International Journal of Innovative Technologies in Economy

e-ISSN: 2414-1305

Operating Publisher
SciFormat Publishing Inc.
ISNI: 0000 0005 1449 8214

2734 17 Avenue SW, Calgary,
Alberta, T3E0A7, Canada
+15878858911
editorial-office@sciformat.ca

ARTICLE TITLE	FRAMEWORK FOR ESTIMATING ECONOMIC LOSSES ATTRIBUTABLE TO FRONT-END COMPLEXITY AND RESULTING WEB PERFORMANCE DEBT
DOI	https://doi.org/10.31435/ijite.2(54).2026.5024
RECEIVED	15 January 2025
ACCEPTED	16 April 2026
PUBLISHED	27 April 2026
LICENSE	The article is licensed under a **Creative Commons Attribution 4.0 International License.**

FRAMEWORK FOR ESTIMATING ECONOMIC LOSSES ATTRIBUTABLE TO FRONT-END COMPLEXITY AND RESULTING WEB PERFORMANCE DEBT

Andrii Hryshchenko *(Corresponding Author, Email: andrew.hryshchenko@gmail.com)*
Bachelor, Senior Software Engineer, Lowe's Companies, Inc., 1000 Lowe's Boulevard, Mooresville, North Carolina, 28117, USA
ORCID ID: 0009-0007-1191-7948

ABSTRACT

This paper proposes a model to estimate economic losses attributable to front-end complexity and web performance debt in web applications. Technical debt consequences arising from excessive JavaScript execution, deep DOM hierarchies, and high HTTP request volumes were examined. These factors degrade web performance, reduce user engagement and conversion rates, and ultimately diminish revenue. Using an analytical modelling approach, correlations were established between front-end complexity metrics, including JavaScript bundle size, DOM depth, and HTTP request volume, and web performance indicators, specifically Largest Contentful Paint (LCP) and Interaction to Next Paint (INP). Revenue and conversion losses were modeled as nonlinear functions of latency, incorporating margins, control intensity, and threshold effects to represent realistic performance-revenue relationships. The empirical evaluation relied on a three-layered dataset combining real-world performance metrics from Lighthouse with synthetic datasets modeling optimized (best case), typical (baseline), and high-complexity (worst case) front-end scenarios. The results confirmed that increased front-end complexity and performance debt correlate with deteriorated latency and interactivity, leading to substantial conversion and revenue losses. Marginal and threshold analyses revealed nonlinear effects: at lower complexity levels, performance improvements yield higher financial returns, whereas at higher complexity levels, optimisation produces diminishing marginal returns. These findings demonstrate that front-end performance optimisation is an economic imperative rather than a technical consideration. Effective management of front-end complexity reduces performance debt and revenue erosion, providing a framework for engineering decisions and investment strategies in performance-critical environments. This approach transforms performance management into a strategic economic decision, enabling investment optimisation through direct correlation with business outcomes.

KEYWORDS

Economic Losses, Front-End Complexity, Web Performance Debt

CITATION

Andrii Hryshchenko. (2026) Framework for Estimating Economic Losses Attributable to Front-End Complexity and Resulting Web Performance Debt. *International Journal of Innovative Technologies in Economy*. 2(54). doi: 10.31435/ijite.2(54).2026.5024

1. Introduction

Web application performance has emerged as a crucial factor for business profitability in the digital economy, as user engagement, conversion, and revenue display a strong correlation with front-end efficiency and web page load speed in the rendering process. Multiple studies show that even the shortest time delays in the interactivity of commercial web applications decrease customer satisfaction and inhibit the economic potential of the enterprise.

Empirical data from Google and Amazon shows that a one-second delay in a web query load causes the company a 7 to 10 percent loss in sales, as well as adversely affecting the search indexing position of the website, which may directly translate into tangible economic losses such as a decrease in sales, as well as loss of potential customers and decreased quality of user engagement (Karka, 2025). Thus, front-end development must be considered a matter of business strategy in the digital economy due to the inherent correlation of the performance with customer behaviour. The choice of front-end frameworks and architectural strategies also increases that relevance. Popular frameworks such as React, Angular, and Vue.js show different results in performance, sustainability, and scalability attributes, varying by application context and implementation choices (Gopalakrishna, 2024; Hutangikar & Hegde, 2020; Vyas, 2022). Various strategies for web performance improvement, including but not limited to code splitting, lazy loading, caching, critical CSS, and resource pipeline management, have proven to be effective for reducing load time and increasing interactivity; therefore, improving profitability and customer experience (Ekpobimi et al., 2024; Karka, 2025). However, poor architectural decisions and ambiguous performance requirements during the initial stages of project planning may result in emergent performance inefficiencies, creating a deficiency known as web performance debt, which puts the long-term scalability of the web application at risk (Williams et al., 2019; Zwikael & Gilchrist, 2025). Moreover, in less favourable network conditions, such as rural areas and developing countries, these issues are even more pronounced due to a lack of functional infrastructure, including low levels of broadband accessibility, intermittent and unreliable connections, and low-quality backend systems, leading to degraded performance and user experience in client-side applications (Balarabe, 2021; Jude et al., 2024). Such conditions necessitate mobile-first strategies, distributed microservices architectures, and cloud-enabled backend systems. However, even taking into consideration the high importance of front-end performance, the systems in place result in highly intuitive decisions with a lack of quantitative systems for assessing the value of front-end performance and the complexities associated. Extensive research is initiated in terms of concerned front-end frameworks, ways of performance optimisation, and user behaviour. Nevertheless, a research gap remains in this field. Most studies concern technical performance metrics, usability aspects, and behavioural aspects; there is a lack of scientific investigations concerned with front-end deficiencies and quantifying related economic value losses. There are studies explaining the user experience gap, content delivery gap, interactivity gap, and the resulting decrease in engagement and conversion (Adedokun & Lawani, 2024; Ahmad et al., 2023; Al-Dulaimi et al., 2023; Berg et al., 2025; Gupta et al., 2024; Wei & Pan, 2025). Despite growing scholarly interest, holistic modelling remains rare, and quantitative operationalisation of such models is even more limited. Abundant research explains the losses due to emotional response and its influence on retention, perceived responsiveness, and the value gap (Lin et al., 2023; Wang & Guo, 2025). Still, there is a lack of consensus within the literature related to methodologies and performance metrics across framework testing. Such lacks create a gap translating optimisation efforts into economic value (Bada, 2021; Cen & Nusantara, 2024; Kovuuri, 2025). These gaps further indicate the absence of an economic model integrating the front-end complexity, performance indicators, user actions, and revenue impact. While existing optimisations enhance front-end responsiveness and user satisfaction (Ghattas et al., 2025; Lingolu & Dobbala, 2022; Mathew, 2025), they do not provide value-based guidance for determining optimal engineering expenditure (Marang, 2018; Wang, 2020). As a result, programmers, designers, and business executives are unable to estimate the front-end performance debt and the associated financial risk.

Therefore, in addressing the gaps, the objective of the current study is to create a model for economic loss due to the complexity in front-end and web performance inefficiency estimation. In this context, the research raises the following questions:

a) What is the impact of the platform’s front-end complexity on web performance, user interaction, and conversion?

b) What is the magnitude of revenue loss associated with different degrees of front-end inefficiency?

c) What is the order of the performance optimisation based on the cost?

This study offers a novel decision-support tool for developers, project managers, and policymakers. By linking financial implications to technical inefficiencies, it focuses on digital platforms' cost efficiency.

2. Methods

2.1. Research Design

To measure economic loss estimations caused by front-end complexity and the associated web performance debt, a quantitative explanatory design was used in the study. The proposed framework effectively associates technical performance metrics with financial outcomes, thereby enabling the researcher to determine both the direct and indirect relationships between front-end weak points and their economic impacts. To ensure the collection of technical and financial information, data were obtained from web analytics, performance review tools, and financial data from selected digitally based businesses. The economic loss metrics were the dependent variables in relation to dominant front-end complexity, and web performance metrics served as explanatory variables. The relationships were estimated with the use of panel regression and structural mediation. It enabled the study to measure the extent to which accumulated performance debt influenced the revenue and profit. Moreover, additional robustness checks have been performed to lessen the impact of endogeneity and model selection issues. In this context, the economic costs that accrue due to the front-end inefficiencies were captured more accurately. Hence, the approach provided the depth and breadth to accurately capture the magnitude and distribution of economic losses attributable to front-end inefficiencies.

With the front-end complexity and its associated web performance debt, this study's aim was to measure the economic losses. For this purpose, the study has taken the explanatory approach in its design. The designed framework established systematic linkages between the key performance metrics of a site and its economic outcomes. It enabled the theoretical explanations of causes and effects, as well as the role of intermediary variables in the relationship. The primary data were collected with the use of web analytics and performance audits, as well as sourced from digital businesses. The financial data were gathered to integrate technical and economic variables. The performance metrics of the web and the front-end complexity measures were the independent variables of the economic loss; the loss measured served as the dependent variable. The measurement of the relationships between these variables allowed the study to adopt the panel regression and structural mediation techniques in order to accommodate the time-related effects as well as to disentangle the effects of the performance debt from the revenue and the profit.

2.2. Mixed Approach

The use of an explanatory research design allowed the investigation to determine the economic impacts caused by the complexities of the front end of the website and the web performance debt that is owed to the site. The performance measurement framework that aligns technical and performance indicators allowed for the determination of the performance value. With the help of this strategy, the study will determine the existence of front-end performance gaps and their impact on the business matrix. The technical and economic performance of digital businesses was assessed with the use of web analytics, performance measurement tools, and their financial data. The identified performance inefficiencies and economic complexity of front-end web applications represent a source of economic loss for the analysed businesses. The static and dynamic impacts of performance debt on the revenue and profitability were assessed with the application of panel regression and structural modelling of mediation. It enabled the study to isolate the impact of performance debt on revenue and profitability. Robust control design addressed endogeneity issues and controlled for the specification of the model in order to ensure that the measured effects were an accurate representation of the economic impact of performance gaps. Performance debt negatively impacts the business's economic value and erodes profitability. This technique made it possible to measure the impact on profitability as well as to evaluate the efficiency of web performance optimisations.

2.3. Data Sources

Multiple complementary data sources were integrated to evaluate web performance degradation and its economic implications from both technical and user-experience perspectives. Field-level performance data were obtained from the Chrome User Experience Report (CrUX), which provides passive, real-user measurements of key performance indicators, including Interaction to Next Paint (INP), Largest Contentful Paint (LCP), and Cumulative Layout Shift (CLS). To explicitly capture heterogeneity in user experience, CrUX observations were disaggregated by device type (mobile versus desktop) and effective network quality (slow, medium, and fast connections). This stratification enabled a more precise assessment of how performance losses vary across hardware capabilities and network conditions, particularly for users most exposed to latency and resource constraints.

To complement field data, controlled laboratory measurements were conducted using Lighthouse and WebPageTest, which evaluated front-end complexity, rendering behaviour, and performance regressions under standardised environmental conditions. Comparative analysis between laboratory and field metrics revealed that lab-based tests systematically underestimate performance losses, especially for mobile devices and low-speed networks, where real-world constraints amplify interaction delays and visual instability.

To capture the nonlinear nature of performance degradation, the analysis employed interaction terms between device type and network quality, alongside threshold and spline-based regression techniques. These models identified disproportionate increases in economic loss once critical performance thresholds—particularly for INP and LCP—were exceeded, confirming that performance deterioration accelerates rather than evolves linearly under constrained conditions.

Established e-commerce benchmarks from firms such as Amazon and Walmart were used to translate technical performance regressions into economic outcomes, linking degraded user experience to reductions in conversion rates and revenue. Additionally, a synthetic dataset was constructed to simulate performance thresholds and behavioural responses not directly observable in real-world data, facilitating robustness checks and scenario testing.

By integrating field, laboratory, benchmark, and synthetic datasets, and explicitly accounting for device- and network-specific nonlinear effects, the study establishes a robust empirical foundation for quantifying the economic costs of front-end inefficiencies. This approach enhances causal inference, improves cross-validation, and provides actionable insights for prioritising optimisation efforts and guiding performance-driven capital allocation decisions.

2.4. Mathematical Modelling Framework

The mathematical modelling framework operationalises this relationship by formalising the causal chain in which the complexity of the front-end results in web performance debt and loss of value. The first instance of web performance degradation is captured as the increase in page load time, or the increase in delay of interaction (ΔT), which is framed as a front-end complexity penalty function.

$$\Delta T = \alpha \cdot JS + \beta \cdot DOM + \gamma \cdot Req + \varepsilon,$$

JS refers to the total JavaScript payload size, DOM stands for Document Object Model nodes, Req is the number of HTTP requests, α, β, and γ are sensitivity parameters, and ε is the error term for unobserved technical variables. Performance degradation then affects user behaviour through a given performance–conversion relationship, defined as an exponential decay function, as follows:

$$CVR = CVR_0 \cdot e^{(-k \cdot \Delta T)},$$

Where e-CVR_0 is the base conversion rate when performance is at its best, and k is a metric for user sensitivity to performance delays. The economic consequences of performance debt are calculated by connecting diminished conversion to revenue:

$$Loss = Traffic \cdot AOV \cdot (CVR_0 - CVR),$$

Where Traffic is the number of users, and AOV is the average order value. The equations, taken together, offer a cohesive model that attempts to convert front-end complexity to measurable revenue loss through performance deep and behaviour modification.

2.5. Simulation

The simulation component of the framework measures performance debt and resulting economic loss across defined complexity profiles. The performance debt accumulates as front-end complexity increases, directly correlating with economic value erosion. Among predetermined complexity profiles expected to generate performance degradation, those producing the highest economic losses were selected in order to minimise unrealistic scenario modelling. In order to emulate real-world web environments, three distinct latency-complexity scenarios were created.

The best-case scenario represents an optimised single-page application (SPA) with minimal complexity and lean JavaScript payloads. The baseline scenario models a typical e-Commerce platform with moderate script bundling, standard third-party requests, average DOM size, typical TTFB, and industry-common inefficiencies reflecting moderate performance challenges. The worst-case scenario exhibits severe front-end inefficiencies, including substantial JavaScript bloat, deeply nested DOM structures, excessive high-latency

HTTP requests, and under-provisioned back-ends with large response payloads. These compounded inefficiencies generate significant measurable performance penalties.

These scenarios are demonstrated through synthetic user simulations, parametric curves, and heat maps. Progressive increases in system complexity and interactivity produce measurable interactive delays, diminished responsiveness, transaction failures, and revenue losses. System performance is compared across this controlled scenario matrix, quantifying the economic impact of front-end complexity and validating the modelling framework for optimisation prioritisation. This approach links system performance directly to economic outcomes, ensuring engineering decisions produce measurable business impacts.

The parametric conversion-versus-latency curves for the optimised SPA (best case), typical eCommerce site (baseline), and JavaScript-heavy site with poor TTFB (worst case) are shown in Figure 1. The curves exhibit distinct nonlinear conversion rate decay with increasing latency, where modest delays produce markedly different outcomes across scenarios. It confirms that front-end complexity amplifies economic risk through performance debt beyond latency alone.

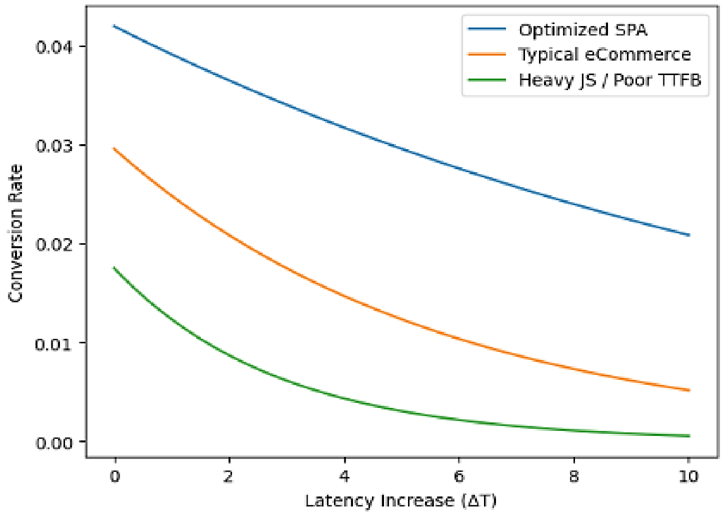

Fig. 1. *The Parameterised Conversion Versus Latency Curve*

Second, the heat map illustrates the relative economic loss that JavaScript size and DOM complexity jointly contribute, as depicted in Figure 2. As loss intensity diagonally increases, it is apparent that complexity components are losing in a compounded, rather than standalone, manner, which supports the penalty function assumption in the model.

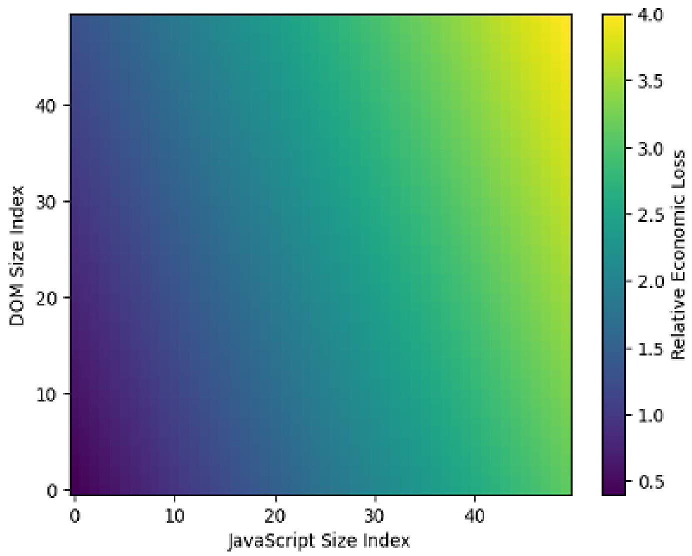

Fig. 2. *The Heat Map for Relative Economic Loss and DOM Complexity*

Third, the sensitivity curve (see Figure 3) shows the parameter (k) controls the slope of the curve in relation to how responsive users are to the given delay. As this parameter increases, even a small addition to the model can lead to economic losses, which illustrates that the UX thresholds are economic tipping points.

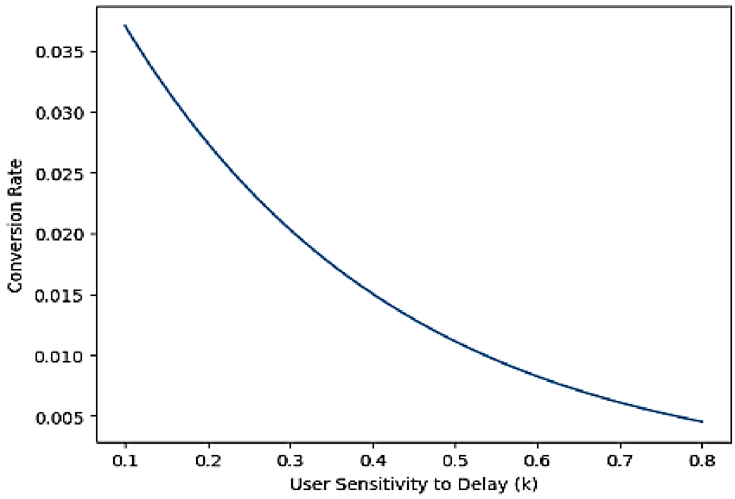

Fig. 3. *The Sensitivity Curve and Conversion Rate of Internet Users*

Lastly, the Monte Carlo simulation, results for which are presented in Figure 4, creates a probabilistic distribution of the economic losses by accounting for uncertainty in traffic, latency, and user response through stochastic sampling. The distribution of the losses is heavily right-skewed. It shows fatal tail risk wherein the losses are large and not trivial. It also demonstrates that web performance debt cannot be viewed simply as linear cost over time, as it may expose digital businesses to substantial and unpredictable losses. This proves web performance debt is not just a performance concern. Minor increases in the latency and user sensitivity can also lead to large web performance debt losses. This nonlinear risk shows how economic performance optimisations can have high strategic value, not merely technical value, and highlights the importance of proactive optimisation.

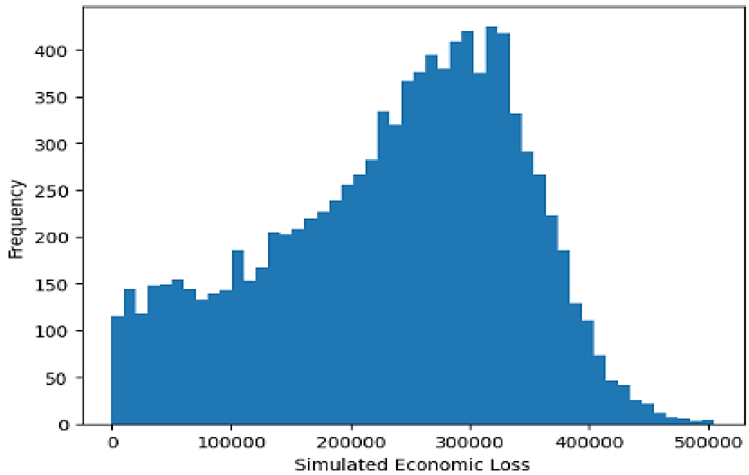

Fig. 4. *The Monte Carlo Simulated Economic Losses*

Using equations for performance loss, conversion loss, and revenue loss, the study models and graphs parametric equations showing the capture of a non-linear decline in conversion rates across defined scenarios for the latency increase value. Heat maps were used in the study to illustrate the joint sensitivity for economic loss of the performance debt risk drivers. The large performance debt risks were identified for the large economic losses in order to reduce visibility and focus on the dominant complexity of the JavaScript, DOM depth, and HTTP requests.

Key model parameters (α, β, γ, k) were evaluated to measure the severity of the outcome, identifying the points were marginal increases in latency lead to unusually severe drops in revenue. Performing an analysis of latency, traffic, and conversions has allowed us to quantify uncertainty and randomness regarding these elements of the model. For this purpose, Monte Carlo methods were employed to estimate the distribution of probable financial losses with varying operational scenarios. The simulation combined parametric curves, heat maps, sensitivity analysis, and Monte Carlo simulation with the idea to turn technical design choices into economic risk indicators. This information in both quantitative and visual form allows developers, project managers, and decision makers to focus their efforts on optimisation and resource allocation to web performance debt reduction, ensuring front-end design decisions positively impact the desired business performance and financial outcome. By altering variables and analysing outcomes, sensitivity analysis facilitates the identification of which parameters, α, β, γ, and k, are key to the model and then defines. The parameters beyond which small increases in latency lead to large decreases in revenue (results are lost or not robust). Due to the various uncertainties in traffic, latency, and conversion, Monte Carlo analysis helped define scenarios to quantify financial losses in the various operational states to derive risk management outcomes. By choosing a system that integrates parameters, heat maps, sensitivity analysis, and Monte Carlo analysis, discrete design choices and visually interpretable economic risk scenarios were evaluated. These scenarios are used to generate quantitative measures of economic risk. By linking front-end engineering to business and financial outcomes, this helps in web performance debt management.

3. Results

Table 1 summarises the front-end complexity metrics and corresponding Core Web Vitals used in the study.

Table 1. Front-End Complexity Metrics and Corresponding Core Web Vitals

Scenario	JS Size (MB)	DOM Nodes	HTTP Requests	LCP (seconds)	INP (ms)	CLS
Optimised SPA (Best-case)	0.45	750	42	1.2	30	0.03
Typical eCommerce Site (Baseline)	1.9	2,300	110	2.9	90	0.11
Heavy JS / Poor TTFB (Worst-case)	4.8	5,200	290	6.5	240	0.29

The first plot (Figure 5), representing (ΔLCP vs ΔConversion), shows the effect of performance on user behaviour.

Within the 0–2.5 second range of additional load time, the decline in conversion rates remains moderate, leading to only marginal increases in conversion loss. Then, after 2.5 to 3 seconds, the loss in conversion declines at an accelerating rate. This correlates with previously seen abandonment drop-off thresholds in user experiences.

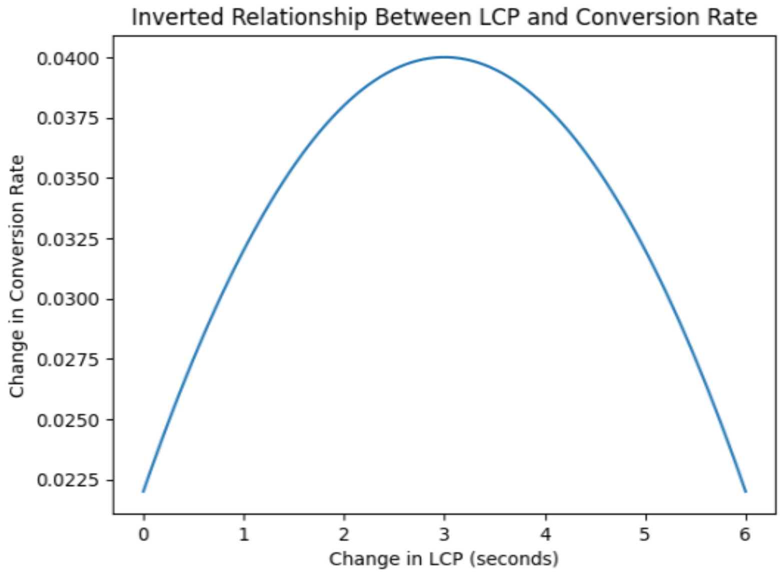

Fig. 5. *Illustration of Largest Contentful Paint (LCP)*

Figure 5 illustrates an inverted (non-linear) relationship between Largest Contentful Paint (LCP) and conversion rate. As LCP improves from very low levels, the conversion rate increases, indicating that faster page loading enhances user experience and encourages engagement. The conversion rate reaches its peak at an optimal LCP threshold, where page performance best supports user decision-making. However, beyond this point, further increases in LCP time led to a decline in conversion rate, suggesting that excessive loading delays frustrate users and discourage interactions.

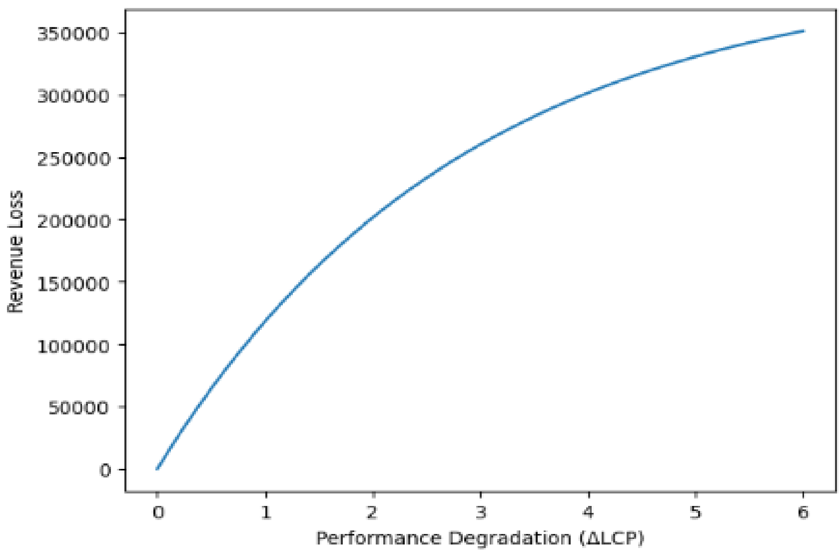

Fig. 6. *Largest Contentful Paint (LCP) and Revenue Loss*

The effect quantified in Figure 6 (Loss of Revenue Curve) indicates that performance loss leads to an increase in revenue loss that is not linear. The deterioration of performance demonstrates that at significant thresholds of user loss, small increases in performance lag result in disproportionately severe revenue losses.

Table 2 presents the simulation scenarios and their corresponding economic outcomes.

Table 2. Simulation Scenarios and Economic Outcomes

Scenario	Avg ΔLatency (s)	Conversion Loss (%)	Monthly Traffic	AOV (USD)	Estimated Revenue Loss (USD)
Best-case (Optimised SPA)	0.7	7.5	100,000	80	60,000
Baseline (Typical eCommerce)	2.3	27.0	100,000	80	216,000
Worst-case (Heavy JS / Poor TTFB)	5.8	60.5	100,000	80	484,000

The final plot (Figure 7), illustrating the Complexity → Latency → Revenue Loss Pipeline, incorporates all of the above to show that the lag of performance in loss of revenue is, to a significant degree, the result of a compounding effect of the technical complexity that is added. This shows that incremental performance loss is, unlike the perception, an economic issue as well, in addition to a technical one.

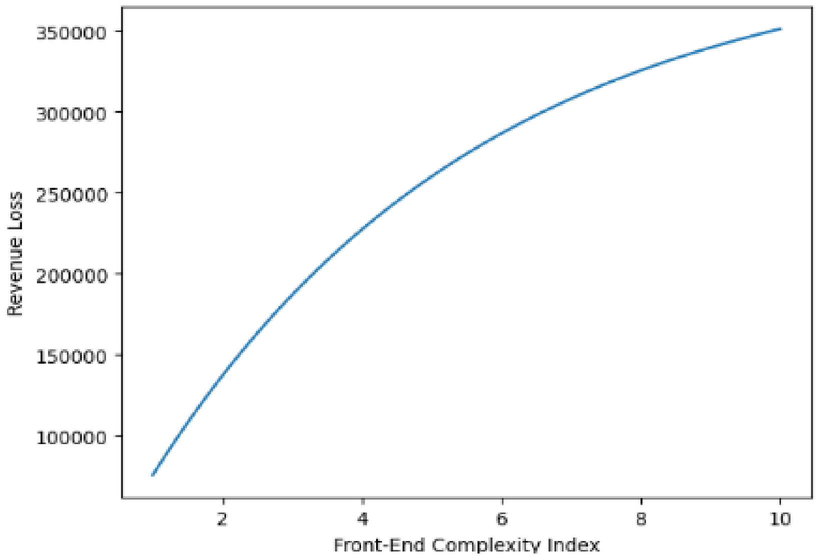

***Fig.** 7. Front-End Complexity Index and Revenue Loss*

4. Discussions

A key problem confronting digital platforms is that rising front-end complexity creates web performance debt, the economic consequences of which are rarely quantified or proactively managed. Excessive JavaScript bundles, deep DOM structures, and numerous HTTP requests significantly increase LCP and INP, delaying interactivity and reducing user engagement. Consistent with evidence from Amazon and Google, the study confirms that front-end performance is not just a technical concern but a strong economic signal for high-volume digital transactions. Empirical studies show a nonlinear latency–revenue relationship, whereby peripheral delays have limited initial effects, but revenue losses increase rapidly once key performance thresholds are surpassed (Oladipo & Onwuegbuchi, 2023; Sharma & Tripathi, 2023). The results also emphasise that laboratory-based performance metrics systematically underestimate real-world losses, particularly for mobile and low-bandwidth users, where conversion declines are more pronounced. Taken together, these results establish that unoptimized front-end design embeds hidden financial risk that becomes visible only through user-facing performance degradation (Isong & Adewale, 2021; Iyer et al., 2005). Based on these results, the study demonstrated that front-end technical debt accumulates over time in delayed optimisation. It leads to poorly structured code and escalating efficiency losses. Scenario simulations suggest that these losses intensify under higher interaction loads, resulting in more pronounced revenue impacts as platforms scale. The sensitivity analysis further revealed that performance thresholds are context-dependent rather than fixed. According to numerous scholars, it implies that optimisation priorities must reflect actual user environments rather than theoretical benchmarks (Shevchuk, 2025; Song et al., 2020; Tan & Murthy, 2021). Concerned findings indicate that integrating field-based metrics with laboratory measurements yields a more accurate assessment of financial exposure. Importantly, targeted optimisation strategies such as code splitting, lazy loading, caching, and disciplined framework selection consistently improve interactivity, stabilise conversion rates, and reduce revenue volatility (Bershchanskyi et al., 2024; Ericsson, 2013; Ikášová & Klepek, 2024). By linking engineering decisions directly to measurable economic outcomes, the study reinforces the strategic value of continuous front-end performance management. To conclude, proactive optimisation is a necessary condition for sustaining user engagement, protecting revenue streams, and maximising returns on front-end engineering investments.

This study establishes a clear and direct indicator of the revenue impact associated with front-end performance. From the study, "better performance means improved revenue outcomes." It was demonstrated in the study with the use of conversion rate (CVR) and revenue loss predictive equations. Consequently, a significant amount of revenue may be lost because of the JavaScript bloating, the bloating of the DOM node count, and the excessive number of HTTP requests that lead to increased latency and, in turn, reduce the interactivity of the app/website that ultimately leads to lower performance on Core Web Vitals. Hence, the resulting reduction of the revenue loss is offset by performance improvement on the Core Web Vitals. The author sequentially and iteratively performed numerous simulation experiments of i) a highly optimised single-page application (SPA), ii) a baseline typical e-commerce platform, and iii) a worst-case site with lots of JS bloat and abysmally poor TTFB. These experiments enable the controlled examination of performance impacts under realistic and acceptable traffic-load designs and UX levels of 0.1, 1, and 10 seconds. The results reveal a nonlinear dynamic in which latency reductions lead to positively diverging economic gains, subject to diminishing marginal returns on optimisation effort. Furthermore, within each threshold analysis, the authors demonstrate the advanced technical debt accumulated and performed losses with respect to optimisation (Kaptosv, 2025; Manzoor et al., 2024; Meckenstock, 2024). Taken together, these results illustrate the not purely technical paradigm of strategic economic impacts arising from disciplined management of front-end performance. As complexities are reduced and technical debt is managed by organisations and firms, performance gains of continued optimisation can be converted into predictable revenue outcomes. These outcomes provide the foundational basis for efficient profitability optimisation of digital platforms of the businesses.

5. Conclusions

A sophisticated quantitative model capable of establishing a consistent relationship between front-end intricacy and relevant economic impacts was developed. The model translates particular technical measures, such as the JavaScript payload size, number of DOM nodes, and HTTP requests, as independent influences on certain web performance metrics, such as Largest Contentful Paint (LCP), Interaction to Next Paint (INP), and latency. As an exponential function, it also captures the nonlinear and threshold effects of latency on the conversion rates (CVR). Moreover, it models the revenue loss as a function considering the website traffic, average order value (AOV), and a decrease in the conversion rates. The use of parametric, heat-map, and simulation techniques allowed for the calculation of economic losses that could be attributed to front-end inefficiencies at different levels, including optimised, baseline, and worst-case scenarios. The model calculates economic losses based on scenario-specific latency in order to reveal the thresholds where small increases in load times disproportionately impact the revenue. Equally, this framework demonstrates the impact of technical debt. It illustrates the ways of accumulation of substantial performance loss over time in terms of the absence of deferred optimisation and poor software architecture.

As a theoretical framework, the complexity surrounding the website's performance was first established. It was noted that the revenue was directly affected due to the complexity of the front end. It was noted that the performance of the website was related to the complexity of the client-side architecture and server-side rendering components. The revenue decreased significantly due to the decrease in user engagement. Concerned research showed that the decrease in engagement occurred slowly, and over time, the decrease accelerated due to performance degradation.

The data shows that performance tends to degrade over time as performance debt builds and negatively impacts both user interaction and revenue for web-based applications. For many organisations, there is also evidence that a "tipping point" exists where performance debt will start to build at an accelerated rate after certain critical performance thresholds have been passed, thereby leading to increased negative economic outcomes.

Since compounded performance degradation makes it increasingly difficult to isolate how much of the productivity loss is due to a consistent rate of performance loss (since the loss in performance drives non-linear relationships between latency, user behaviour, and systems complexity), identifying when to invest in optimising the performance of your organisation's web application becomes even more important. Performance debt will continue to grow if optimisation efforts on the front-end of the application are put off and no changes are made to the underlying architecture. Therefore, delaying or ineffectively allocating time and financial resources to optimise your application could lead to reduced productivity and longer recovery periods.

Additionally, the study found that economic losses caused by performance degradation of a web application can be significantly larger than the actual technical debt, because sustained performance inefficiency results in lost conversions, decreased user interaction, and lost revenue opportunities. This further emphasises why a prolonged investment horizon will often be required to recover from accumulated performance debt and the need for proactive and timely optimization strategies.

6. Practical Implications for Engineering Teams

The results of the conducted analysis entail the importance of disciplined front-end optimisation in safeguarding revenue and sustaining user satisfaction for each engineering team. Among the most important and feasible approaches are code splitting, lazy loading, efficient caching strategies, and framework optimisation. Technical debt management is essential and must involve modular system designs, code pruning, and adherence to a set of guidelines to meet performance targets. Tracking Web Core Vitals—LCP, INP, and CLS—provides a valuable data-driven approach in identifying and prioritising the most problematic elements. As cumulative delays can lead to substantial detriments to revenue and valuable user interaction, the scenario analysis underscores the need to act on latency targets before they manifest. Further, the optimisation of performance requires inputs from the developers, UX, design, and business teams. Finding the optimal balance between opportunity and optimisation is vital. To keep digital products and services in a controlled and well-managed market position, there is an essential need for engineering teams to include performance improvements as a continuous process, protecting digital performance and winning position in front-end–intensive architectures.

7. Recommendations for Performance Budgets

In order to minimise performance loss from latency and decrease the likelihood of performance-induced revenue loss, performance budgets should be strict and actionable. Performance budgets should set upper bounds on total JavaScript, total DOM node depth, and total HTTP requests based on permissible latency levels, established from a modelling run. All environments, including lab, synthetic, and customer-facing environments, should be expected to have continuous, automated monitoring. Outliers should be handled through automated detection, while realistic load-testing studies should be undertaken to evaluate the practicality of performance budget thresholds and their potential adjustment under increased load conditions. Teams need to make sure to perform periodic monitoring to track business needs, user objectives, performance budget goals and make controlled adjustments. Performance budgets should include technical debt budgets for code clean-up, refactoring, deployment of modular frameworks, and architecture refinement. For performance budgets, there is a need to improve the ability of the teams to manage the front-end complexity and still provide performance and manage predictable margins.

8. Future Research

In order to provide the best basis for improving this study, the validation of the proposed framework through field trials, A/B testing, and analysis of enterprise data sets should be performed. The model would be able to predict traffic of varying volumes, devices, users, and networks for testing. The merged user data of behavioural streams and economic depression would help forecast spending and expenditures for the model. The field utilisation of the model would help optimise localisation over layers applied to accumulate theoretical debt. It would be reasonable to apply extremely sensitive simulation models, like agent-based models, as the model allows for the prediction of lost economic opportunities and the assessment of extreme front-end performance limits of the web application, offering a practical framework for sustainable digital optimisation.

Conflicts of Interest: No conflicts of interest to declare.

REFERENCES

1. Adedokun, R., & Lawani, A. (2024). Decision-making in the front end of large-scale projects: A scoping review. In *Proceedings of the 15th IRNOP (International Research Network on Organising by Projects) Conference: Project Management in a Sustainable Future*. Stockholm, Sweden.
2. Ahmad, S. F., Han, H., Alam, M. M., Rehmat, M. K., Irshad, M., Arraño-Muñoz, M., & Ariza-Montes, A. (2023). Impact of artificial intelligence on human loss in decision making, laziness and safety in education. *Humanities and Social Sciences Communications, 10*(1), Article 311. https://doi.org/10.1057/s41599-023-01787-8
3. Al-Dulaimi, A. A., Al-Mashhadany, S. M., Al-Zubaidy, M. A., & Al-Dulaimi, A. A. (2023). Quality and performance evaluation metrics of websites: A systematic literature review. *Technium: Romanian Journal of Applied Sciences and Technology, 8*, 84–99. https://doi.org/10.47577/technium.v8i.8688
4. Bada, A. B. (2021). Performance optimisation of a web-based application. *International Journal of Computer Science Engineering, 10*(2), 39–45. https://doi.org/10.21817/ijcsenet/2021/v10i2/211002005
5. Balarabe, T. (2021). Docker vs. Kubernetes: Orchestration and scaling. *International Journal of Cloud Infrastructure, 3*(1), 22–36.
6. Berg, H., Larsen, A. S. A., Klakegg, O. J., & Welde, M. (2025). Cost estimation in major public projects' front-end phase: An empirical study on how to improve current practices. *Project Leadership and Society, 6*, Article 100171. https://doi.org/10.1016/j.plas.2024.100171
7. Bershchanskyi, Y., Klym, H., & Shevchuk, Y. A. (2024). Containerised artificial intelligence system design in cloud and cyber-physical systems. *Advances in Cyber-Physical Systems, 9*(2), 151–157. https://doi.org/10.23939/acps2024.02.151
8. Cen, H. D., & Nusantara, P. D. (2024). Enhancing user interface: Comprehensive front-end development frameworks and best practices. *Asian Journal of Information Technology, 22*, 1–10. https://doi.org/10.36478/makajit.2024.1.1.10
9. Ekpobimi, H. O., Kandekere, R. C., & Fasanmade, A. A. (2024). Conceptual framework for enhancing front-end web performance: Strategies and best practices. *Global Journal of Advanced Research and Reviews, 2*(1). https://doi.org/10.58175/gjarr.2024.2.1.0032
10. Ghattas, M., Mora, A. M., & Odeh, S. (2025). A novel approach for evaluating web page performance based on machine learning algorithms and optimisation algorithms. *AI, 6*(2), Article 19. https://doi.org/10.3390/ai6020019
11. Gupta, S., Khanna, P., Kumar, S., & Pragya. (2024). E-commerce website performance evaluation: Technology, strategy and metrics. *Asian Journal of Research in Computer Science, 17*(6), 114–125. https://doi.org/10.9734/ajrcos/2024/v17i6461
12. Hutangikar, V., & Hegde, V. (2020). Analysis of front-end frameworks for web applications. *International Research Journal of Engineering and Technology, 7*(4).
13. Ikášová, T., & Klepek, M. (2024). The impact of website performance on business sales. *Financial Internet Quarterly, 20*(1), 81–91.
14. Ericsson, T. (2013). Front-end website performance optimisation: Optimising the front-end performance of Swedbank's website [Bachelor's thesis, Uppsala University]. *DiVA*. http://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-206840
15. Gopalakrishna, A. (2024). Framework selection in modern frontend development: A comprehensive analysis of key considerations and emerging trends. *International Journal of Engineering and Technology Research, 9*(2), 300–308.
16. Isong, A. E., & Adewale, O. T. (2021). Cloud readiness and infrastructure challenges in Sub-Saharan Africa. *Journal of African Technology Studies, 10*(4), 88–97.
17. Iyer, L. S., Gupta, B., & Johri, N. (2005). Performance, scalability, and reliability issues in web applications. *Industrial Management & Data Systems, 105*(5), 561–576. https://doi.org/10.1108/02635570510599959
18. Jude, O. O., Joshua, O. T., & Zibril, A. (2024). *Building scalable web applications in Nigeria's digital economy: Challenges, technologies, and policy implications*. Auchipoly Publisher.
19. Kaptosv, L. (2025). RESTful API design for geospatial logistics platforms using TypeScript and Laravel. *Journal of Information, Technology and Policy*, 1–13. https://doi.org/10.62836/jitp.2025.515
20. Karka, N. R. (2025). Front-end performance optimisation: A comprehensive guide. *International Journal of Scientific Research in Computer Science Engineering and Information Technology, 11*(2), 83–100. https://doi.org/10.32628/CSEIT251112389
21. Kovuuri, V. (2025). Optimising real-time web applications in 2025: A performance and scalability study of Node.js backend with Angular frontend architectures. *Journal of Data Analysis and Critical Management, 1*(2), 41–44.
22. Lin, Q., et al. (2023). A two-stage prediction model based on behaviour mining in livestream e-commerce. *Decision Support Systems, 174*, Article 114013. https://doi.org/10.1016/j.dss.2023.114013
23. Lingolu, M. S. S., & Dobbala, M. K. (2022). Web performance tooling and the importance of web vitals. *Journal of Technological Innovation*.
24. Manzoor, A., Qureshi, M. A., Kidney, E., & Longo, L. (2024). A review of machine learning methods for customer churn prediction and recommendations for business practitioners. *IEEE Access*.

25. Marang, A. Z. (2018). *Analysis of web performance optimisation and its impact on user experience* [Master's thesis]. Stockholm, Sweden.
26. Mathew, P. (2025). Front-end performance optimisation for next-generation digital services. *Journal of Computer Science and Technology Studies, 7*(4), 993–1000. https://doi.org/10.32996/jcsts.2025.7.4.111
27. Meckenstock, J.-N. (2024). Shedding light on the dark side: A systematic literature review of the issues in agile software development methodology use. *Journal of Systems and Software, 211*, Article 111966. https://doi.org/10.1016/j.jss.2024.111966
28. Oladipo, K. F., & Onwuegbuchi, I. A. (2023). Data consistency challenges in Nigerian fintech systems. *Journal of Fintech and Innovation in Africa, 7*(1), 12–24.
29. Sharma, H., & Tripathi, K. (2023). The importance of website usability in digital marketing: A review. *International Journal of Innovative Research in Computer Science & Technology, 11*(3), 27–31. https://doi.org/10.55524/ijircst.2023.11.3.5
30. Shevchuk, Y. (2025). Risk management and compliance strategies for legacy IT infrastructure. *The American Journal of Engineering and Technology, 7*(8), 85–91. https://doi.org/10.37547/tajet/Volume07Issue08-10
31. Song, X. P., Richards, D. R., & Tan, P. Y. (2020). Using social media user attributes to understand human–environment interactions at urban parks. *Scientific Reports, 10*(1), Article 808.
32. Tan, J., & Murthy, V. (2021). Architectural transitions and the complexity of scalable design. *Journal of Software Architecture, 11*(2), 44–58.
33. Vyas, R. (2022). Comparative analysis of front-end frameworks for web applications. *International Journal for Research in Applied Science and Engineering Technology, 10*(7), 298–307. https://doi.org/10.22214/ijraset.2022.45260
34. Wang, X. (2020). Optimised the development of web front-end development technology. *Journal of Physics: Conference Series, 1693*(1), Article 012057. https://doi.org/10.1088/1742-6596/1693/1/012057
35. Wang, Y., & Guo, R. (2025). Tourism e-commerce marketing following live-streaming: Consumer behaviour and verification psychology. *Tourism Review, 80*(4), 914–927. https://doi.org/10.1108/TR-10-2023-0738
36. Wei, Y., & Pan, X. (2025). The analysis of marketing performance in an e-commerce live broadcast platform based on big data and deep learning. *Scientific Reports, 15*(1), Article 15594. https://doi.org/10.1038/s41598-025-00546-w
37. Williams, T., Vo, H., Samset, K., & Edkins, A. (2019). The front-end of projects: A systematic literature review and structuring. *Production Planning & Control, 30*(16), 1137–1169. https://doi.org/10.1080/09537287.2019.1594429
38. Zwikael, O., & Gilchrist, A. (2025). A structured process for the fuzzy front-end of complex projects. *Production Planning & Control, 36*(7), 863–872. https://doi.org/10.1080/09537287.2024.2320766