İstanbul
Bilgi Üniversitesi

Complaint Management System - Report 2

by

Mustafa Deniz Demirhas, 121200144
Emre Cem Kenarci, 121200150

Supervised by

Dr. Emel Küpçü

Submitted to the
Faculty of Engineering and Natural Sciences
in partial fulfillment of the requirements for the

Bachelor of Science

in the
Department of Computer Engineering

Jun, 2025

Abstract

This document serves as Report 2 for the Complaint Management System project, building upon the foundational work detailed in Report 1. This report focuses on realistic constraints including social, economic, and environmental impacts, comprehensive cost analysis, and adherence to engineering standards and ethics. It also provides an in-depth risk analysis regarding data privacy, LLM API dependencies, scalability, and synthetic dataset considerations, alongside a summary of project contributions.

Table of Contents

List of Tables

List of Symbols

$\sim$	Approximately / Around
$\times$	Multiplication
$\approx$	Approximately equal to
\$	US Dollar currency symbol

List of Abbreviations

AI	Artificial Intelligence
API	Application Programming Interface
CSV	Comma-Separated Values
DB	Database
FAQ	Frequently Asked Questions
HTML	HyperText Markup Language
IEEE	Institute of Electrical and Electronics Engineers
ISO	International Organization for Standardization
JS	JavaScript
KVKK	Kişisel Verileri Koruma Kanunu
LLM	Large Language Model
NSPE	National Society of Professional Engineers
NLP	Natural Language Processing
SVM	Support Vector Machine
TL	Turkish Lira
VS Code	Visual Studio Code

1 Realistic Constraints

1.1 Social, Environmental and Economic Impact

The main purpose of this project is to make the process of users reporting their feedback digital and accessible. Instead of traditional physical complaint boxes, a web-based QR integrated system was developed. The interface is especially designed to be compatible with mobile devices, allowing users to provide feedback mostly using their mobile devices. This saves time and eliminates the need to physically fill out a form or reach an authorized person.

In addition, the automatic classification and reporting features offered by the system enable corporate employees to process complaints faster and contribute to improving service quality. Structured storage of all submissions also allows management teams to track recurring issues, monitor department-specific feedback trends, and take proactive action accordingly.

1.2 Cost Analysis

Since there was no hardware development process in the project, there were no physical parts or equipment costs. However, both manpower and some external services were used during the software development process. These costs are explained in detail below.

We took part in the project as two people and worked actively in research, software development, model training, and integration processes for about 5 months.

If the average salary of junior-level software developers in Turkey (for $\sim$2025) is estimated to be around 1.5 to 2 times the minimum wage:

- Net minimum wage (2025): 22,104.67 tl
- Junior developer wage: 22,104.67 tl$\times$ 1.7 $\approx$ 37,500 tl
- 2 people $\times$ 5 months $\Rightarrow$ **375,000 tl**

Google Colab Pro service was preferred in the model training phase of the project.

- Colab Pro cost: \$10 per month
- Exchange rate (approx.): \$1 = 38 tl
- Duration of use: 2 months

- Total cost: \$10 × 2 × 38 tl = **760 tl**

This project is a system developed for corporate integration, not for direct sale to a company. Therefore, hosting and domain costs are not directly paid by the project team; it is assumed that they will be covered by the organization requesting the integration.

Throughout the development process, the team used their own personal computers and open-source tools.

- No hardware upgrades or special licenses were purchased
- Code editors and packages (VS Code, Django, Angular CLI, etc.) are free to use
- Artificial data was generated using free versions of LLM services

Item	**Cost (tl)**
Manpower (2 people, 5 months)	375,000 tl
Google Colab Pro (2 months)	760 tl
Hosting / Domain	0 (to be covered by institutions)
Hardware / Licenses	0
Total	**375,760 tl**

Table 1: Estimated Cost Summary

1.3 Standards & Ethics

The AI-Enhanced Complaint Management System was designed and developed in compliance with relevant engineering standards and ethical guidelines to ensure reliability, security, and responsible implementation.

Software and System Design Standards While developing the web application, industry standards and best practices were followed, and a modular client-server architecture (Angular frontend, Django REST backend) was implemented to ensure maintainability and scalability. For API design, RESTful principles were followed for compatibility. Complete adaption of standards like **ISO/IEC/IEEE 29119 (Software Testing)**[1] or **ISO/IEC 25010 (SQuaRE - Quality Models)**[2] is extensive for this project's scope, yet core principles from these standards regarding functional suitability, usability, and reliability were useful in guiding the development and manual testing processes.

Data Management, Security, and AI Ethics As the system handles complaint data ,which may potentially be sensitive, principles from **ISO/IEC 27001 (Information Security)**[3] were considered, especially for data integrity and controlled access through user authentication. As the current project supports Turkish language, in order to achieve full live deployment, compliance with **KVKK (Kişisel Verilerin Korunması Kanunu)**[4] is required. This influenced the choice of allowing anonymous complaint submission options and it was an important consideration for the AI-assisted data augmentation process, where there were efforts to generate anonymous and hypothetical scenarios so that using real personal details in training the ML model were avoided. The

Engineering Code of Conduct The project was constructed in accordance to the fundamental canons of engineering ethics (mainly following IEEE/NSPE). To this end, there are some important commitments, such as: **hold paramount the public welfare (NSPE Canon 1, IEEE Principle 1)**[5]; the system aims to improve the safety and well-being of the occupants by enabling faster and more accurate responses to complaints. Another important point is **areas of competence (NSPE Canon 2, IEEE Principle 5)**[6] which was followed by using established machine learning algorithms for classification and utilizing an advance pre-trained LLM for summary reports rather than overestimating our development scope. All our findings, which includes the 0.9564 test accuracy that was achieved and the known limitations of AI-generated data, are reported in the papers in accordance with **honesty and integrity (NSPE Canon 3, IEEE Principle 3 & 7)**. The system was designed to serve as a trustworthy tool for administrators, showing the team's duty as **faithful agents (NSPE Canon 4, IEEE Principle 4 & 6)** with careful considerations to data privacy (KVKK) and the responsible use of AI.

2 Risk Analysis and Limitations

2.1 Data Privacy KVKK Compliance

Some personal data are directly stored in the database. In particular, the following two tables require attention in terms of KVKK compliance:

- `employees` table: stores username, email, role, and password — all considered personal data.

- `complaints_suggestions` table: may contain email, submission timestamps, and user-written descriptions.

Under Law No. 6698 (KVKK), such data must be processed with explicit user consent, for specific and lawful purposes, and protected against unauthorized access.

Risks:

- Collecting personal data without user consent violates KVKK.
- Storing unencrypted passwords or emails may lead to unauthorized access.
- Retaining user data indefinitely may conflict with data minimization principles.

Mitigation Measures:

- Displaying a consent checkbox and privacy notice in the user interface.
- Hashing sensitive fields such as passwords.
- Enforcing role-based access control for employee data visibility.
- Providing users the ability to request data deletion.

2.2 LLM API Dependency & Security Risks

The current AI-based complaint summary report feature relies on an external LLM API (Groq with Llama-3 models). As of writing this paper, Groq currently offers generous free tier limits, yet this dependency brings a few risks that cna be discussed:

- **Service Availability and Changes:** In the future, the LLM service provider may change its terms of service, API availability, discontinue the current model, or pricing terms, potentially impacting the summary report generation feature's functionality and cost-effectiveness in the future.
- **Performance Variability:** For LLMs, especially ones that use external APIs, the quality of the response or response time can vary, in which case user experience may be affected for the summary report generation feature, even though current testing with Groq and Llama-3 has shown satisfactory performance.

- **Data Privacy (External Processing):** Although the API providers are widely known and used, sending any data to an external API inherently brings about some concerns regarding the degree of trust in the provider's data handling and privacy standards, even when it's from a reputable company.

Given that the complaint summary report feature is intended for administrators and it is primarily designed to be used for periodic reporting rather than continuous and high-frequency requests, the current risk level is deemed feasible. However, as future policy changes regarding API providers aren't known, in order to maintain long-term scalability-or in case usage rates change-some options such as self-hosting open-source LLMs or more robust enterprise-level API solutions can be considered.

2.3 Scalability Performance

During development, the project was only tested in a local environment. Therefore, there has been no comprehensive performance testing of how the system will behave under real user traffic.

Once deployed, the system is expected to handle user requests simultaneously, run the classification model quickly, and access the database efficiently. However, it is currently unclear how well these operations will perform under high user load.

Measures that can be taken:

- Performance tests that simulate system behavior under load should be performed.
- Classification and reporting processes running in the background should be designed asynchronously if possible.
- If necessary, the system should be reconfigured within a scalable cloud infrastructure.

2.4 Synthetic Dataset Risks

The use of Large Language Models (LLMs) for augmenting the training dataset (`sikayet_v1.csv`) was crucial for addressing initial data scarcity. However, this approach carries inherent risks:

The usage of Large Language Models(LLMs) in augmenting the training dataset (`sikayet_v1.csv`) was a crucial and necessary application for addressing data scarcity. However, this approach inevitably carries some risks too:

- **Limited Real-World Diversity:** Despite utilizing methods such as using multiple LLMs, assigning personas, and tuning temperature settings, AI-generated data still may not fully capture the entire possibility of linguistic differences or unforeseen complaint types that may be encountered in organic user submissions. This can be a limitation on the model's ability to generalize to real-world scenarios.

- **Potential for Bias Propagation:** LLMs can also reflect inherent biases that exist in their training data, which can involuntarily introduced to the synthetic dataset and used to train the classification model, leading to wrongful and biased outcomes.

- **Overfitting to Synthetic Patterns:** The classification model can learn specific grammatical structures or sentence styles that form during the AI generation process, which can potentially cause underperformance on pure real-world data as opposed to the test set from the synthetic dataset.

- **Dataset Stagnation:** If the dataset doesn't get continuous updates, the AI-generated portion of the dataset may over time become less and less representative of real-world complaint data, which can cause model to lose value for real-world applications.

Mitigation Approaches:

In order to mitigate these risks to some degree, the strategy included using diverse LLMs and prompting techniques. The model was also evaluated with hand-written test complaints written by the team members of the project, which indicated promising generalization. For a successful real-world implementation, continuous monitoring of the model and periodic retraining with new, validated real-world data (and perhaps refreshed synthetic data) would be crucial. This project acknowledges that while AI can be a powerful tool, maintaining a dataset that can truly reflect real-world data diversity is an ongoing challenge in the world of Computer Engineering that is yet to be solved.

3 Contributions

We developed this project as a two-person team; Emre Cem Kenarcı and Mustafa Deniz Demirhas. Our responsibilities were divided primarily based on the needs of the project and overall workload.

Emre Cem Kenarcı (Infrastructure & Full Stack Development)

- Designed and implemented the database schema using MySQL.
- Developed the backend using Django and built RESTful APIs.
- Designed and implemented the frontend with Angular (HTML, CSS, TypeScript).
- Integrated frontend, backend, and database for seamless communication.

Mustafa Deniz Demirhas (AI & Backend Development)

- Trained and evaluated a complaint classification model (SVM-based), including the generation of synthetic training data.
- Integrated LLM-powered report generation via external API.
- Developed backend endpoints related to AI functionalities.
- Assisted in frontend development when needed, especially in resolving specific component issues.

We shared the responsibility for documentation, report writing, and presentation preparation equally. Throughout the five-month development process, we regularly met with our supervisor via bi-weekly Zoom meetings and maintained daily communication through WhatsApp.

4 Conclusion

This project successfully developed an AI-enhanced Complaint Management System, featuring an automated Turkish complaint classification module and LLM-powered report summarization. A key feature was the AI-assisted synthetic data generation for the custom dataset (`sikayet_v1.csv`) to get 3899 samples to be used with an optimized LinearSVC model to achieve a test accuracy score of 0.9564. The system utilizes a Django and Angular architecture, designed for scalability.

The Project's social impact lies in improving service accessibility, quality, and effectiveness, which can all lead to better well-beings for the users of this system. Economically, the main cost was manpower (375,000 TL plus

760 TL for Colab Pro, as detailed in Section 1.2), with potential more future costs that depend on the scale of deployment. Development followed Software Engineering best practices and ethical guidelines, including KVKK considerations, and principles from ISO/IEC standards for quality, and security, and NSPE/IEEE canons for integrity of the engineering profession. Key risks (Section 2) included LLM API dependency and the great challenge of ensuring real-world diversity in synthetically generated datasets. These risks were identified and mitigated where possible, while acknowledging that continuous monitoring is important.

Future improvements can focus on real-world live deployment, continuos dataset improvements, multilingual support, and adapting the system yo to work as a framework for other complaint subjects. Advanced analytical features and enhanced LLM report creation capabilities also remain as potential improvements.

In summary, this project demonstrates the effective use of AI-augmented data to build a high-performing complaint classification system, offering a valuable tool for organizations and a strong foundation for future advancements in intelligent feedback management.

In summary, this project, the Complaint Management System, demonstrates the effective use of AI-augmented data to build a sufficient complaint management system, offering a valuable tool for organizations and a strong foundations for future advancements in automated feedback management.

References

[1] "Iso/iec/ieee 29119 (series) software and systems engineering – software testing," YYYY.

[2] "Iso/iec 25010:2011 systems and software engineering – systems and software quality requirements and evaluation (square) – system and software quality models." https://www.iso.org/standard/35733.html, 2011. Accessed: Month DD, YYYY.

[3] "Iso/iec 27001:yyyy information technology – security techniques – information security management systems – requirements," YYYY.

[4] Türkiye Büyük Millet Meclisi, "Kişisel verilerin korunması kanunu (kanun no. 6698)." https://www.mevzuat.gov.tr/mevzuatmetin/1.5.6698.pdf, 2016.

[5] National Society of Professional Engineers (NSPE), "Code of ethics for engineers." https://www.nspe.org/resources/ethics/code-ethics, 2025. Accessed: 20.06.2025.

[6] Institute of Electrical and Electronics Engineers (IEEE), "Ieee code of ethics." https://www.ieee.org/about/corporate/governance/p7-8.html, 2025. Accessed: 20.06.2025.