” fields. This provides users with a clear indication of how reliable the audit results are, addressing a critical limitation in many existing tools that assume fully accurate inputs. Where data quality is low, the system explicitly warns that results may be less reliable, encouraging users to improve data completeness before making decisions.

This focus on data quality directly reflects the practical challenges identified during development and ensures that SecureComply remains robust and usable even when operating under real-world constraints. By combining synthetic data, host-derived inputs, and validation mechanisms, the system provides a more realistic and trustworthy approach to GDPR compliance assessment.

2.2.3 SME Usability Requirements

Usability was a key design consideration in the development of SecureComply, particularly given the target users of the system are Small and Medium Enterprises (SMEs), which may lack dedicated cybersecurity or legal expertise. Many existing GDPR compliance tools are designed for enterprise environments and can be overly complex, requiring significant configuration and specialist knowledge. As identified in earlier sections, this creates a barrier for SMEs seeking accessible compliance solutions.

To address this, SecureComply was designed with a focus on simplicity, clarity, and ease of use. The system provides a command-line interface (CLI) with clear commands such as --help, --usage, and --demo, allowing users to quickly understand how to operate the tool without requiring extensive technical knowledge. This approach ensures that the tool remains lightweight and easy to deploy, while still offering sufficient functionality for meaningful compliance assessment.

Another important usability consideration was the interpretability of outputs. Rather than presenting raw technical data or abstract scores, SecureComply generates a structured HTML report that includes visual elements such as score breakdowns, risk bands, and prioritised recommendations. This allows users to quickly understand their compliance posture and identify areas requiring improvement without needing to interpret complex technical details.

The system also incorporates explainable scoring outputs, where each control includes a justification and associated rule. This enables users to understand not only their score, but also why that score was assigned, supporting informed decision-making. This is particularly important for SMEs, where users may rely directly on the tool's output without external consultation.

In addition, usability was improved by reducing the burden of manual input. The integration of host-based telemetry data allows certain technical controls to be automatically populated, while the ingestion process handles missing values in a consistent manner. This reduces the likelihood of user error and simplifies the data input process.

The inclusion of a data quality score further enhances usability by providing users with immediate feedback on the completeness of their input data. This helps users understand the reliability of the results and encourages better data input practices.

Finally, the optional AI-generated executive summary improves accessibility by translating technical results into a high-level narrative suitable for decision-makers. Importantly, this feature remains optional and does not affect the underlying scoring, ensuring that usability improvements do not compromise transparency or reliability.

This usability focus also informed the development of a companion website to support onboarding, documentation, and clearer presentation of the tool. SecureComply Tool Link Here

Overall, the design of SecureComply prioritises usability by combining a simple interface, clear outputs, and supportive features that guide users through the compliance assessment process. This ensures that the system remains accessible and practical for SMEs, aligning with the core objectives of the project.

3. Project Planning and Considerations

3.1 Assumptions

Assumption	Impact
SMEs can provide basic GDPR-related input data	Enables system to generate compliance scores
Users have minimal technical knowledge	Justifies need for simple CLI and clear reports
Host system access is available for telemetry	Enables automated data collection
AI API availability is not guaranteed	Requires fallback mechanism

3.2 Constraints

Data Availability Constraint:

- Limited access to real GDPR datasets due to privacy restrictions

Time Constraint:

- Development limited to academic project timeline

Technical Constraint:

- System designed without reliance on complex enterprise infrastructure

Resource Constraint:

- No access to commercial compliance tools or SME environments

AI Dependency Constraint:

- External API usage subject to availability and quotas

3.3 Risks

Risk Description	Likelihood	Impact	Mitigation
Inaccurate input data	Medium	High	Validation + data quality scoring
AI API failure	Medium	Medium	Fallback deterministic summary
User misunderstanding outputs	Low	High	Clear reports + explanations
System errors during processing	Low	Medium	Input validation + error handling

3.3.1 Technical Risks

Technical risks relate to the reliability and correct functioning of the SecureComply system during execution. As the system relies on multiple components, including data ingestion, validation, scoring, and report generation, failures in any stage could impact overall functionality.

One key risk is system failure during processing, which may occur due to unexpected input formats or runtime errors. This was mitigated through the implementation of structured validation and error handling mechanisms.

Another technical risk is the dependency on external services, specifically the AI API used for generating executive summaries. API unavailability, rate limits, or misconfiguration could prevent this feature from functioning as expected. To address this, SecureComply includes a fallback mechanism, where a deterministic summary is generated if the AI component fails. This ensures that the system remains fully functional regardless of external dependencies.

3.3.2 Data Risks

Data-related risks were a significant consideration throughout the project, particularly due to the challenges associated with obtaining real-world GDPR datasets.

One major risk is the use of incomplete or inaccurate input data, which can lead to misleading compliance assessments. This is especially relevant for SMEs, where full visibility of data processing activities may not be available.

Additionally, the reliance on synthetic data introduces the risk that test scenarios may not fully reflect real-world complexity. While synthetic datasets enable safe and controlled testing, they may lack the variability and unpredictability of real organisational data.

To mitigate these risks, SecureComply incorporates a validation engine to enforce correct input formats and allowed values. Furthermore, a data quality scoring mechanism was introduced to assess the completeness of input data and provide users with an indication of result reliability. The integration of host-based telemetry data also helps reduce reliance on purely synthetic inputs by automatically populating certain technical controls with real system data.

3.3.3 Usability Risks

Usability risks arise from the possibility that users may misinterpret system outputs or struggle to effectively use the tool.

One key risk is user misunderstanding of compliance results, particularly if outputs are too technical or lack sufficient explanation. This is a critical concern for SMEs, where users may not have specialised knowledge in GDPR or cybersecurity. Another risk is incorrect usage of the system, such as providing improperly formatted input data or misunderstanding command-line options.

To address these risks, SecureComply was designed with a focus on clarity and accessibility. The system provides clear CLI commands, including help and usage options, to guide users. Additionally, outputs are presented through a structured HTML report, including visual breakdowns, risk classifications, and prioritised recommendations.

Explainable scoring further reduces usability risk by allowing users to understand how each result was derived.

3.3.4 Mitigation Strategies

A range of mitigation strategies were implemented throughout the development of SecureComply to address identified risks across technical, data, and usability domains.

From a technical perspective, robust validation and error handling mechanisms ensure that the system can handle unexpected inputs without failure. The inclusion of a modular architecture also improves maintainability and reduces the impact of individual component failures.

To address data-related risks, the system incorporates input validation, data normalisation, and data quality scoring, ensuring that unreliable inputs are identified and clearly communicated to the user. The addition of host-based telemetry integration further enhances data accuracy by supplementing user-provided inputs.

Usability risks are mitigated through the use of a simple command-line interface, clear documentation, and structured reporting. The inclusion of visual outputs and prioritised recommendations ensures that results are accessible and actionable.

Finally, the implementation of a hybrid AI approach with a fallback mechanism ensures that optional features do not compromise system reliability, maintaining consistent performance under all conditions.

4. System Design and Architecture

4.1 System Overview

SecureComply is a modular, command-line based system designed to assess GDPR compliance for SMEs through a structured and explainable audit pipeline. The system processes organisational input data and transforms it into a quantitative compliance score and a detailed report, enabling users to understand and improve their data protection posture.

At a high level, the system follows a linear pipeline architecture. User-provided data, in the form of a structured JSON file, is first ingested and prepared for processing. This data then passes through a validation stage to ensure consistency and correctness before being evaluated by the compliance scoring engine. The results of this evaluation are used to generate a structured HTML report, presenting both quantitative scores and actionable insights.

In addition to standard input data, the system supports optional host telemetry integration. This allows real-world system security data to be incorporated into the audit process, providing additional context alongside user-provided information without altering the core scoring model.

An optional AI component can be integrated at the final stage to enhance the interpretability of results by generating a high-level narrative summary. This component operates independently of the core scoring logic, ensuring that the system remains deterministic and explainable.

The overall design emphasises modularity, allowing each stage of the pipeline to operate as an independent component. This approach improves maintainability, supports extensibility, and enables additional features, such as benchmarking and host telemetry integration, to be incorporated without affecting the core system functionality.

4.2 Architecture Diagram

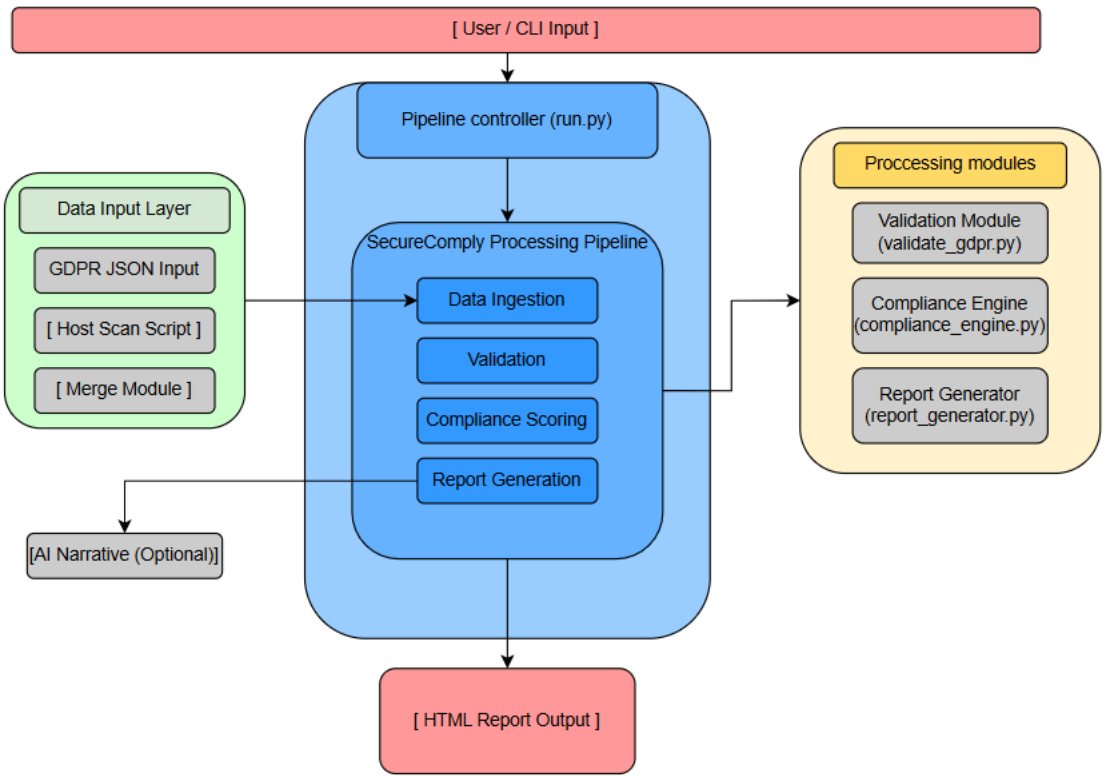

Figure (1) – SecureComply base architecture design

4.3 Core Components

SecureComply is designed as a modular pipeline composed of distinct components, each responsible for a specific stage of the GDPR audit process. This architecture enables clear separation of concerns, ensuring that data processing, validation, scoring, and reporting are handled independently while remaining tightly integrated within the overall workflow.

The system supports both structured organisational input and optional host-based telemetry, allowing it to combine declared compliance practices with observed technical indicators. Each component contributes to transforming raw input data into a structured, interpretable compliance assessment.

4.3.1 Data Ingestion Module

The data ingestion module is responsible for loading and preparing input data for processing. SecureComply accepts GDPR-related data in a structured JSON format, which is parsed and transformed into a consistent internal representation.

As part of this process, nested data structures are flattened and normalised to ensure compatibility with downstream components. Missing or incomplete values are standardised using a placeholder approach, allowing the system to handle real-world datasets without failure while maintaining consistent processing behaviour.

The ingestion stage also supports optional integration of host telemetry data. This data is generated externally through a host scanning script and merged with the primary dataset prior

to execution. The inclusion of telemetry allows the system to incorporate observable system-level indicators alongside user-provided inputs.

4.3.2 Validation Engine

The validation engine ensures that all ingested data conforms to predefined structural and logical requirements. It performs strict checks on data types, allowed values, and numerical ranges, ensuring that only valid and meaningful inputs are processed further.

In addition to field-level validation, the system enforces cross-field logic rules to maintain consistency between related inputs. For example, dependencies between privacy policy presence and clarity are validated to prevent contradictory configurations.

Invalid records are not processed in the scoring stage and are instead returned with detailed, user-friendly error messages. This improves transparency and ensures that users can identify and correct issues efficiently. The validation process also works in conjunction with the ingestion module's handling of missing data, ensuring robustness without compromising data integrity.

4.3.3 Compliance Scoring Engine

The compliance scoring engine evaluates validated inputs against a predefined set of GDPR-inspired controls. Each control is assessed using a deterministic rule-based model, assigning weighted scores based on the strength and completeness of the organisation's practices.

The scoring process is structured across key categories, including security measures, transparency obligations, and internal governance controls. Individual control results are aggregated into category-level scores and an overall compliance score, which is then mapped to a qualitative risk band.

This deterministic approach ensures that results are consistent, explainable, and reproducible. In addition, the scoring engine generates prioritised recommendations based on identified gaps, allowing users to understand which areas have the greatest impact on their overall compliance posture.

4.3.4 Report Generation Module

The report generation module converts the output of the scoring engine into a structured and user-friendly HTML report. The report presents key findings, including overall scores, category breakdowns, control-level results, and prioritised remediation recommendations.

To improve interpretability, the report includes visual elements such as score indicators and comparative breakdowns, enabling users to quickly identify areas of weakness. It also incorporates additional contextual information, such as host telemetry data, which is displayed as supplementary insight without influencing the core compliance score.

The module also supports benchmark comparison and remediation planning, providing users with a clearer understanding of their relative performance and potential improvement

pathways. The overall design ensures that technical outputs are translated into actionable insights suitable for non-specialist users.

4.3.5 AI Narrative Module

The AI narrative module provides an optional enhancement to the reporting process by generating a high-level summary of the audit results. This component analyses the structured output from the scoring engine and produces a concise, human-readable narrative outlining the organisation's compliance posture, key risks, and recommended actions.

The module operates independently of the core scoring logic, ensuring that all compliance results remain deterministic and unaffected by AI-generated content. In cases where AI functionality is unavailable, the system falls back to a predefined deterministic summary, maintaining consistent output generation.

This approach balances explainability and usability, allowing the system to provide more accessible insights while preserving transparency and reliability in the underlying assessment.

5. Implementation

5.1 Command-Line Interface Design

The SecureComply system was designed as a command-line interface (CLI) application, prioritising simplicity, portability, and ease of use for SME users with limited technical expertise. The CLI serves as the primary interaction layer, allowing users to execute the full GDPR audit pipeline with a single command, while also supporting additional usability features such as help and usage guidance.

The core execution model is based on a single entry point (run.py), which accepts an input JSON file and orchestrates the full pipeline. This design reduces complexity for the end user, enabling execution through a simple command such as:

python run.py data/input.json

This approach aligns with the project goal of minimising setup overhead while maintaining flexibility for different input datasets.

To enhance usability, the CLI includes structured feedback during execution, displaying clear progress indicators for each stage of the pipeline, including ingestion, validation, and report generation etc. This provides transparency to the user and improves trust in the system's operation.

In addition, dedicated documentation was implemented through a structured input guide, which defines the required JSON format, allowed values, and common errors. This acts as a companion to the CLI, reducing user error and improving input correctness.

The CLI design also supports modularity and extensibility, allowing optional features such as host telemetry integration and benchmarking to be executed independently of the main pipeline. For example, host data can be merged prior to execution, and benchmark generation is performed as a separate offline process.

Error handling was a key design consideration. The CLI provides clear and specific validation feedback, ensuring that users are informed of exact issues within their input data rather than receiving generic failure messages. This is achieved through structured validation logic, where errors are returned per record and displayed in a readable format.

Overall, the command-line interface was intentionally designed to balance simplicity, clarity, and robustness, ensuring that SecureComply remains accessible to non-expert users while still supporting advanced functionality required for a comprehensive GDPR compliance assessments.

5.2 Data Processing and Normalisation

The SecureComply system implements a structured data processing pipeline to transform raw SME input data into a consistent and analysable format suitable for compliance assessment. This stage is critical, as real-world organisational data is often incomplete, inconsistently formatted, or incorrectly typed.

The data processing workflow begins with the ingestion of user-provided JSON input through the ingestion module. Input data follows a nested structure, grouping fields into categories such as basic security measures, transparency and user rights, and internal controls. To support downstream processing, this structure is flattened into a single unified record, ensuring all fields can be accessed consistently during validation and scoring.

A key design feature of this stage is the handling of missing or incomplete data. The system introduces a standard placeholder value ("empty") to represent missing inputs. This is implemented through a recursive function that replaces null or blank values across all fields. This approach ensures that the pipeline does not fail when encountering incomplete datasets and allows missing values to be handled systematically in later stages .

Following this, the system performs data normalisation, converting inputs into consistent data types and formats. Boolean values are standardised from multiple possible representations (e.g., "yes", "1", "true") into true Boolean types, while numeric fields such as response times are converted into integers where applicable. Additionally, categorical (Enum) values are cleaned through trimming and lowercasing to ensure accurate matching during validation and scoring.

This normalisation process is essential for ensuring compatibility with the scoring engine, as inconsistent input formats could otherwise lead to incorrect evaluations or runtime errors. By enforcing uniform data structures prior to validation, the system improves both reliability and accuracy.

Another important aspect of the data processing stage is the integration of host-based telemetry data. When available, system-level information such as HTTPS status and encryption configuration is merged into the input dataset before processing. This enrichment step enhances data completeness and reduces reliance on manual user input, improving the realism and reliability of the assessment.

Overall, the data processing and normalisation stage was designed to ensure that all input data, regardless of quality or format, is transformed into a clean, standardised, and robust structure. This enables accurate validation, consistent scoring, and reliable report generation, while also supporting real-world scenarios where data quality may be limited.

5.3 Validation Logic

The validation logic in SecureComply ensures that all input data is accurate, consistent, and suitable for compliance scoring before further processing. This stage acts as a safeguard within the pipeline, preventing invalid or misleading data from affecting audit results.

The system is designed with robustness and usability in mind. Rather than rejecting incomplete datasets outright, missing values are standardised using a placeholder ("empty"), allowing records to proceed through the pipeline while ensuring such fields receive a score of zero during evaluation.

Validation is performed at multiple levels:

Data Type Validation:

- Fields are checked to ensure they match expected types. For example, boolean fields must be True or False, while numerical fields such as response times must be valid integers within defined ranges.

Enumerated Value Validation:

- Many inputs are restricted to predefined categories (e.g. password hashing methods, MFA enforcement levels). Any value outside the allowed set is flagged as invalid, ensuring consistency and preventing ambiguity in scoring.

Range Validation:

- Numeric fields such as DSAR response time and breach notification time are validated against realistic and regulatory-aligned thresholds to ensure meaningful input data.

Cross-Field Logical Validation:

- The system enforces logical relationships between fields to reflect real-world GDPR requirements. For example, if a privacy policy is not present, its clarity must be marked as “missing”, and if consent is used as a lawful basis, an appropriate cookie consent mechanism must be in place.

When validation errors occur, they are not silently ignored. Instead, the system provides clear, structured error messages in the command-line interface, identifying the exact field and issue. Invalid records are excluded from scoring but retained in the output for transparency and debugging.

In the final version of SecureComply, support for “N/A” values was introduced to distinguish between genuinely missing data and controls that were not applicable to a given SME context. This improved the fairness of the assessment by preventing non-applicable controls from being treated as standard missing values. Where a control was marked as not applicable, it was excluded from the effective scoring base and tracked separately in the output, improving both transparency and accuracy in the final compliance result.

Overall, this validation layer ensures that SecureComply produces reliable, consistent, and explainable compliance assessments while maintaining usability for non-technical users.

5.4 Scoring Model Implementation

The scoring model in SecureComply translates validated input data into a quantitative GDPR compliance score, providing a structured and explainable assessment of an organisation's security and privacy posture.

The model is rule-based and deterministic, ensuring transparency and reproducibility of results. Each control is evaluated individually and assigned a score based on predefined mappings that reflect cybersecurity best practices and GDPR principles.

Scoring Structure

The overall score is calculated out of 100 and is divided across three core categories:

- Basic Security Measures (30 points)
- Transparency & User Rights (40 points)
- Internal Controls (30 points)

Each category contains multiple controls, with specific weightings assigned based on their importance to GDPR compliance.

Control-Level Scoring

Each input field is mapped to a score using predefined rules. For example:

- Strong password hashing algorithms (e.g. Argon2, bcrypt) receive higher scores than weaker methods such as MD5 or plaintext
- Continuous or frequent security testing scores higher than ad-hoc or no testing
- Full encryption and organisation-wide MFA enforcement receive maximum points

These mappings are implemented using lookup-based scoring functions, ensuring consistency across all records.

For numerical fields, threshold-based scoring is applied. For instance:

- DSAR response times of ≤30 days receive full marks, delays beyond this threshold result in reduced scores

This approach aligns scoring with regulatory expectations while maintaining simplicity.

Dependency Handling

The scoring model also incorporates logical dependencies between controls. For example:

- If no privacy policy is present, the clarity score is automatically set to zero

This ensures that related controls are evaluated in context rather than independently, improving the realism of the assessment.

Score Aggregation and Risk Classification

Individual control scores are aggregated to produce:

Category-level scores

- Overall compliance score (0–100)

The final score is then mapped to a risk band:

- 85–100: Strong
- 70–84: Moderate
- 50–69: Weak
- 0–49: High Risk

This classification provides a clear, high-level interpretation of organisational risk.

Recommendation Generation

For each control where points are lost, the system generates a prioritised recommendation. These are ranked based on the number of points lost, ensuring that the most critical gaps are addressed first.

This transforms the scoring model from a purely evaluative system into a practical decision-support tool, guiding organisations on how to improve their compliance posture.

Overall, the scoring model provides a transparent, structured, and explainable mechanism for assessing GDPR compliance, balancing simplicity with meaningful security insights.

5.5 Report Generation

The report generation component of SecureComply is responsible for transforming the computed audit results into a structured, user-friendly output that clearly communicates an organisation's GDPR compliance posture.

The system generates a dynamic HTML report, designed to be both visually clear and accessible to non-technical users such as SMEs, managers, or auditors.

Report Structure

The generated report presents key information in a logical and easy-to-understand format, including:

- **Overall Compliance Score and Risk Band**
  A high-level summary of the organisation's GDPR posture, including a numerical score (0–100) and corresponding risk classification.

- **Category-Level Breakdown**
  Scores are grouped into the three main areas:
  - Basic Security Measures
  - Transparency & User Rights
  - Internal Controls

This allows users to quickly identify which areas require the most attention.

- **Control-Level Results**
  Each individual control is displayed with:
  - Input value
  - Score achieved vs maximum
  - Justification explaining the result

This supports transparency and explainability of the scoring model.

- **Prioritised Recommendations**
  The report highlights the most critical gaps, ranked by impact, and provides actionable recommendations for improvement.

AI-Generated Narrative (Optional)

An optional feature of the system is the inclusion of an AI-generated CISO-style risk summary. This narrative:

- Summarises the organisation's overall risk posture
- Highlights key weaknesses
- Explains potential business impacts
- Suggests priority actions

The narrative is generated using an external API, with a fallback mechanism in place if the service is unavailable. To ensure transparency, the report explicitly states that this section is AI-generated and should be reviewed by a human expert.

Output and Accessibility

The final report is saved as an HTML file within the /reports directory and is automatically opened in the user's default web browser upon completion of the audit.

This approach provides several advantages:

- No additional software required to view results
- Easy sharing and distribution
- Clean visual presentation compared to raw JSON output

Design Considerations

The report was designed with usability as a priority:

- Clear visual hierarchy for quick interpretation
- Minimal technical jargon to support non-expert users
- Structured layout to reduce cognitive load
- Integration of both quantitative scores and qualitative insights

Overall, the report generation component ensures that complex compliance data is translated into meaningful, actionable insights, making SecureComply practical and accessible for real-world use.

5.6 Benchmarking System

The benchmarking system in SecureComply was designed to provide contextual meaning to individual audit results by comparing them against a broader dataset of SME security and compliance profiles. While a standalone score (e.g. 68/100) indicates performance, it does not clearly communicate how that organisation compares to others. The benchmarking layer addresses this by positioning results within a relative context.

The system operates by generating and utilising a synthetic dataset of SME GDPR profiles, created using the project's data generation module. This dataset models a wide range of realistic organisational behaviours, from low-maturity environments with minimal controls to more mature implementations with stronger security and governance practices.

Once generated, these benchmark records are processed through the same pipeline as user input data using the identical scoring model. By doing this, consistency is maintained across all evaluations, ensuring that comparisons are fair and methodologically sound. The benchmark results effectively form a reference distribution of scores across SMEs.

The user's audit result is then compared against this distribution. This provides a practical comparative perspective through score positioning and risk band classification. This allows users to understand whether their organisation falls into a stronger or weaker category relative to typical SME profiles.

The decision to use synthetic benchmarking rather than real organisational data was driven by several factors. Firstly, access to real GDPR compliance datasets is limited due to privacy and confidentiality concerns. Secondly, synthetic data allows full control over input variability,

ensuring coverage across all scoring scenarios. Finally, it enables reproducibility, which is important for both testing and academic evaluation.

From a design perspective, the benchmarking system enhances the interpretability of results without introducing unnecessary complexity. This helps users understand not only their compliance level but also how it compares within a broader SME context.

5.7 Host Telemetry Integration (Optional)

An optional feature of SecureComply is the integration of host-level telemetry, designed to enhance the accuracy and realism of compliance assessments by incorporating actual system security data alongside user-provided inputs.

This functionality is implemented through a lightweight Bash script (host_scan.sh), which performs basic security checks on the host system. The script collects information such as HTTPS availability, disk encryption status, firewall configuration, automatic updates, system logging, and open ports. These checks provide a snapshot of the system's security posture at the time of execution.

The output of the script is stored in a structured JSON file, which is then merged with the SME GDPR dataset using a dedicated merging module.

A key design decision was to ensure that host telemetry does not directly influence the compliance score. Instead, it is presented in the final report as contextual information. This prevents environmental factors or system-specific configurations from unfairly impacting the standardised scoring model, while still providing valuable insight to the user.

The inclusion of this feature was motivated by the limitation of self-reported data, which may be inaccurate or overly optimistic. By incorporating automated host checks, the system introduces an element of objective verification, bridging the gap between declared policies and actual technical implementation.

From a usability perspective, the feature is optional and can be executed independently via command-line. This ensures that the tool remains lightweight and accessible, while still offering extended functionality for more advanced use cases.

Overall, the host telemetry integration enhances the practical value of SecureComply by combining policy-level assessment with real-world system data, without compromising the consistency or fairness of the scoring model.

6. System Evolution and Iterative Development

6.1 Overview of Development Approach

The development of SecureComply followed an iterative and incremental approach, with each version building upon the limitations and findings of the previous iteration. Rather than attempting to design a complete solution from the outset, the system evolved through multiple stages, allowing for continuous refinement in terms of functionality, usability, and real-world applicability.

Each iteration focused on addressing specific challenges identified during development, including data handling, validation, explainability, and user experience. This approach enabled the system to gradually transition from a basic prototype to a more robust and feature-complete compliance auditing tool.

The iterative process also allowed for the incorporation of feedback and practical insights, particularly in relation to SME requirements, data quality limitations, and the need for transparent and explainable outputs. As a result, SecureComply reflects a progression from a simple scoring concept to a comprehensive and user-focused compliance assessment system.

6.2 Version 1 – Initial Prototype

Version 1 of SecureComply focused on developing a basic proof-of-concept compliance scoring system. At this stage, the system implemented a simple rule-based model to evaluate a limited number of GDPR-related controls and produce an overall compliance score.

The primary objective of this version was to validate the feasibility of automating GDPR assessments using a deterministic approach. While functional, Version 1 lacked structured input handling, validation mechanisms, and user-friendly output formats.

As a result, the system was limited in usability and could not reliably handle real-world data inputs, highlighting the need for improved data structuring and validation in subsequent versions.

6.3 Version 2 – Input Structuring and Validation

Version 2 introduced significant improvements in data handling and input validation. A structured JSON format was defined to standardise input data, ensuring consistency across assessments.

A validation layer was implemented to enforce allowed values and data types, reducing the risk of invalid inputs affecting the scoring process. This marked an important step towards improving the reliability of the system.

Version 2 represented a significant refinement of the original input model used in the early prototype. The initial version relied on a smaller set of basic indicators across security measures, transparency, and internal controls, with several fields represented as simple Boolean values. In Version 2, this structure was expanded into a more detailed and realistic audit schema through the introduction of maturity-based enumerated fields, additional governance and transparency indicators, and more explicit operational controls. Examples of these additions include encryption at rest, MFA enforcement, patch management frequency, lawful basis identification, third-party sharing disclosure, breach notification timing, records of

processing, and DPIA process maturity. This improved the realism and granularity of the assessment while preserving the same overall 100-point category weighting, ensuring continuity between versions.

However, while input quality improved, the system still lacked robustness in handling missing data and did not yet provide meaningful output beyond basic scoring.

6.4 Version 3 – Improved Data Handling

Version 3 focused on improving the system's ability to handle incomplete and imperfect data, which is a common issue in SME environments. A data ingestion and normalisation process was introduced, allowing missing values to be handled consistently using predefined placeholders.

This enabled the system to continue processing even when data was incomplete, improving robustness and real-world applicability. Despite these improvements, the system still lacked user-friendly outputs and remained primarily technical in nature.

6.5 Version 4 – Report Generation Introduction

Version 4 marked a significant shift in the project by introducing structured report generation. Instead of outputting raw scores, the system began producing more meaningful results, including category breakdowns and initial recommendations.

This improved the interpretability of outputs and began transitioning the system from a technical prototype to a more user-oriented tool. However, the presentation of results was still limited and lacked visual clarity.

6.6 Version 5 – Usability and Visual Enhancements

Version 5 focused on enhancing usability and presentation, particularly for SME users. A structured HTML/CSS report was introduced, incorporating visual elements such as score breakdowns, risk classifications, and prioritised recommendations.

Additional improvements included the introduction of CLI usability features, such as help and usage commands, making the tool more accessible to non-expert users.

At this stage, the system had evolved into a functional compliance tool however, further improvements were required to enhance realism and interpretability.

6.7 Version 6 – Final System Enhancements

Version 6 represents the final iteration of SecureComply, incorporating several key enhancements to improve functionality, realism, and usability.

To further improve accessibility and presentation, a companion website was developed for SecureComply. The site provides a central location for tool overview, setup guidance, usage instructions, example input, CLI output examples, and supporting information such as terms and conditions. This was intended to reduce the usability barrier for new users by complementing the command-line interface with a more approachable front-end information layer. While the website does not replace the core downloadable tool, it enhances usability by making documentation, onboarding, and project presentation more accessible.

SecureComply Tool Link Here

6.7.1 AI Integration

An optional AI component was introduced to generate a CISO-style executive summary, improving the readability of results for non-technical users. Importantly, this feature does not influence the underlying scoring logic and includes a fallback mechanism to ensure reliability

6.7.2 Benchmarking Feature

A benchmarking system was implemented to allow results to be compared against synthetic SME baseline data, providing additional context for interpreting compliance scores.

6.7.3 Remediation Task Planning

The system was enhanced to include prioritised remediation recommendations, allowing users to identify and address the most critical compliance gaps efficiently.

6.7.4 Data Quality Scoring

A data quality scoring mechanism was introduced to assess the completeness of input data. This feature highlights when results may be unreliable due to missing inputs, directly addressing a key limitation identified during development.

6.8 Summary of Improvements

The development of SecureComply demonstrates a clear and structured progression from an initial proof-of-concept to a fully functional, explainable, and user-focused GDPR compliance auditing tool. Each version introduced targeted improvements that addressed specific limitations identified during earlier stages of development.

In the early stages, the system evolved from a basic rule-based prototype (Version 1) into a more structured solution through the introduction of standardised input formats and validation mechanisms (Version 2). This significantly improved the reliability of the system by ensuring that only valid and consistent data could be processed.

As development progressed, Version 3 addressed a critical real-world challenge: handling incomplete and imperfect data. The introduction of data normalisation and default handling mechanisms enabled the system to operate effectively even when input data was missing, reflecting realistic SME scenarios.

Version 4 marked a transition from a technical prototype to a more practical tool through the introduction of structured report generation. This significantly improved the interpretability of outputs, allowing users to move beyond raw scores and begin understanding their compliance posture.

In Version 5, the focus shifted towards usability and user experience. The implementation of a structured HTML report with visual elements, along with improved CLI functionality, made the system more accessible to non-expert users. This version emphasised clarity, ensuring that outputs were not only accurate but also easy to understand and act upon.

The final iteration, Version 6, introduced several advanced features that significantly enhanced the overall capability of the system. These included the integration of an optional AI-generated executive summary, providing high-level insights without compromising the deterministic nature of the scoring model. A benchmarking system was also introduced, allowing users to compare their compliance posture against representative SME data.

A key contribution of the final version is the introduction of a data quality scoring mechanism, which evaluates the completeness of input data and highlights potential reliability issues in the results. This directly addresses a major limitation identified in both research and earlier system versions, where compliance assessments often assume fully accurate input data.

In addition, the system incorporated host-based telemetry integration, enabling the automatic collection of technical security indicators and reducing reliance on manual input. This enhancement improves both the accuracy and realism of the assessment process by combining user-provided data with externally derived system information.

Across all versions, there was a consistent emphasis on explainability, consistency, and real-world applicability. The deterministic rule-based scoring model remained central throughout

development, ensuring that all outputs are transparent and traceable. At the same time, usability improvements and AI-assisted features were introduced in a controlled manner to enhance accessibility without compromising trust.

Overall, the iterative development process allowed SecureComply to evolve into a well-rounded system that addresses key challenges faced by SMEs, including limited resources, incomplete data, and the need for clear, actionable insights. The final system reflects a balance between technical robustness and user-focused design, demonstrating continuous improvement and informed decision-making throughout the project lifecycle.

7. Evaluation and Testing

The testing strategy for SecureComply was designed to ensure that the system operates reliably, produces accurate results, and remains robust when handling real-world data conditions. Given the modular pipeline architecture of the system, testing was conducted across each stage of execution, including data ingestion, validation, compliance scoring, and report generation.

A combination of functional and data-focused testing approaches was used. Functional testing was performed to verify that the full audit pipeline executes correctly from input to output, ensuring that valid datasets result in accurate compliance scores and correctly generated reports. In parallel, validation testing was used to assess how effectively the system detects and handles invalid or inconsistent input data. This included testing incorrect data types, unsupported values, and logical inconsistencies between related fields.

Particular emphasis was placed on testing the system's handling of incomplete data, as this reflects realistic SME scenarios. The ingestion process was tested with partially filled datasets to ensure that missing values are handled consistently without causing system failure. This allowed the robustness of the data normalisation and validation layers to be evaluated under non-ideal conditions.

To further ensure reliability and real-world applicability, the system was tested across multiple environments. SecureComply was executed on the primary development machine, as well as within an Ubuntu WSL environment and a separate virtual machine. In each case, the project was cloned from its GitHub repository and run without modification, confirming that the system is portable and not dependent on a specific configuration or environment.

Test datasets included a range of synthetic SME profiles representing varying levels of compliance, from low-maturity configurations with weak controls to more advanced scenarios with stronger security and governance practices. This enabled verification that the scoring model behaves consistently across different input conditions and produces appropriate score distributions and risk classifications.

Overall, the testing strategy ensured that SecureComply is functionally correct, resilient to invalid and incomplete data, and capable of operating consistently across different environments. This provides confidence in the reliability and usability of the system in practical use cases.

7.1 Testing Strategy

The testing strategy for SecureComply was designed to ensure that the system operates reliably, produces accurate compliance assessments, and remains robust when handling realistic SME data conditions. Due to the modular pipeline architecture of the system, testing was conducted across each stage of execution, including data ingestion, validation, compliance scoring, and report generation. This approach ensured that each component functioned correctly both independently and as part of the overall system.

A combination of functional and data-focused testing approaches was applied. Functional testing verified that the complete audit pipeline executes successfully from input to output, ensuring that valid datasets produce correct compliance scores and structured reports. In parallel, validation testing assessed the system's ability to detect and handle invalid inputs, including incorrect data types, unsupported enumeration values, and logical inconsistencies between related fields.

Particular emphasis was placed on handling incomplete or imperfect data, reflecting real-world SME scenarios. Test datasets were intentionally designed with missing or partial values to evaluate how the system responds under non-ideal conditions. The ingestion and validation modules were tested to ensure that missing data does not cause system failure, and that such cases are handled consistently within the scoring process.

To further evaluate reliability and portability, SecureComply was tested across multiple environments. The system was executed on the primary development machine, within an Ubuntu WSL environment, and on a separate virtual machine. In each case, the project was cloned from its GitHub repository and executed without modification, confirming that the system is environment-independent and can be deployed consistently without configuration issues.

In addition, multiple synthetic datasets representing varying levels of GDPR compliance were used. These ranged from low-compliance scenarios with weak controls to higher-maturity configurations. This ensured that the scoring engine was tested across a broad range of inputs and produced consistent and meaningful output distributions.

Testing Type	Purpose	Outcome
Functional Testing	Verify full pipeline execution	Successful report generation and correct scoring
Validation Testing	Detect invalid inputs and logical inconsistencies	Errors correctly flagged and invalid records excluded
Data Robustness Testing	Assess handling of missing or incomplete data	No system crashes and consistent fallback handling applied
Cross-Environment Testing	Ensure system runs across different platforms	Consistent execution across all environments tested
Scenario-based Testing	Evaluate scoring across different SME compliance levels	Accurate score validation and correct risk classification

7.2 Functional Testing

Functional testing was conducted to verify that SecureComply performs all core operations correctly across the full audit pipeline. The objective of this testing was to ensure that each stage of the system from data ingestion through to final report generation operates as intended and produces consistent, accurate results when provided with valid input data.

Testing began with the ingestion of structured JSON datasets representing SME GDPR profiles. These datasets were processed through the ingestion module to confirm that input data is correctly loaded and normalised into a format suitable for downstream processing. Particular attention was given to ensuring that all expected fields were correctly recognised and passed into the validation stage without loss or corruption of data.

Following ingestion, the validation component was tested to ensure that correctly formatted datasets pass without error and proceed through the pipeline. Valid datasets were confirmed to produce no validation warnings, allowing the system to move directly to the scoring phase. This verified that the validation logic does not incorrectly flag compliant input data.

The compliance scoring engine was then tested using multiple datasets with varying levels of GDPR maturity. These tests confirmed that the scoring model produces appropriate results based on the input values provided. For example, datasets containing strong security controls, clear privacy policies, and well-defined governance processes resulted in higher compliance scores, while weaker configurations produced lower scores and higher risk classifications. This demonstrated that the scoring logic behaves consistently and reflects expected compliance outcomes.

The final stage of functional testing focused on report generation. The system was tested to ensure that, upon completion of scoring, a structured HTML report is generated successfully. These reports were verified to include key elements such as the overall compliance score, category-level breakdowns, and targeted recommendations. The structure and readability of the report output were also reviewed to ensure that results are presented clearly for non-technical users.

Functional testing was repeated across multiple datasets to confirm consistency of results. Identical inputs were found to produce identical outputs, confirming the deterministic nature of the system. Additionally, the full pipeline was executed across different environments, including a local machine, Ubuntu WSL, and a virtual machine, with consistent behaviour observed in each case.

Overall, the results of functional testing confirm that SecureComply performs reliably across all core components, producing accurate compliance assessments and structured outputs in a consistent and repeatable manner.

Component Tested	Test Description	Result
Data Ingestion	Load and normalise structured JSON input	Data processed correctly
Validation Engine	Verify valid datasets pass without errors	No false validation errors
Scoring Engine	Evaluate datasets with varying compliance levels	Accurate score generation

Report Generation	Generate HTML report with results and recommendations	Report produced successfully
Pipeline Consistency	Repeat tests with identical inputs	Identical outputs confirmed
Cross-Environment Test	Execute SecureComply on Local Machine, WSL and VM	Consistent behaviour across systems

7.3 Validation Testing

Validation testing was carried out to assess the system's ability to correctly identify, report, and handle invalid or inconsistent input data. As SecureComply relies on structured input to generate accurate compliance assessments, ensuring the integrity of incoming data is a critical requirement. This stage of testing focused on verifying that the validation engine enforces expected data formats, detects errors accurately, and prevents invalid data from influencing the final compliance score.

A range of invalid input scenarios were intentionally introduced to evaluate the effectiveness of the validation logic. These included incorrect data types, unsupported enumeration values, and logical inconsistencies between related fields. For example, fields such as password_storage_method were tested with values outside of their allowed sets, while numerical fields such as dsar_response_time_days were tested with non-numeric inputs. In each case, the system successfully identified the invalid entries and flagged them during the validation stage.

In addition to type and value validation, logical consistency checks were also tested. Scenarios were created where dependent fields conflicted with one another, such as specifying a clear privacy policy when no policy was present. The validation engine was able to detect such inconsistencies, ensuring that the dataset reflects a realistic and logically sound compliance profile before scoring is applied.

When validation errors were detected, the system was observed to halt progression to the scoring phase for the affected records and provide clear feedback to the user via the command-line interface. Error messages were structured to identify the specific record and field where the issue occurred, allowing users to correct input data efficiently. This behaviour was verified across multiple test cases to ensure consistency and clarity of feedback.

Validation testing also confirmed that valid datasets pass through the validation stage without unnecessary warnings or interruptions. This ensures that the validation logic is strict enough to catch errors, while not being overly restrictive on correctly formatted data.

Overall, the validation testing process demonstrated that SecureComply is capable of enforcing data integrity, detecting a wide range of input errors, and providing clear, actionable feedback. This contributes significantly to the reliability of the system, ensuring that compliance scores are generated only from valid and meaningful input data.

Test Case	Description	Result

Invalid Enum Values	Unsupported values in enumerated fields	Correctly detected and flagged
Incorrect Data Types	Non-numeric or malformed inputs in numeric fields	Validation error triggered
Logical Inconsistencies	Conflicted field relationships	Detected and reported
Error reporting	CLI feedback for invalid records	Clear, field-specific error messages shown
Valid Data Handling	Properly formatted datasets	Passed without errors
Scoring Prevention	Invalid data proceeding to scoring	Prevented successfully

7.4 Handling of Invalid and Missing Data

Handling invalid and incomplete data was a key consideration in the design of SecureComply, as real-world SME datasets are often inconsistent, partially complete, or incorrectly formatted. Rather than assuming ideal input conditions, the system was developed to manage these scenarios in a controlled and predictable manner without compromising stability or reliability.

For invalid data, the system relies on the validation engine to detect and isolate issues before any scoring is performed. When an invalid value is encountered—such as an unsupported enumeration or incorrect data type—the affected record is flagged and excluded from further processing. This prevents inaccurate or misleading compliance scores from being generated. Clear error messages are provided through the command-line interface, identifying the exact field and issue, allowing users to correct the input efficiently.

In contrast, missing data is handled differently to reflect more realistic usage conditions. Instead of rejecting incomplete datasets entirely, the ingestion module normalises missing or empty fields using a consistent placeholder value. This ensures that the system can continue processing without failure, even when certain inputs are not provided. During the scoring phase, these missing values are treated conservatively, typically contributing a score of zero for the affected control.

This approach ensures that incomplete data does not artificially inflate compliance scores, while still allowing the system to generate a meaningful overall assessment. It also reflects a practical assumption that the absence of evidence for a control (e.g., no documented policy or process) should be treated as a potential compliance weakness.

The handling of missing data also contributes to usability. SMEs may not always have full visibility over all compliance areas, particularly in early-stage assessments. By allowing partial datasets to be processed, SecureComply enables users to gain insight into their current position without requiring perfect input data. This makes the tool more accessible and better aligned with real-world organisational conditions.

Overall, the system distinguishes clearly between invalid and missing data: invalid inputs are rejected to preserve accuracy, while missing data is handled gracefully to maintain usability. This balance ensures both the integrity of the scoring process and the practical applicability of the tool.

7.5 Example Outputs

To demonstrate the practical operation of SecureComply, a series of example outputs were captured during testing. These outputs illustrate the system's behaviour across key stages of execution, including pipeline processing, validation feedback, and final report generation. Together, they provide evidence of the system's functionality, reliability, and usability.

The first example shows the execution of the full audit pipeline through the command-line interface. When a valid dataset is provided, the system progresses through each stage in sequence, including data ingestion, validation, scoring, and report generation. This confirms that the pipeline operates end-to-end without interruption and provides clear feedback to the user during execution.

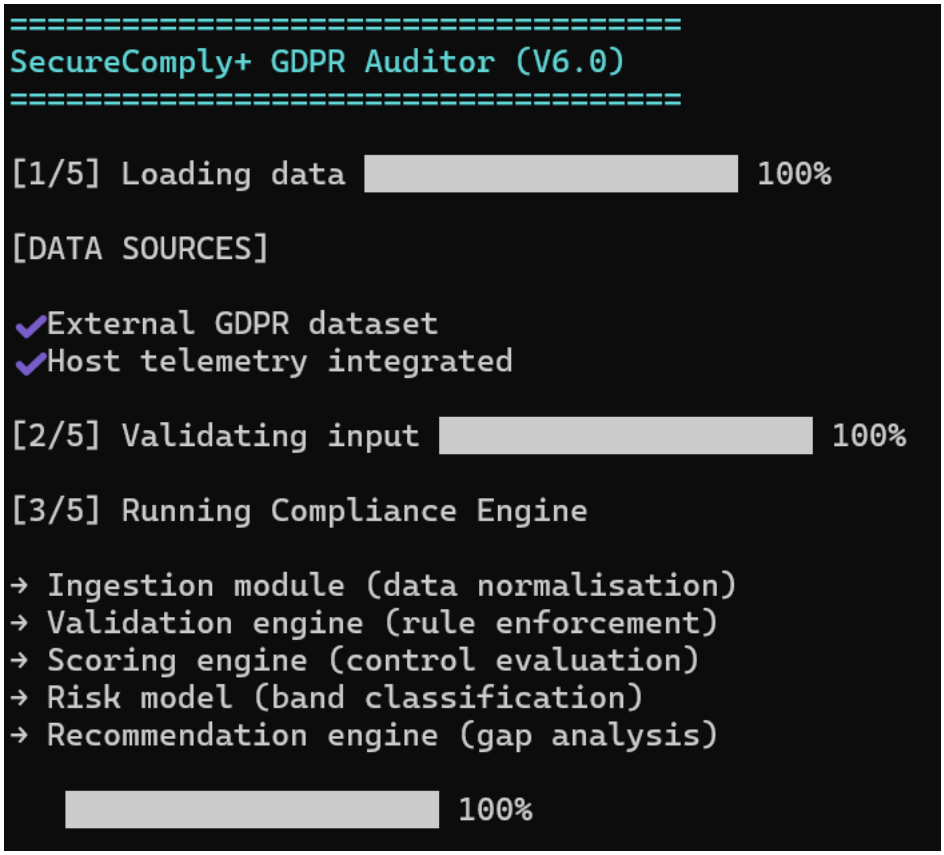

Figure 8.1: CLI Execution of SecureComply Audit Pipeline

This output demonstrates that the system processes input data in a structured manner, with each stage clearly indicated. The progress-style feedback improves usability by allowing users to track execution in real time.

The second example highlights the system's validation capabilities when incorrect input data is supplied. system successfully detected the issues inputted and returned a clear, structured error message identifying the specific field and record affected.

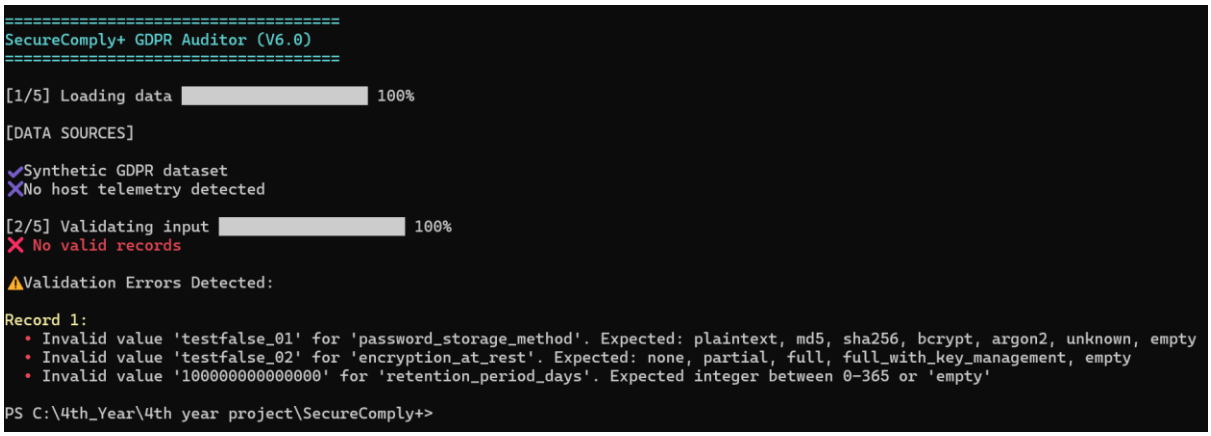

Figure 8.2: Validation Error Output for Invalid Input Data

This behaviour confirms that invalid data is correctly identified and prevented from progressing further into the scoring stage. The clarity of the error message ensures that users can quickly identify and correct issues, improving overall usability.

The third example presents the final compliance output generated from a valid dataset. Upon successful processing, the system produces an overall compliance score along with a corresponding risk classification. This output reflects the deterministic nature of the scoring model, where identical inputs consistently produce the same results.

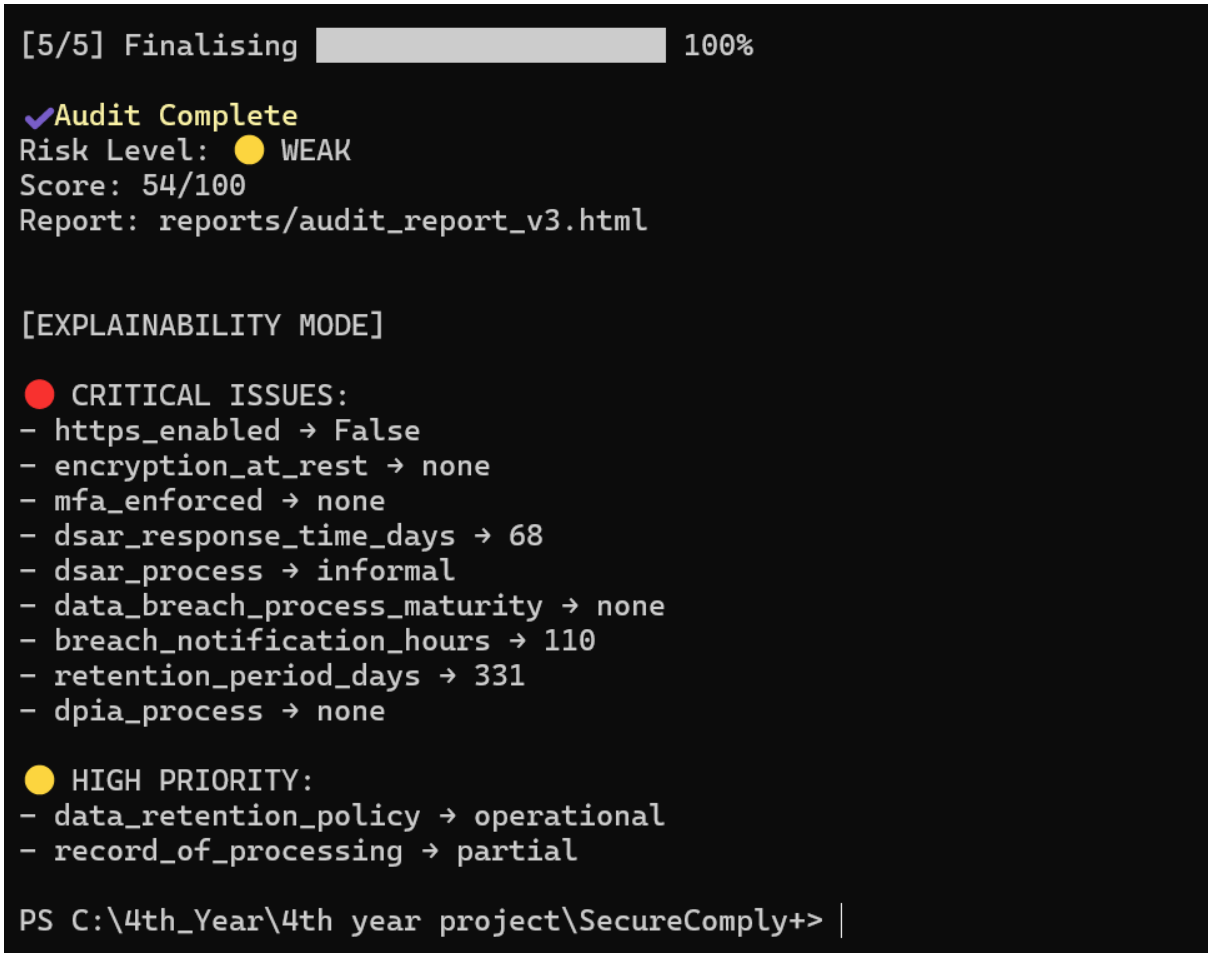

Figure 8.3: Compliance Score and Risk Classification Output, with use of --explain

Finally, the system generates a structured HTML report containing a detailed breakdown of results. This includes the overall score, category-level scoring, and targeted recommendations for improvement. The report is designed to be easily interpreted by non-technical users, translating technical compliance data into clear and actionable insights.


GDPR Audit Report (v6.0)
Overall Posture
54/100
54
Good
Category Score Chart
Benchmark Comparison
Your Score: 54
SME Average: 49.21
Median: 49
Top Quartile (Q3): 57
Top Gaps (Points Lost)
Recommendations
CISO Risk Statement
Based on this audit, the organisation's GDPR control posture is rated 'Weak' with an overall score of 54/100 (High risk). The most material risk drivers are: https_enabled (basic_security_measures), data_breach_process_maturity (internal_controls), breach_notification_hours (internal_controls). If unaddressed, these gaps increase the likelihood of non-compliance outcomes (e.g., insufficient transparency, weak security governance, or delayed rights handling), potentially resulting in regulatory scrutiny, reputational impact, and operational disruption. Immediate remediation should prioritise: Enforce HTTPS across all services (HSTS recommended). Next: Create and test an incident response plan with roles and notification steps. Evidence capture and re-audit are recommended to demonstrate measurable improvement and auditability.
Operational Security Indicators

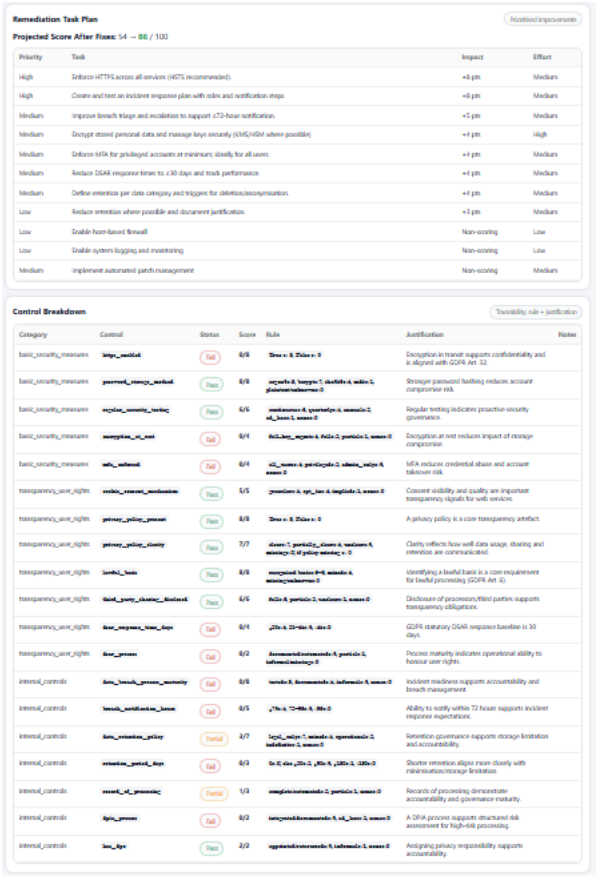

Figure 8.4: Generated HTML Compliance Report

The outputs presented demonstrate that SecureComply operates correctly across all stages of the audit pipeline. Each stage produces clear and interpretable results, from command-line feedback during execution to structured validation messages and final report generation. This confirms that the system is both functionally reliable and capable of presenting compliance results in a manner that is accessible to non-technical users.

8. Results and Discussion

8.1 Analysis of Audit Results

The results generated by SecureComply demonstrate the system's ability to produce consistent and meaningful GDPR compliance assessments across a range of input scenarios. Using synthetic SME datasets with varying levels of security, transparency, and governance maturity, the system produced a distribution of compliance scores that reflected the quality of the input data.

Across multiple test runs, compliance scores were observed to fall within all defined scoring bands, including high-risk, weak, moderate, and strong classifications. Lower scores were typically associated with datasets lacking key controls such as encryption, clear privacy policies, or defined breach response procedures. In contrast, higher scores were produced when datasets included stronger security practices, well-documented processes, and clear

transparency measures. This variation confirms that the scoring engine responds appropriately to differences in organisational maturity.

The results also demonstrated consistency in output. Identical datasets produced identical scores across repeated executions, confirming the deterministic nature of the scoring model. This ensures that results are reproducible and suitable for audit-style use, where consistency and traceability are essential.

In addition to overall scores, the system generated category-level breakdowns, allowing results to be analysed across key GDPR areas such as security measures, transparency and user rights, and internal controls. This provided a more detailed view of organisational strengths and weaknesses, rather than relying solely on a single aggregate score.

The generated outputs were presented through structured HTML reports, which included compliance scores, risk classifications, and targeted recommendations. These reports translated technical scoring outcomes into clear and interpretable insights, supporting usability for non-technical users. The inclusion of recommendations ensured that results were not only descriptive but also actionable.

Overall, the results confirm that SecureComply is capable of producing reliable, consistent, and interpretable compliance assessments across a range of SME scenarios. The variation in scoring, combined with consistent output behaviour, indicates that the system functions as intended and provides a meaningful representation of GDPR compliance posture.

8.2 Interpretation of Scores and Risk Bands

The compliance scores generated by SecureComply are mapped to four predefined risk bands: High Risk, Weak, Moderate, and Strong. These classifications are not arbitrary, but are derived from the structured scoring model and informed by both GDPR requirements and established approaches to compliance maturity assessment.

The scoring model assigns weighted values across three key categories: security measures, transparency and user rights, and internal governance. This structure reflects the prioritisation within GDPR itself, where accountability, lawful processing, and data protection safeguards form the foundation of compliance (GDPR Articles 5, 6, and 32).

The thresholds used to define each risk band were selected to reflect meaningful differences in organisational maturity. Lower score ranges correspond to the absence or weakness of multiple core controls, while higher ranges indicate consistent implementation of both technical and organisational measures. This approach aligns with metric-based compliance models, where numerical scoring is used to translate complex regulatory requirements into measurable indicators that can be easily interpreted.

(4)

To ensure interpretability, each score band represents a distinct level of compliance readiness:

- High Risk (0–49) reflects organisations with significant gaps in fundamental GDPR controls. This includes missing technical safeguards, lack of transparency mechanisms,

and minimal governance structures. Such configurations indicate a high likelihood of non-compliance with key GDPR principles, particularly those relating to security and accountability.

- Weak (50–69) represents partial compliance, where some controls are present but lack consistency or maturity. This aligns with early-stage compliance models, where organisations have begun implementing measures but have not yet achieved full operational effectiveness.

- Moderate (70–84) indicates a balanced level of compliance, where most key controls are in place but may require refinement. This reflects a transitional stage in maturity, where organisations are broadly aligned with GDPR expectations but still exhibit identifiable weaknesses.

- Strong (85–100) corresponds to high compliance maturity, where both technical and organisational controls are consistently implemented. This level reflects alignment with best practices in data protection, including clear documentation, robust security measures, and well-defined processes.

The interpretation of these bands is further supported by the principle of risk-based compliance, which underpins GDPR enforcement. Organisations are expected to implement controls proportionate to the risks associated with their data processing activities. By grouping scores into risk categories, SecureComply translates numerical outputs into practical risk levels, allowing organisations to prioritise improvements effectively.

In addition, the use of scoring bands reflects common practices in compliance and security frameworks, where maturity levels are used to benchmark organisational capability and track improvement over time. This ensures that results are not only descriptive, but also actionable and comparable across different assessments.

Overall, the scoring thresholds and associated risk bands provide a structured and justifiable method for interpreting compliance results. They bridge the gap between technical evaluation and practical understanding, ensuring that outputs are both grounded in regulatory context and accessible to SME users.

8.3 Impact of Data Quality on Results

1. **Data Quality Handling in SecureComply**

SecureComply is designed to handle imperfect real-world data through preprocessing mechanisms implemented in the ingestion layer. Missing or null values are standardised using a placeholder ("empty"), ensuring the pipeline remains robust and does not fail during execution.

However, this design decision has a direct impact on scoring outcomes:

- "empty" values are treated as missing inputs
- Missing inputs are assigned a score of 0 in the compliance engine

This introduces a systematic bias toward lower scores when data completeness is poor.

2. Data Quality Scoring Mechanism

To address transparency, SecureComply introduces a Data Quality Score, calculated during report generation. This score represents the proportion of non-missing inputs used in scoring:

- Total fields evaluated = number of controls
- Missing fields = inputs marked "empty"
- Output = percentage completeness

This is implemented in the reporting module to provide context for interpreting results.

3. Impact on Compliance Scoring

The compliance scoring engine evaluates each control independently based on provided inputs. If inputs are missing:

- Controls default to minimum score (0)
- Category scores decrease
- Overall score and band classification are negatively affected.

Scenario	Data Completeness (%)	Input Quality	Result
High completeness	>=60%	Good	Results are considered reliable for decision making
Low completeness	<60%	Low	Results may underestimate compliance due to missing data

An additional refinement introduced during development was the handling of “N/A” values. This prevented non-applicable controls from unfairly reducing the compliance score, while still preserving visibility through separate reporting of excluded controls. As a result, the final score more accurately reflected actual compliance posture rather than penalising organisational irrelevance

4. Design Justification

The decision to penalise missing data is intentional and aligns with cybersecurity best practices:

- Encourages complete and accurate data submission
- Reflects real-world audit conditions where lack of evidence indicates risk
- Maintains conservative risk assessment (preferred in compliance contexts)

5. Summary

Factor	Effect on System
Missing data	Reduces control scores
Invalid data	Rejected entirely
Low completeness	Lowers overall score
High completeness	Improves accuracy of assessment
Data Quality Score	Provides transparency to user

8.4 Benchmark Comparison Insights

The benchmark comparison feature in SecureComply was implemented to provide context to compliance scores, allowing results to be interpreted relative to a baseline rather than in isolation. A standalone score (e.g., 55/100) offers limited insight without understanding whether this reflects typical SME performance or a significant deviation.

Due to the lack of publicly available GDPR compliance datasets, synthetic data was used to construct the benchmark. Real organisational compliance data is often restricted due to privacy, confidentiality, and regulatory concerns, making it unsuitable for use within an academic project. As a result, a synthetic SME dataset was generated to simulate a range of realistic compliance scenarios.

The data generation process incorporates controlled randomness, where values are randomly selected but constrained within predefined categories and logical rules. For example, certain dependencies are enforced, such as privacy policy presence influencing clarity, or retention policies determining retention periods . This ensures that while the dataset is randomly generated, it still reflects plausible organisational behaviours rather than completely arbitrary inputs.

These synthetic records are then processed through the same ingestion, validation, and scoring pipeline as real inputs, ensuring consistency in evaluation. By generating a sufficiently large sample, the system produces a distribution of scores that approximates typical SME compliance levels, which is then used to calculate averages and quartiles for comparison .

The benchmark is generated offline and reused across executions to improve performance and ensure consistency. This avoids the computational overhead of repeatedly simulating and evaluating large datasets during each audit run.

The use of synthetic benchmarking is therefore justified as a practical compromise. It enables statistical comparison and reproducibility while avoiding the ethical and accessibility issues associated with real-world datasets. Although synthetic data cannot fully capture all real-world complexities, the structured and constrained generation approach ensures that the benchmark remains meaningful for comparative analysis.

9. Critical Analysis

9.1 Strengths of the System

SecureComply demonstrates strong input robustness, which is critical for SME-focused tools. The ingestion process automatically standardises missing or empty values rather than failing execution, ensuring the system can handle imperfect real-world data without breaking .

The system also provides clear and actionable validation feedback. Instead of rejecting data silently, it identifies exact fields and errors, improving usability and making the tool practical for non-expert users interacting via the CLI.

A key strength is the deterministic and transparent scoring model. Each control has explicit scoring rules and weightings, allowing results to be fully traceable and explainable. This avoids black-box behaviour and aligns with the requirements of compliance-based systems where justification is essential .

The inclusion of benchmarking adds meaningful context to results. Rather than presenting a standalone score, the system compares outputs against a synthetic SME baseline (e.g. average =49), allowing users to interpret whether their posture is below, at, or above expected levels .

Another strength is the separation of scoring and telemetry data. Host scan outputs (e.g. open ports, firewall status) are integrated into reports but deliberately excluded from scoring, maintaining the integrity of the compliance model while still providing useful operational context .

Finally, the system maintains strong usability for an examiner or SME user. Features such as CLI guidance (--usage, --data-instructions etc), structured HTML reporting, and optional AI summaries ensure the tool is accessible while still delivering technically meaningful outputs.

9.2 Limitations

Limitation	Impact	Notes
No regulatory validation	Outputs are not formally recognised	Would require alignment with regulatory bodies or certified frameworks
AI narrative dependency	Potential inconsistency or unavailability	Relies on external API which may fail or vary output
Reliance on synthetic data	May not fully reflect real SME environments	Generated data cannot capture all real-world complexities
Lack of real-world data	Limits empirical validation	Access to genuine GDPR audit data is restricted due to sensitivity
Evolving GDPR interpretation	Risk of model becoming outdated	Regulatory expectations and enforcement practices change over time

9.3 Design Trade-offs

Trade-off	Decision Made	Justification
Deterministic scoring vs ML/AI scoring	Used rule-based engine	Provides transparency and explainability
CLI tool vs Web app	Implemented CLI tool	No hosting complexity (ready to use tool)
Optional AI vs Fully AI-driven system	AI limited to narrative only	Prevents core system reliance on external, non-deterministic outputs

10. Future Work

10.1 Technical Improvements

SecureComply could explore integration with external compliance frameworks and automated regulatory updates, allowing the system to adapt to evolving GDPR guidance.

10.2 Additional Features

Functionality could be extended by integrating real-time data sources, expanding host telemetry to influence scoring, and supporting multi-record analysis for organisational comparisons. A lightweight web interface could also improve accessibility beyond the current CLI-based interaction.

10.3 Real-World Deployment Considerations

As SecureComply is designed as a downloadable tool, deployment would depend on SME-side requirements. This includes the ability to generate and structure compliant input data, maintain secure local environments, and correctly execute supporting components such as host scans. Basic technical familiarity, along with appropriate data handling practices, would be necessary to ensure accurate and secure use of the system.

11. Conclusion

11.1 Summary of Achievements

- Designed and implemented SecureComply, a complete GDPR auditing pipeline from data input to final report generation
- Developed a structured data ingestion module capable of handling nested SME GDPR inputs and normalising inconsistent or missing values
- Implemented robust validation logic with clear, field-level error feedback to improve usability and data quality
- Built a deterministic compliance scoring engine with transparent rules, weightings, and justifications for each control
- Created a fully automated audit pipeline, integrating ingestion, validation, scoring, and reporting into a single execution flow
- Designed and generated a professional HTML audit report, including scores, category breakdowns, top gaps, and prioritised recommendations
- Integrated benchmarking functionality using a synthetic SME dataset to provide contextual comparison of results
- Developed a synthetic GDPR data generator to simulate realistic SME scenarios where real datasets are unavailable
- Implemented optional host telemetry integration, enriching reports with real system-level security indicators while maintaining scoring integrity
- Added an AI-powered narrative module with fallback behaviour, ensuring resilience when external APIs are unavailable
- Delivered a CLI-based tool with built-in guidance and features (--help, --usage, --data-instructions, --demo, --explain, --no-ai)
- Ensured modular system architecture, allowing components such as validation, scoring, and reporting to be independently extended
- Delivered a fully runnable, self-contained tool, suitable for download and execution without complex setup or dependencies
- Published the project on GitHub: https://github.com/caseycormac/SecureComply

11.2 Key Learnings

Learning	Insight Gained
Data quality & validation	Reliable outputs depend heavily on correct and well-structured input data
Modular system design	Separating components improves maintainability and scalability
Explainability in compliance	Transparent logic is essential for trust and auditability
Working with synthetic data	Required careful design to simulate realistic SME scenarios
Usability considerations	Clear CLI guidance significantly improves user interaction
Managing external dependencies	Fallback mechanisms are necessary for reliability (e.g AI module)
End-to-end development	Building a full pipeline highlighted integration challenges across components

11.3 Final Reflection

SecureComply represents a complete and well-executed implementation of a GDPR auditing system, successfully integrating all core components into a cohesive and functional tool. The project demonstrates strong application of cybersecurity, data processing, and compliance principles, resulting in a solution that is both technically sound and practically usable. Overall, it reflects a high-quality outcome that fully meets the intended objectives and showcases the ability to design and deliver a complete system end-to-end.

12. References

1 European Commission. (2020). *Data protection and SMEs*.

2 ENISA. (2021). *Guidelines for SMEs on cybersecurity and data protection*.

3 Information Commissioner's Office (ICO). (2023). *Guide to GDPR*.

4 Brodin, M. (2019). A Framework for GDPR Compliance for Small- and Medium-Sized Enterprises. *European Journal for Security Research*, 4. doi: https://doi.org/10.1007/s41125-019-00042-z.

13. Appendices

13.1 Sample Input Data

The following example illustrates the structured JSON format used as input for SecureComply. It shows how GDPR-related organisational data is grouped into core assessment categories before ingestion and scoring.


```
[
  {
    "basic_security_measures": {
      "https_enabled": true,
      "password_storage_method": "sha256",
      "regular_security_testing": "none",
      "encryption_at_rest": "partial",
      "mfa_enforced": "none"
    },
    "transparency_user_rights": {
      "cookie_consent_mechanism": "implied",
```

```
        "privacy_policy_present": false,
        "privacy_policy_clarity": "missing",
        "lawful_basis": "legal_obligation",
        "third_party_sharing_disclosed": "partial",
        "dsar_response_time_days": 97,
        "dsar_process": "missing"
      },
      "internal_controls": {
        "data_breach_process_maturity": "informal",
        "breach_notification_hours": 142,
        "data_retention_policy": "none",
        "retention_period_days": 0,
        "record_of_processing": "automated",
        "dpia_process": "none",
        "has_dpo": "informal_role"
      }

    }
]
```