Research Article

A Web-Based Credit Card Payment Architecture for Dealer Portals: Android POS Integration, Microservice Design, and Behavioural Segmentation for Data-Driven Dealer Management

Adnan Erdoğan[1*], Hüseyin Oktay Altun[2]

[1] Turkcell Ödeme ve Elektronik Para Hizmetleri A.Ş. (Paycell R&D Center), Orcid ID: https://orcid.org/0009-0004-8271-1865, E-mail: adnan.erdogan@turkcell.com.tr
[2] Boğaziçi University, Institute of Data Science and Artificial Intelligence, Istanbul-Turkey, Orcid ID: https://orcid.org/0000-0002-8875-6878, E-mail: oktay.altun@bogazici.edu.tr
[*] Correspondence: adnan.erdogan@turkcell.com.tr; Tel.: +905332102491

Received: 22 July 2025
Revised: 28 October 2025
2nd Revised: 28 November 2025
3rd Revised: 02 December 2025
Accepted: 09 December 2025
Published: 24 December 2025

Reference: Erdoğan, A., & Altun, H. O. (2025). A web-based credit card payment architecture for dealer portals: Android POS integration, microservice design, and behavioural segmentation for data-driven dealer management. *The European Journal of Research and Development, 5*(1), 635–647.

Abstract

Digital transformation in financial services has accelerated the need for secure, scalable, and user-centric payment infrastructures across various industries This study presents the design and implementation of a web-based credit card payment architecture integrated into the Dealer Web Portal (BWP), enabling dealer-initiated bill payments through Android POS ecosystem. The work covers three major dimensions: the development of a microservice-based web architecture using REST/SOAP services; real-time, bi-directional communication between the web portal and Android POS devices; and an unsupervised machine learning framework for behavioural segmentation using large-scale bill payment data. Multiple clustering algorithms, including K-Means, DBSCAN, Mean Shift, Spectral Clustering, and Hierarchical Clustering, were evaluated, with K-Means yielding the most meaningful segmentation results based on Purity, NMI, and Silhouette metrics. Segment outputs enabled dynamic commission policies, targeted dealer *interventions, and time-series behavioral insights. The results demonstrate that the proposed architecture significantly enhances operational efficiency and data-driven decision making. This study provides one of the first integrated examples of Android POS–web portal interoperability combined with large-scale behavioural segmentation in Türkiye's bill-payment ecosystem.*

Keywords: Web-based payment systems, Android POS, machine learning, segmentation analysis, big data analytics, microservice architecture.

1. Introduction

The global shift from cash-based to digital payments has been driven by increasing internet penetration, the proliferation of mobile devices, and evolving consumer expectations. Recent studies [1,2,7] highlight the exponential growth of digital payment methods and emphasize the need for businesses to adopt secure and scalable payment infrastructures. In Türkiye, millions of customers routinely pay institutional bills—telecommunications, electricity, gas, insurance—through dealer locations [7]. Despite the increasing adoption of digital platforms, conventional dealer-based workflows continue to dominate specific segments of the billing market.

The Dealer Web Portal (BWP) has long served as a primary interface for bill payment transactions, but historically, it only supported cash-based payments. This limitation required dealers to divert customers to separate ÖKC (fiscal POS) terminals for credit card transactions, causing fragmented user flows, operational inefficiencies, and inconsistent data collection. The absence of an integrated credit-card payment option also limited ability to capture behavioural data across channels.

The objective of this study is to modernize the existing infrastructure by enabling credit card payments directly within BWP and integrating these processes with the Android POS system. Additionally, this work introduces a comprehensive segmentation analysis of behavioural payment data to inform dealer operations, commission strategies, and customer experience enhancements [14,15]. This research is significant for combining real-time POS integration with large-scale behavioural clustering—an area that remains underexplored in the digital payment's literature, particularly within the context of Türkiye's institutional billing market [7,8,14,15].

2. System Overview

2.1. Architecture Summary

The architecture of the proposed system is structured as a unified framework that connects the web-based dealer interface, the backend microservice environment, and the

Android POS terminal [9]. The Dealer Web Portal (BWP) acts as the front-facing interaction layer, allowing dealers to perform invoice queries, initiate payments, and monitor outcomes. Beneath this interface, the backend microservices manage the operational workflow by validating invoices, triggering payment sessions, coordinating transaction logic, and maintaining secure communication with external systems. Once a payment request is initiated on the portal, the backend transmits the required transaction information to the Android POS, which then conducts the card-based payment through its customer-facing interface [9,10]. Upon completion, the POS communicates the transaction result back to the backend through callback mechanisms, ensuring that the system immediately updates both its internal records and the dealer-facing interface.

In this setup, the BWP ensures usability and accessibility, the backend ensures security and business-level consistency, and the POS handles the execution of card transactions. Together they form a cohesive end-to-end payment architecture capable of supporting high-volume bill payment workflows with reliability and low latency. The high-level structure of this architecture is illustrated in **Figure 1**.

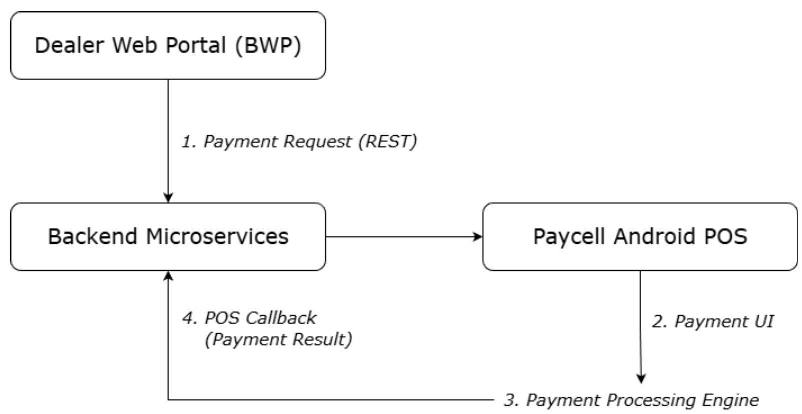

Figure 1. High-Level System Architecture

2.2. Development Environment

The development environment supporting the proposed payment architecture was established using enterprise technologies selected for their stability, performance, and interoperability with existing payment systems. The backend services were implemented using Java 8 and above, an industry-standard language for financial and transactional platforms, supported by both REST and SOAP service frameworks to ensure compatibility with legacy integrations as well as modern microservice-based components. Continuous integration practices were maintained through Maven-based builds and Jenkins pipelines, with Git repositories and Artifactory providing structured version control and artifact management.

Quality assurance was integrated into the development pipeline through automated static analysis tools such as SonarQube and Checkmarx, enabling early detection of security vulnerabilities and code-level defects. The production runtime environment operated on WebLogic Server within a Unix-based infrastructure, while data persistence was handled using Oracle Database, ensuring the ACID properties crucial for financial operations. Together, this development ecosystem created a robust, extensible foundation capable of supporting high-volume transactions, real-time POS communication, and the analytics-driven modules incorporated into the project.

Table 1. Technologies and Components Used in the Development Environment

Category	Technology / Tool
Programming Language	Java 8+
Service Frameworks	REST Services, SOAP Services
Build / CI	Maven, Jenkins
Version Control	Git, Artifactory
Code Quality & Security	SonarQube, Checkmarx
Runtime Environment	WebLogic Server on Unix
Database	Oracle DB
POS Integration	Android POS APIs

2.3. Payment Workflow

The payment workflow follows a sequential, event-driven structure that links the Dealer Web Portal, backend services, and the Android POS into a unified transactional pipeline. When a dealer initiates a payment on the BWP interface, the system first validates the invoice through the backend microservices, which ensure that the bill information, amount, and payment eligibility comply with the required business rules. Once validated, the backend triggers the POS device by transmitting a structured payment request that encapsulates all necessary transaction parameters. The POS then presents the card payment interface to the customer, processes the financial transaction through its secure hardware components, and records the result.

Upon completion of the card transaction, the POS device sends a callback message containing the payment result—successful, declined, or errored—back to the backend services. This callback enables the system to finalize the transaction by updating records, generating a synchronized payment result, and immediately reflecting the outcome on the dealer's screen. Because the workflow depends on real-time communication between the web platform, the backend, and the POS terminal, the architecture supports low latency, high reliability, and consistent transactional integrity across all components. The overall process is illustrated in Figure 2.

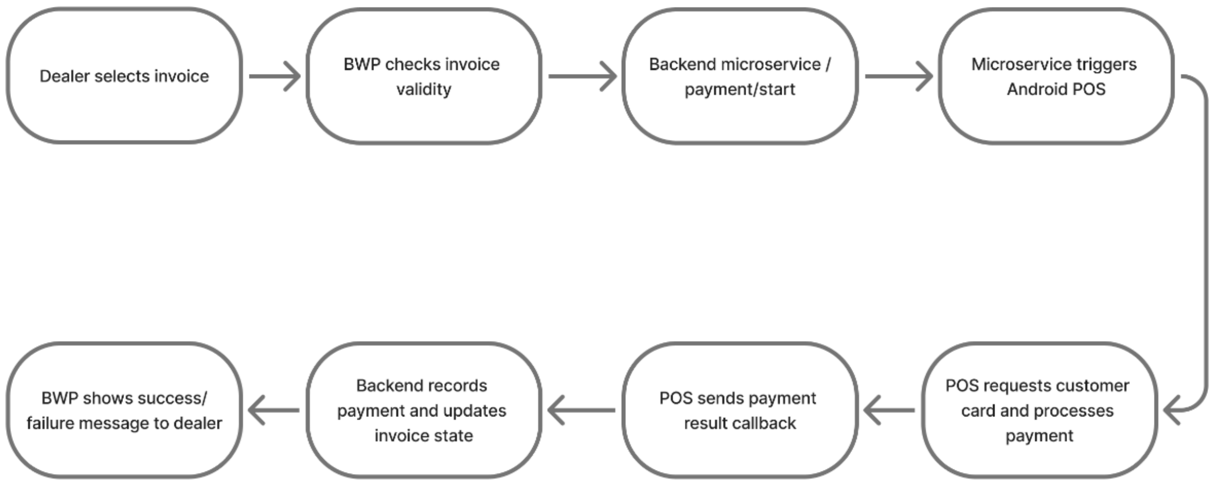

Figure 2. The end-to-end payment workflow begins when the dealer selects an invoice on the Dealer Web Portal, prompting the system to verify the invoice's validity. Once validation is complete, the portal initiates the payment sequence by calling the backend microservice at the /payment/start endpoint. The backend then activates the Android POS device, which requests the customer's card and processes the payment locally on the terminal. After the transaction is completed, the POS sends a callback to the backend indicating whether the payment succeeded or failed. The backend records the result, updates the status of the corresponding invoice, and forwards the final outcome to the web portal, where the dealer receives a clear success or failure message.

2.4. Android POS Integration

The integration with the Android POS device forms the core of the system's ability to execute credit card transactions securely and reliably. The POS operates as an intelligent payment terminal that interacts with the backend microservices through standardized REST endpoints, allowing the web platform to extend its functionality to the physical device without requiring any manual intervention from the dealer. When a payment is initiated on the Dealer Web Portal, the backend constructs a structured payment instruction containing the invoice details, payment amount, and a unique transaction reference. This information is transmitted directly to the POS device, which then activates its native payment interface and prompts the customer to present their credit or debit card.

The POS device is responsible for all card-level security, including chip, magnetic stripe, or contactless processing, as well as PIN verification when required [11]. These operations are performed using its embedded EMV-certified software stack, ensuring strict compliance with national and international payment standards [11,12]. Recent security analyses of EMV contactless payment systems further underline the importance of robust protocol design in such environments [13]. Once the card transaction is completed, the POS system immediately generates a callback to the backend containing the payment status, authorization codes, and finalized transaction details. The backend then synchronizes this result with the Dealer Web Portal, enabling real-time visibility and eliminating discrepancies between the POS and the web environment.

This integration approach ensures a clear separation of responsibilities: the backend manages business logic and transaction routing, the POS device executes secure card processing, and the web portal maintains the dealer-facing experience. The workflow establishes a tightly coordinated but decoupled design that allows updates, feature expansions, and scaling operations to be carried out independently across system components. The overall POS integration pipeline is illustrated in Figure 2.

2.5. Integration and Deployment Strategy

The integration and deployment strategy for the system was designed to ensure a seamless transition from development to production while maintaining high availability, transactional consistency, and backward compatibility with existing dealer workflows. The integration phase centered on progressively combining the front-end portal components, backend microservices, and the Android POS interfaces within controlled staging environments. During this phase, each module was validated through iterative testing cycles that evaluated functional correctness, API stability, and system-level interoperability. Particular emphasis was placed on ensuring that the POS callbacks, invoice validation sequences, and multi-step payment flows behaved consistently under varying network and load conditions.

Deployment followed a staged approach, beginning with isolated development sandboxes, moving into a pre-production environment, and finally propagating into the live production system using controlled release procedures. This pipeline leveraged continuous integration and delivery practices, where automated build verification, regression suites, and static analysis checks were executed before any component was promoted to the next stage. Rollout into production employed a blue–green deployment model, allowing the new services to operate in parallel with the existing infrastructure until stability was fully verified. This strategy minimized downtime and enabled rapid rollback if the need arose. Once verified, the new payment functionality was activated for dealers through configuration-based toggles, enabling a gradual and monitored adoption of the credit card payment capabilities. The overall integration and deployment approach ensured that the system could scale reliably while accommodating iterative enhancements without disrupting ongoing operations.

2.6. AI-Supported Dealer Support Platform

The *Sesini Duyuyoruz* ("We Hear You") platform functions as an intelligent support and communication layer integrated into the broader architecture of the Dealer Web Portal ecosystem. Its primary purpose is to enhance the efficiency and responsiveness of dealer operations by automating the management of feedback, service requests, and informational updates. At the core of the platform is a large language model–based classification engine capable of analyzing incoming dealer e-mails and support messages, identifying their intent, and routing them to the appropriate organizational units [16]. This automated triage significantly reduces manual intervention, shortens response times, and ensures that operational issues are directed to the teams most capable of resolving them.

Beyond its automated routing mechanisms, the platform centralizes all complaint and ticket handling into a unified interface, offering both dealers and internal teams a transparent and consistent means of tracking issue resolution. It also serves as a communication hub by delivering critical announcements, commission updates, and campaign recommendations tailored to dealer behavior patterns identified by the segmentation engine. Moreover, *Sesini Duyuyoruz* integrates several operational tools—including dealer-level customer registration workflows and POS activation screens—that streamline daily processes and reduce friction in service delivery. In combination, these features position the platform

3. AI-Driven Segmentation and Behavioural Analytics

The analytical intelligence layer of the system was designed to convert large-scale transactional data into actionable insights that could improve dealer performance, optimize commission strategies, and support operational decision-making across the organization. As the integration of the Dealer Web Portal and Android POS significantly expanded the volume and granularity of available payment data, the project required a machine-learning framework capable of identifying latent behavioural patterns without relying on manually defined categories. To address this need, an unsupervised segmentation engine was developed to model dealer-level payment behaviour and uncover distinct clusters that reflect meaningful operational characteristics [14,15].

The segmentation pipeline with the consolidation of dealer payment histories, capturing variables such as transaction counts, aggregated bill types, temporal activity profiles, seasonal spending trends, and the presence of high-value or irregular payments. This heterogeneous dataset is standardized through a multi-stage preprocessing workflow consisting of feature normalization, outlier detection, and extraction of statistical descriptors that summarize dealer activity. These steps ensure that the clustering algorithms operate on a stable and interpretable representation of dealer behaviour, making the subsequent segmentation process both robust and scalable.

A range of clustering methods [3, 4, 5, 6, and 7] was evaluated to determine the most appropriate modelling approach for the data. Hierarchical, density-based, and graph-based algorithms were tested alongside centroid-based models. Although several methods provided partially coherent groupings, K-Means delivered the most stable and interpretable clusters when assessed using Purity and Normalized Mutual Information metrics. These clusters aligned strongly with practical usage patterns observed in the field, typically separating high-volume dealers from low-frequency outlets, identifying dealers with seasonally concentrated payments, and isolating those handling large, sporadic bill transactions. By mapping dealers into these operationally meaningful categories, the segmentation engine enabled targeted commission adjustments, promotional strategies, and differentiated support interventions.

Beyond its immediate analytical contributions, the segmentation module provides foundational intelligence for other components of the system, including dynamic recommendation mechanisms and dealer-specific communication workflows. Its outputs are integrated into the Sesini Duyuyoruz platform to tailor announcements, offer personalized campaign suggestions, and identify dealers who may require training or outreach. Collectively, this AI-driven analytical layer enhances the system's ability to adapt to evolving usage patterns, elevates dealer engagement, and supports data-informed digital transformation across the payment ecosystem.

4. System Evaluation and Results

The proposed payment architecture was evaluated using operational data collected from production environments between 2023 and 2025. This period corresponds to the full deployment lifecycle of the system, beginning with the introduction of credit card payments through the Android POS and extending to the maturation of dealer adoption and large-scale usage. The evaluation focused on three principal dimensions: transactional performance, financial impact, and system reliability.

From a transactional perspective, the integration of POS-based credit card processing into the Dealer Web Portal produced a marked shift in payment behaviour among dealers. Prior to deployment, bill payments conducted through the portal relied exclusively on cash, limiting throughput and preventing customers from utilizing alternative payment methods. After the new architecture was introduced, credit card transactions increased rapidly, growing by approximately 35-fold between 2023 and 2025. This expansion demonstrates not only the viability of the technical integration but also the strong uptake by dealers and customers once a more flexible payment method became available. The corresponding payment volume grew by approximately 190-fold over the same period, indicating that the system succeeded in capturing a substantially larger segment of customer demand for bill-payment services.

Financially, the impact of the deployment was equally significant. Commission revenue increased by approximately 7-fold between 2023 and 2025. This growth reflects a combination of higher transaction counts, larger average payment values, and improved dealer engagement supported by the segmentation engine described in Section 3. Dealers identified as high-potential segments contributed disproportionately to the revenue increase, validating the use of AI-driven behavioural insights to inform promotional strategies and commission schemes.

System reliability was assessed by monitoring real-time performance indicators during peak operational periods, including month-end bill surges and periods with high density of POS activity. Throughout the observation window, the backend microservices maintained stable low-latency operation, and the callback mechanism from POS devices exhibited strong consistency with negligible failure rates. The end-to-end payment cycle—spanning the initial portal request, backend processing, POS interaction, and final callback—remained within acceptable response-time thresholds, even under elevated workloads. No critical availability incidents were recorded during this period, demonstrating that the distributed architecture was capable of sustaining high-volume financial operations while preserving transactional integrity.

When combined with the behavioural segmentation outputs, the evaluation results offer a comprehensive picture of system performance. Not only did the architecture deliver measurable operational gains, but it also enabled new business capabilities—such as dynamic commission optimization, targeted dealer interventions, and predictive workload planning—that would not have been feasible using traditional reporting methods. These results confirm the effectiveness of integrating machine learning-driven analytics within a modernized payment infrastructure and highlight the broader potential of AI-supported digital transformation in large-scale financial service ecosystems.

5. Discussion

The deployment and evaluation of the proposed payment architecture highlight several important implications for digital payment ecosystems, dealer operations, and the integration of AI-driven analytics within large-scale financial infrastructures. The substantial increase in both transaction volume and monetary throughput demonstrates that dealers readily adopt alternative payment methods when the underlying system minimizes operational complexity and reduces friction for both employees and customers. The seamless interaction between the Dealer Web Portal, backend microservices, and the Android POS suggests that integrating heterogeneous components—each designed for different layers of the payment stack—can produce a coherent and high-performing end-to-end workflow when supported by a well-structured architectural model.

A key observation emerging from the study is the significance of behavioural heterogeneity among dealers. The segmentation results clearly indicate that dealers exhibit vastly different operational patterns, influenced by factors such as customer demographics, regional bill-payment habits, and the business focus of the dealer itself [14]. Such diversity underscores the limitations of uniform commission schemes or one-size-fits-all incentive models. By providing a data-driven method to identify and characterize these groups, the AI-based segmentation engine creates opportunities for targeted interventions that improve both dealer performance and customer experience. Importantly, these interventions need not be limited to commission optimization; they can extend to training, campaign design, platform configuration, and localized promotional strategies.

The integration of the *Sesini Duyuyoruz* AI support platform further demonstrates that machine learning can play a central role beyond analytical modelling. By automating message classification and routing, the platform reduces operational overhead, standardizes response quality, and enhances the service responsiveness experienced by dealers. This is particularly relevant in environments where large dealer networks generate substantial volumes of feedback and support requests. The results indicate that LLM-driven support systems are capable of augmenting—not just automating—existing workflows by offering more consistent, timely, and contextually aware responses.

Despite strong performance, several challenges and considerations emerged during the deployment. Although the POS callback mechanism proved reliable, its dependency on external network quality means that sporadic latencies or environmental disruptions remain difficult to eliminate entirely. Additionally, the evolving nature of dealer behaviour suggests that segmentation models may require periodic retraining to remain representative as patterns shift over time. Care must also be taken to ensure that the AI components are monitored for potential drift, misclassification, or unintended bias—particularly as dealer support and communication channels grow in scale and complexity.

Overall, the results affirm that the fusion of a modernized payment architecture with AI-powered analytical and operational tools can produce a high-impact transformation in bill-payment ecosystems. The findings also highlight the need for continuous monitoring, incremental enhancement, and strategic governance to sustain long-term scalability and reliability. These considerations set the stage for future improvements, particularly in the areas of predictive modelling, anomaly detection, and reinforcement-learning-based decision optimization, as discussed in the concluding section.

6. Conclusion and Future Work

This study presented the design, implementation, and evaluation of a fully integrated credit card payment architecture for the Dealer Web Portal, combining a microservice-based backend with seamless Android POS integration and AI-driven analytical components. The results demonstrate that the proposed system successfully expands the operational capabilities of the existing dealer infrastructure by enabling secure card transactions, increasing payment flexibility for customers, and significantly improving transaction throughput and revenue generation. The architecture's stability under real-world production workloads further confirms the effectiveness of the distributed design and its ability to sustain high-volume financial operations while maintaining transactional integrity and responsiveness.

A central contribution of the work lies in the integration of machine learning–driven segmentation and the *Sesini Duyuyoruz* support platform into the payment ecosystem. These components illustrate how AI can enhance dealer operations not only through behavioural analytics but also through automated feedback management, personalized communication, and data-informed decision-making [17]. The positive operational outcomes observed during deployment suggest that AI-supported digital transformation can deliver substantial performance gains when tightly coupled with core transactional workflows.

Despite these achievements, several avenues for future work remain. The first concerns the development of more advanced predictive analytics to anticipate dealer activity, customer demand cycles, and potential system bottlenecks. Complementing the segmentation engine with anomaly-detection models could improve fraud detection, identify irregular payment behaviours, and enhance overall system security [16]. Another promising direction is the exploration of reinforcement learning for dynamic commission optimization, enabling the system to adapt commission structures in real time based on dealer performance, regional payment patterns, and evolving business objectives.

From an operational standpoint, future enhancements may include the integration of serverless or event-driven architectures to simplify scaling and reduce infrastructure overhead. Additional improvements to the *Sesini Duyuyoruz* platform—such as multilingual LLM support, context-aware response generation, and tighter integration with dealer-facing dashboards—could further strengthen user engagement and streamline support workflows. Finally, expanding the platform’s capabilities to incorporate customer-level analytics would open the door to new service models, personalized communication strategies, and a deeper understanding of payment behavior at the individual user level.

7. Acknowledge

We extend our appreciation to the Corporate Platforms Technology Solutions team at Paycell for their significant contributions to this study. Generative AI tools were employed exclusively for translation and language refinement. All scientific content, analyses, and conclusions were entirely conceived, validated, and approved by the authors.

References

[1] Stankic, R., Aleksic, J. S., & Jakovljevic, A. (2019). DIGITAL TRANSFORMATION TRENDS IN PAYMENT SYSTEMS. *New Economist/Novi Ekonomist*, (26).

[2] Khando, Khando, M. Sirajul Islam, and Shang Gao. "The emerging technologies of digital payments and associated challenges: a systematic literature review." *Future Internet* 15.1 (2022): 21.

[3] Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* (pp. 226–231). AAAI Press

[4] Ng, A. Y., Jordan, M. I., & Weiss, Y. (2002). On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems (NIPS 14)*, 2001/2002.

[5] Comaniciu, D., & Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(5), 603–619. https://doi.org/10.1109/34.1000236

[6] Frey, B. J., & Dueck, D. (2007). Clustering by passing messages between data points. Science, 315(5814), 972–976. https://doi.org/10.1126/science.1136800

[7] Kahveci, E., & Gurgur, T. (2025). Digital Payments and Sustainable Economic Growth: Transmission Mechanisms and Evidence from an Emerging Economy, Turkey. Journal of Theoretical and Applied Electronic Commerce Research, 20(2), 142. DOI: https://doi.org/10.3390/jtaer20020142

[8] Teker, S., Teker, D., & Orman, I. (2022). Digital Payment Systems: A Future Outlook. PressAcademia Procedia, 15(1), 175–176. DOI: https://doi.org/10.17261/Pressacademia.2022.1613

[9] Dhandapani, A. (2025). Microservices Architecture in Financial Services: Enabling Real-Time Transaction Processing and Enhanced Scalability. European Journal of Computer Science and Information Technology, 13(32), 145–159. DOI: https://doi.org/10.37745/ejcsit.2013/vol13n32145159

[10] Sitek, T., & Kotulski, Z. (2018). POS-originated transaction traces for downtime and risk management in banking and payment card systems. EURASIP Journal on Information Security, 2018(1), 1–15. DOI: https://doi.org/10.1186/s13635-018-0076-9.

[11] Yang, M.-H. (2014). Security Enhanced EMV-Based Mobile Payment Protocol. The Scientific World Journal, 2014, 864571. DOI: https://doi.org/10.1155/2014/864571.

[12] Luo, J.-N., & Yang, M.-H. (2019). EMV-Compatible Offline Mobile Payment Protocol with Mutual Authentication. Sensors, 19(21), 4611. DOI: https://doi.org/10.3390/s19214611.

[13] Nezhad, M. M., et al. (2025). SoK: Security of EMV Contactless Payment Systems. arXiv preprint. arXiv: https://arxiv.org/abs/2504.12812.

[14] Diyabi, N., Çakır, D., Gül, Ö. M., Aytekin, T., & Kadry, S. (2024). Evaluating Customer Segmentation Techniques in the Retail Sector. International Journal of Interactive Multimedia and Artificial Intelligence, 9(3), 175–190. DOI: https://doi.org/10.9781/ijimai.2025.05.001.

[15] An Exploration of Clustering Algorithms for Customer Segmentation in the UK Retail Market. Analytics, 2(4), 809–823. DOI: https://doi.org/10.3390/analytics2040042.

[16] AI-based Classification of Customer Support Tickets: State of the Art and Implementation with AutoML. arXiv preprint, arXiv:2406.01789. arXiv: https://arxiv.org/abs/2406.01789.

[17] TaDaa: Real Time Ticket Assignment Deep Learning Auto Advisor for Customer Support, Help Desk, and Issue Ticketing Systems. arXiv preprint, arXiv:2207.11187. arXiv (PDF): https://arxiv.org/pdf/2207.11187