Executive Summary

Retail organisations today face the challenge of managing vast quantities of data as they operate across physical and digital environments. Transactions, customer feedback, and service requests occur on numerous platforms—ranging from traditional point-of-sale systems to e-commerce websites and mobile applications—and often generate large volumes of structured and unstructured data. In addition, rapid technological developments have given rise to so-called “untact” services, which refer to low-contact or fully automated customer interactions, including self-service kiosks, online customer support chats, and mobile order platforms (Kim et al., 2018; Lee & Lee, 2020). These new modes of communication and transaction require organisations to establish robust and flexible database infrastructures that accurately record customer journeys while ensuring compliance with legal and regulatory standards such as the European Union’s General Data Protection Regulation (GDPR).

The project detailed in this summary involved designing, developing, and deploying a logical database aimed at unifying customer, product, and transactional information within a single, consistent schema. The solution was required to capture and manage client feedback, orders, transactions, returns, and other customer service requests. Since many retailers now integrate both physical and digital touchpoints, the database needed to incorporate e-commerce functionality alongside in-store operations. The ultimate goal was to improve customer satisfaction by streamlining request handling and feedback loops, reduce operational friction through data centralisation, and ensure compliance with relevant privacy regulations. This document provides an easy-to-understand account of the project’s objectives, methodology, design decisions, key findings, and recommendations, highlighting both technical and managerial considerations.

1. Work Carried Out

The driving force behind this project stemmed from the need for a unified repository that could track every phase of a retail customer’s interaction, from initial inquiries about products through to post-purchase feedback or refund requests. Prior to this undertaking, many retail organisations relied on fragmented systems in which customer data, order records, and feedback logs were stored on separate platforms, causing duplication, inconsistencies, and difficulties in compliance tracking (Stone et al., 2020). The new database aimed to integrate these disparate datasets into a single “source of truth,” improving data accuracy and operational efficiency.

During an initial requirements-gathering phase, the project team consulted stakeholders from departments such as sales, customer support, logistics, finance, and compliance. This cross-departmental input clarified the variety of data points necessary for daily operations. For instance, the customer support division required speedy access to client histories and open requests, while the finance team needed accurate transaction logs for auditing and tax reporting. The logistics function required updates regarding product returns, exchanges, or restocking. Collecting these perspectives helped the team map out the entities and relationships integral to the future data model (Abiteboul et al., 1995).

After gathering requirements, the team produced an Entity Relationship Diagram (ERD) that encompassed entities including Client, Employee, Department, ClientRequest, CustomerFeedback, Order, Transaction, Product, Store, and Website. The Client entity served as the central hub for customer information. Employee and Department entities were required for internal management and delineation of responsibilities, ensuring that all support requests and feedback looped back to the relevant staff. Orders and Transactions were split into separate entities so that each transaction could be analysed financially and each order could be tracked according to shipping or pickup details. The ClientRequest and CustomerFeedback entities captured service issues and user impressions, respectively, while Product and Store data provided insights into the lifecycle of merchandise across physical and online retail channels.

In order to validate the proposed schema, a prototyping phase followed, during which sample data was inserted into the schema using a SQL-based system (in this case, MySQL). The project team tested referential integrity, checked for the correct enforcement of cardinalities, and reviewed constraints on data fields. User acceptance testing involved simulating typical customer-service workflows, such as creating a new order, tagging a transaction, and raising a return request linked to the same client record. This comprehensive approach ensured that the database design remained cohesive and user-friendly. Furthermore, the team carried out a security and compliance review, mapping project requirements to GDPR principles such as data minimisation, accuracy, purpose limitation, and retention policies (Zikopoulos & Eaton, 2011).

Upon successful testing, a final documentation package provided detailed instructions on database operations, including data backup, data quality checks, user roles, and permissible data access levels. A compliance checklist addressed how the system could remain adaptable to ongoing regulatory changes, particularly regarding personal data privacy. By the end of the project, the logical database design provided a robust structure that linked operational processes to customer and transaction data while providing a clear path for future enhancements in analytics or scalability.

2. Database Modelling

Throughout the project, the team adhered to relational database best practices, with particular attention to normalisation, referential integrity, and transaction management (Elmasri & Navathe, 2021). A relational approach was chosen due to the highly structured nature of retail data. Orders, client details, and financial transactions typically fit well into discrete tables, each with consistent fields that benefit from primary and foreign key relationships.

The design included primary keys (PKs) to ensure unique identification of records in each table (for instance, clientID in the Client entity). Foreign keys (FKs) linked data across tables, allowing queries to reference related information without redundancy. A typical example was the presence of a clientID FK in the Order table, ensuring that every order record was associated with a valid client in the Client table. This referential integrity prevented orphan records or inconsistencies where an order might be recorded without a corresponding client.

Normalisation was crucial for minimising duplication and update anomalies, so the team aimed for at least Third Normal Form (3NF). In doing so, the design reduced redundant data storage; for example, instead of storing a client's address in multiple places, the database kept a single reference in the Client table that was linked to Orders or other relevant records. This structure facilitated more reliable updates, since any changes to a client's address would cascade correctly across the system (Kleppmann, 2017).

Despite the strengths of a relational approach, the rigidity of schemas can present challenges if business requirements evolve rapidly. Adding new attributes or entities can require downtime or complex migrations (Stone et al., 2020). This limitation might be relevant if an organisation planned frequent expansions of its data model or intended to store large volumes of semi-structured data, such as real-time social media feeds. Nonetheless, for the scope of this project, relational technology offered the correct balance of reliability, data integrity, and a well-understood ecosystem for querying and indexing.

3. The Outcomes of DBMS Analysis

The project team undertook a comparative study of SQL-based and NoSQL-based database management systems to identify the most suitable technology for the client's requirements. SQL databases, such as MySQL or PostgreSQL, rely on a structured, table-based model. They support powerful, standardised queries (SQL) and enforce ACID (Atomicity, Consistency, Isolation, Durability) transactions. In a retail environment, where transactional integrity is paramount and data is highly relational, SQL is often the default choice because it minimises the risk of anomalies in financial or operational data. The ability to run complex joins and aggregated queries is also invaluable, especially for generating sales or performance reports at departmental or corporate levels (Sarkar & Roychowdhury, 2019; Elmasri & Navathe, 2021).

NoSQL systems—such as MongoDB, Cassandra, or DynamoDB—offer more flexible schemas and can handle large volumes of unstructured or semi-structured data. These solutions typically excel in scenarios requiring high-velocity inserts or real-time analytics, often scaling horizontally across distributed nodes more seamlessly than a traditional relational system (Adam et al., 2021). They can also adapt quickly to changing data structures without requiring the same degree of downtime or data migration. However, NoSQL systems may not provide the same level of ACID transaction support as relational databases, although newer versions of certain NoSQL platforms have introduced improved transaction capabilities.

For the purposes of this project, the high degree of relational interdependence between clients, orders, transactions, and feedback made a SQL-based DBMS the most suitable option. The ability to enforce referential integrity through primary and foreign keys, along with immediate consistency in financial transactions, aligned with operational priorities. Should the retail organisation later expand into areas requiring rapid ingestion of unstructured data—for instance, real-time sentiment analyses of social media activity or large-scale event logging—a hybrid approach could be adopted by integrating a NoSQL platform alongside the main SQL environment. This dual structure would maintain the transactional reliability of the core relational schema while providing the flexibility and scalability needed for big data analytics (Zikopoulos & Eaton, 2011).

4. GDPR and Compliance

In an era defined by heightened consumer data privacy awareness and stringent regulatory frameworks, data handling requirements formed a pivotal aspect of this project. The GDPR principles emphasise lawfulness, fairness, transparency, purpose limitation, data minimisation, accuracy, storage limitation, and confidentiality. The team's data model was designed around collecting only essential personal details necessary for order processing, feedback handling, and record-keeping. For instance, fields for customer demographic information, contact details, and payment references were included only if they served a clear organisational purpose aligned with a legitimate business interest (Bordoloi et al., 2023).

Because GDPR also mandates processes for customer access, correction, and deletion of personal information, the database included pathways for swift retrieval and modification of records upon request. Role-based access control structures were introduced to ensure that only authorised personnel (for example, specific customer service or finance employees) could view personal data. Database encryption, both at rest and in transit, was recommended to secure data against unauthorised breaches. To address GDPR's storage limitation requirement, data retention schedules and automated archival features were proposed: data older than a certain threshold could be anonymised or removed, depending on local legal obligations for record-keeping and auditing (Jalil et al., 2021).

Further legal requirements, including consumer protection legislation, were considered. These laws often involve strict guidelines on product returns, refunds, and warranty periods. The ClientRequest entity in the final model accommodates parameters such as request type (return, refund, exchange) and status (opened, in-progress, resolved, urgent), ensuring that these interactions are tracked with timestamps that facilitate compliance audits. In addition, potential expansions of the data model to incorporate marketing consents or ePrivacy regulations were planned, including fields to log whether a customer has agreed to receive promotional communications (Kleppmann, 2017).

5. Conclusions and Recommendations

This project demonstrated how a well-structured relational database can significantly enhance the ability of a retail organisation to manage customer interactions efficiently. By consolidating client data, purchase records, product details, and feedback loops into a single logical schema, the system facilitates streamlined operations, more accurate analytics, and better control over data privacy. The relational model's emphasis on referential integrity ensures that changes in customer or product information propagate consistently throughout the database, reducing the incidence of errors and inconsistencies. Moreover, the integrated design supports improved customer service, since employees can rapidly locate relevant details about previous purchases, open tickets, or stored feedback.

The chosen SQL-based DBMS was justified by the strongly relational nature of the data. A SQL solution remains the ideal foundation for managing orders, transactions, and client requests that require consistency, ACID properties, and readily comprehensible schemas (Sarkar & Roychowdhury, 2019; Adam et al., 2021). Nonetheless, an eventual hybrid approach could be beneficial if the retailer expands into new domains requiring large-scale or real-time data processing.

In the immediate future, the recommendation is to maintain rigorous access control protocols and encryption to ensure GDPR compliance and protect sensitive customer data (Lee & Lee, 2020). Regular data auditing, including scheduled reviews of data quality, consistency, and retention compliance, is recommended to reduce legal and operational risks. Automated alerts for overdue customer requests can further enhance customer satisfaction by preventing support staff from missing time-sensitive issues. The introduction of more advanced reporting or dashboards can also offer powerful insights into sales trends, feedback loops, and efficiency metrics, enabling decision-makers to rapidly identify areas for improvement (Stone et al., 2020).

A secondary recommendation is to plan for controlled schema evolution to accommodate new products, payment methods, or service channels. Although changes to a relational schema can sometimes require downtime, careful versioning and migration scripts can mitigate disruptions. As digital touchpoints continue to proliferate, the organisation may also need to integrate additional data sources, such as website clickstream data or social media sentiment, potentially supplementing the core relational database with a NoSQL store optimised for such use cases.

Furthermore, the organisational culture around data usage should involve periodic training of employees at all levels to handle personal data responsibly, use the new system effectively, and follow best practices for maintaining data accuracy. In parallel, ongoing updates to legal compliance standards should be monitored, ensuring that database policies and practices remain in line with evolving regulations and industry benchmarks. By approaching data governance as an iterative and continuous process, the organisation can sustain a robust level of data quality, security, and compliance over time.

Overall, this new Retail Customer Service Database provides a future-proof, customer-centric, and legally compliant infrastructure for capturing and analysing critical information across numerous retail channels. Its immediate benefits include increased operational efficiency, more accurate record-keeping, and a heightened capacity to respond to customer requests. Its longer-term potential involves using the relational structure as a foundation for advanced data analytics, predictive modelling, and further technological innovations (Kleppmann, 2017). With a careful and strategic approach to database evolution, the organisation can adapt readily to emerging consumer trends, regulatory changes, and shifts in the competitive landscape. By prioritising data integrity, security, and role-based permissions, the database not only meets current operational demands but also positions the organisation to thrive in an increasingly data-driven retail environment.

References

Abiteboul, S., Hull, R. & Vianu, V. (1995) *Foundations of Databases*. Boston: Addison-Wesley. Available from: https://www.researchgate.net/publication/225035500_Foundations_of_Databases [Accessed 17 January 2025]

Bordoloi, S., Fitzsimmons, J.A. & Fitzsimmons, M.J. (2023) *Service management: operations, strategy, information technology*. 10th edn. New York: McGraw-Hill.

Elmasri, R. & Navathe, S.B. (2021) *Fundamentals of database systems*. 8th edn. London: Pearson.

Jalil, A., Kaur, H. & Jogia, A.V. (2021) 'The Impact of E-Service Quality on Customer Loyalty in the United Kingdom' *International Journal of Applied Business and Management Studies*, 6(1), 69–90. Available from: 04_JALIL-KAUR-JOGIA_PB.pdf [Accessed 17 January 2025]

Kim, R., Jeon, M., Lee, H., Choi, J., Lee, J., Kim, S., Lee, S., Seo, Y. & Kwon, J. (2018) *Trend Korea*. Seoul: Mirae Book Publishing Co.

Kleppmann, M. (2017) *Designing data-intensive applications*. Sebastopol, CA: O'Reilly.

Lee, S.M., Lee, DH (2020). '"Untact": a new customer service strategy in the digital age'. *Service Business* 14: 1-22. [Accessed 17 January 2025]

Sarkar, T. & Roychowdhury, S. (2019) *Data wrangling with Python: creating actionable data from raw sources*. 1st edn. Birmingham: Packt Publishing Ltd.

Stone, M., Aravopoulou, E., Girardi, G.G.G., Weinzierl, L., Laughlin, P., Stott, R. & Todeva, E. (2017) 'How platforms are transforming customer information management', *The Bottom Line*, 30(3): 216–235. Available from: https://ssrn.com/abstract=3109797. [Accessed 17 January 2025]

Zikopoulos, P. & Eaton, C. (2011) *Understanding big data: analytics for enterprise-class Hadoop and streaming data*. New York: McGraw-Hill.