Journal of the Brazilian Computer Society, 2026, 32:1, doi: 10.5753/jbcs.2026.6572

AI-Driven Software Pricing: An Integrated Approach with Prompt Engineering for Market Analysis

Gregory Fernandes Muniz [**Instituto Federal do Rio Grande do Sul (IFRS)** | ***muniz.gregory1@gmail.com***]
Joelcio de Carvalho Tonera [**Instituto Federal do Rio Grande do Sul (IFRS)** | ***joelciotonera@gmail.com***]
Rodrigo Perozzo Noll [**Instituto Federal do Rio Grande do Sul (IFRS)** | ***rodrigo.noll@ifrs.edu.br***]
Genizia Islabão de Islabão [**Instituto Federal do Rio Grande do Sul (IFRS)** | ***genizia.islabao@gmail.com***]

Programa de Mestrado Profissional em Propriedade Intelectual e Transferência de Tecnologia para a Inovação – Instituto Federal do Rio Grande do Sul (IFRS), Porto Alegre, RS, Brasil.

Received: 15 September 2025 • **Accepted:** 05 February 2026 • **Published:** 04 June 2026

Abstract. Software pricing based on valuation still represents a significant challenge due to its intangibility, variety of business models, and market volatility. This article discusses a pricing protocol by analogy mediated by language models (LLMs) and based on prompt engineering that explores public evidence (sitemaps, functional documentation, and competitor pricing pages). An experimental study was conducted with six software programs applying the same structured prompt in three LLMs, totaling 18 executions with standardized informational scope. The sample software consisted of 5 Innovation Management systems: INTEGRA, HYPE Innovation, IdeaScale, Viima/HYPE Boards, and Qmarkets, and one Customer Relationship Management (CRM) system: Salesforce. The 3 LLMs were: ChatGPT 5.1 Thinking, Gemini 3 Pro, and DeepSeek-V3.2. The LLMs extracted functionalities from sitemaps, mapped competitors, synthesized price benchmarks, and suggested market value ranges. The consolidated orders of magnitude converge, for example, to US$ 8,000–25,000/year in INTEGRA (per-instance license), ∼US$ 1,200–3,600 per user/year in Salesforce (per-seat model), and US$ 50,000–100,000/year in HYPE Innovation (enterprise license), with intermediate levels for IdeaScale (∼US$ 15,000–70,000/year), Viima HYPE Boards (∼US$ 6,000–18,000/year), and Qmarkets (∼US$ 30,000–55,000/year), in line with the functional depth and complexity of integrations observed. As a validation step, the estimates from the three LLMs were compared to actual quotations obtained from reference prices from public sources, after standardization (midpoint when a range existed; periodicity conversion to an annual basis and currency conversions when applicable). The evaluation of the results was done by verifying whether the annual market price was within the range estimated by each LLM (inside/outside the interval) and calculating the quotation (market price ÷ midpoint), as a percentage, as a measure of proximity to the midpoint. Under the interval coverage criterion, ChatGPT showed superiority (5/6), followed by Gemini (4/6) and DeepSeek (2/6), suggesting greater consistency of the first in proposing intervals compatible with the observed prices. Taken together, the results indicate convergence of orders of magnitude, albeit with occasional discrepancies, suggesting that the protocol is more suitable as an exploratory price screening tool, complementary to traditional methods. The main contribution lies in a reproducible and innovative protocol in which, from a single prompt applied in isolated conversations by software and by model, one obtains the functionalities extracted from the sitemap, the competitive benchmarking, the comparative table of functionalities, and the estimation of the market value, enabling a search and price analysis approach based on LLMs.

Keywords: Software pricing; Prompt engineering; LLM; Generative AI; Market analysis; Innovation management.

1 Introduction

Digital transformation has consolidated software as a central component in the strategy of public and private organizations. Solutions that integrate ideation, evaluation, portfolio governance, and project monitoring promise to reduce the time between value discovery and capture, but bring new challenges in selection, implementation, and costing. In particular, the diversity of pricing models (per seat, per instance, enterprise) and the heterogeneity of functional scopes create difficulties in comparisons.

This article investigates sitemaps (generally in XML) published by the vendors themselves, using them as a public source to infer the functional outline of the software from the pages listed on official websites. This allows for a lightweight, reproducible, and auditable comparative mapping. Based on this functional outline, the study also estimates price orders of magnitude through analogy pricing, that is, comparing the product with similar software (of the same category and size) based on public sources and adjusting the benchmark according to differences in functionality and positioning. Thus, a realistic starting point is offered to support acquisition, negotiation, and capacity planning.

Thus, the investigation developed here was guided by the following research question: How can analogy pricing using prompt engineering in LLMs produce software price estimates aligned with market value? Additionally, it explores the extent to which different LLM models, submitted to the same protocol, produce estimates that are equivalent to each other and consistent with real quotations obtained from suppliers. To answer this question, a protocol was produced that combines (i) consolidation of official sitemaps in PDFs, (ii)

functional extraction assisted by LLM with prompt, and (iii) price benchmarking from publicly available materials. The empirical application focuses on six representative and relevant solutions in our analysis contexts: five Innovation Management platforms (INTEGRA, HYPE Innovation, IdeaScale, Viima/HYPE Boards, and Qmarkets) and one for Customer Relationship Management (CRM): Salesforce. They were chosen because they presented high diffusion and technological maturity, complementary functional profiles, public documentation, availability of an official sitemap, market relevance, domain representativeness, diversity of licensing models, and access to public price references, maximizing the empirical representativeness, comparability, and methodological replicability of the study.

Methodologically, this work adopts an applied and descriptive research design, anchored in document collection and LLM-assisted analysis. First, the official URLs of each software's sitemaps are identified, and the complete listings are archived in PDF format with standardized names and minimum metadata (date/time and exact URL). Next, a standardized prompt is applied to three LLMs (ChatGPT 5.1 Thinking, Gemini 3 Pro, and DeepSeek-V3.2) to extract and organize functionalities into comparable categories, maintaining stable inference parameters across all executions. Finally, price benchmarking is performed using public sources (official pages and reference materials), synthesizing orders of magnitude by business model. Human verification and replication with thresholds are used to mitigate inconsistencies.

Scope and limitations of the study. This work is deliberately delimited by: (i) functional extraction restricted exclusively to the textual content of official sitemaps archived in PDF; (ii) price benchmarking based only on public information (official websites and reference materials), without commercial quotations, negotiations or proprietary data; (iii) use of three LLMs (ChatGPT 5.1 Thinking, Gemini 3 Pro and DeepSeek-V3.2) with standardized prompts and isolated chats by software/LLM combination, which reduces, but does not eliminate, variations between executions and risks of hallucination mitigated by human verification and replication with thresholds; (iv) no measurement of deployment costs, professional services, integrations, total cost of ownership, discounts or packages; the reported ranges represent orders of magnitude subject to variations in currency, period, region and segment; and (v) sample selection by convenience of six cases (INTEGRA, HYPE Innovation, IdeaScale, Viima / HYPE Boards, Qmarkets and Salesforce), without claiming to be exhaustive of the market.

The main contributions of the study are: (a) a fast and auditable functional mapping protocol from official sitemaps and public information; (b) a price synthesis of six widely used platforms; (c) practical evidence for short-term decisions (e.g., minimum viable scope, integration planning) and for medium-term strategies (portfolio governance, capability roadmap); (d) reproducible artifacts, PDFs of the sitemaps, prompts, and execution logs that allow scrutiny and replication by third parties; and (e) an exploratory assessment of the convergence between different LLMs and the adherence of the estimated ranges to real market quotations, using two indicators such as range coverage (annual market price within/outside the estimated range) and percentage quotation (market price ÷ midpoint).

The main practical relevance of this article lies in enabling a systematic and reproducible search for prices, linking it to a functional comparison based on public evidence. Using URLs organized in a sitemap and a standardized prompt applied to language models, lists of functionalities, comparative mappings, and market value estimates are obtained. This allows for a transparent and auditable comparison between what is inferred from official pages and the results generated by the LLM. In this way, the proposed protocol offers a practical procedure to support preliminary scope and pricing analyses before formal evaluation and contracting stages.

Section 1 (Introduction) contextualizes the problem of software pricing, explicitly states the study's objective and research questions, and outlines the proposed contribution. Section 2 reviews the theoretical foundations of software cost estimation, LLM, and prompt engineering. Section 3 describes the methodology, including the experimental design and the prompt employed. Section 4 presents the findings, reporting the functionalities and estimated/suggested price ranges for each software, comparing the performance of different LLM models, and highlighting the correlation between functional scope, estimated value, and adherence to actual quotations. Finally, Section 5 (Conclusion) discusses the practical and academic implications, the study's limitations, and directions for future research.

2 Theoretical Framework

This section delves deeper into the discussion of software pricing, exploring traditional methodologies and the growing impact of Artificial Neural Networks on the accuracy of estimates. It then addresses Artificial Intelligence, focusing on Large Language Models (LLMs), their generative capabilities, and inherent limitations. Finally, it discusses Prompt Engineering as a crucial competency for optimizing interaction with LLMs, ensuring the relevance and auditability of outputs.

2.1 Software Pricing

Estimating costs and efforts in software projects is a determining factor in ensuring that deliveries occur within the stipulated time and budget [Rashid *et al*., 2025; Ali *et al*., 2023]. However, even with the vast experience of managers, predicting these costs in advance is a complex task, as it depends on variables such as functional and non-functional scope, requirements, and the technologies that will be used [Rashid *et al*., 2025]. Not surprisingly, inaccurate estimates are among the main causes of failures in software projects [Verner *et al*., 2008]. In this scenario, Artificial Neural Networks (ANNs) have stood out as a solution capable of generating predictions with a fairly reasonable degree of accuracy [Rankovic *et al*., 2021].

Traditional methods, also known as algorithmic or non-language model methods, rely on mathematical equations to calculate costs and efforts [Rashid *et al*., 2025]. Generally, these models use historical data and specific attributes from other projects to make their predictions [Ali *et al*., 2023].

Among them, the Construction Cost Model (COCOMO) is one of the most widely used, considering the size and complexity of the software [Agrawal *et al*., 2016]. However, its first version, COCOMO I from 1981, still has limitations in accuracy and relies excessively on historical data [Saljoughinejad and Khatibi, 2018]. Furthermore, another crucial factor for estimating effort is the quantification of functional complexity, frequently measured using Adjusted Function Points (AFP) [López-Martín, 2015; Hoc *et al*., 2023].

To avoid conceptual ambiguities, this article distinguishes three frequently correlated, but not equivalent, constructs: (i) effort estimation, (ii) development cost estimation, and (iii) pricing. Effort estimation refers to the prediction of the work required to build or evolve software, typically expressed in person-hours/month, and therefore a significant portion of the recent literature directly models effort as a target variable, using statistical and machine learning techniques (e.g., López-Martín 2015; Rankovic *et al*. 2021; Ali *et al*. 2023; Hoc *et al*. 2023), including consolidated evidence in systematic reviews on effort models based on Machine Learning (ML) [Wen *et al*., 2012]. The development cost estimate, in turn, corresponds to the translation of this effort into monetary units, incorporating labor rates, productivity factors, indirect costs, and risk, a topic widely discussed in the literature on software cost estimates [Jørgensen and Shepperd, 2007]. Pricing, on the other hand, corresponds to the determination of a practicable market value range for the commercialization or contracting of a solution, influenced by value perceptions, competitive positioning, billing model, and market conditions, frequently treated as a value-oriented pricing problem in digital services [Harmon *et al*., 2009; Baur *et al*., 2014] and in software license negotiations [Bodendorf *et al*., 2021]. Thus, although the terms 'cost' and 'price' may appear close in the discussion, in this study 'estimate' is linked to the forecast of development effort/cost reported in the literature. Meanwhile, 'pricing' is the methodological focus of the proposed protocol, which seeks an initial market value band through competitive analogy with public evidence.

On the other hand, machine learning (ML) methods employ technologies such as Artificial Neural Networks and fuzzy logic to solve complex problems and, consequently, improve the accuracy of estimates [Qassem and Saleh, 2023]. Compared to traditional models, these approaches are able to learn from historical data to generate more accurate predictions [Rankovic *et al*., 2021]. In this field, techniques such as Case-Based Reasoning (CBR) and Support Vector Regression (SVR) stand out [Wen *et al*., 2012]. More recently, transfer learning in deep learning models has demonstrated performance improvements by reusing pre-trained models [Hoc *et al*., 2023]. However, the strong reliance on historical data remains a limitation [Alauthman *et al*., 2023], and it is important to note that no ML technique can be considered a universal solution for all cases [Villalobos-Arias *et al*., 2020].

At this point, after discussing how Artificial Intelligence has been applied to software cost estimation, it becomes pertinent to move on to an even more recent and disruptive field: generative artificial intelligence. This approach expands the scope of traditional AI by not only predicting but also creating original content in multiple formats, opening up new possibilities for application in different sectors.

2.2 Artificial Intelligence and LLM

Generative artificial intelligence (GenAI) represents an advanced frontier of AI, characterized by its ability to produce original content such as text, digital images, video, audio, or code from specific commands provided by users [De Cremer *et al*., 2023]. This functionality is driven primarily by Large Language Models (LLMs), which employ transformer architectures, deep learning techniques, and natural language processing to predict and generate word sequences [Kietzmann and Park, 2024]. In this sense, such models have transformed productivity in various sectors, expanding AI's scope beyond purely predictive applications [Holmström and Carroll, 2024].

The applications of GenAI are vast and are already manifesting in multiple domains: in healthcare, LLMs are being evaluated for clinical score classification [Zhang *et al*., 2024]; in customer service, tools like ChatGPT offer human-like responses, optimizing costs [Holmström and Carroll, 2024]; in software development, solutions like GitHub Copilot were already being used or planned by more than 70% of developers in 2023 [MacRae, 2023]. Furthermore, GenAI assists in the creation of content on a large scale [Henrickson and Meroño-Peñuela, 2023] and in the production of business rhetoric, mimicking the styles of market leaders [Short and Short, 2023]. This dissemination confirms the ubiquity of AI in contemporary daily life, where virtually all digital services are already permeated by algorithms [Santaella, 2023].

Despite their great potential, LLMs still face significant limitations. Research indicates that these models operate as "black boxes," marked by biases, a lack of transparent auditing, and a strong dependence on intensive material infrastructure [AI Anatomy Map, 2020]. However, limitations regarding hallucinations and reasoning can lead to inconsistent or incorrect responses [Ji *et al*., 2023]. Studies also show that, although structured prompting can elicit improved reasoning, LLMs still struggle with complex tasks and require stronger evaluation and human oversight [Wang *et al*., 2023; Huang and Chang, 2023]. Furthermore, excessive reliance on LLMs can compromise creativity and limit the diversity of ideas, as they reproduce patterns already present in historical data [Dell'Acqua *et al*., 2023].

In this scenario, prompt engineering emerges as an essential competence to guide human-machine interaction, ensuring greater control, precision, and auditability of results. In practical terms, it involves formulating clear and structured instructions to guide models in generating relevant outputs [Robertson *et al*., 2024; Korzynski *et al*., 2023]. Strategies such as chain-of-thought prompting and task decomposition can improve step-by-step reasoning and make problem-solving more transparent [Wang *et al*., 2023; Liu *et al*., 2023]. Thus, the quality of the analyses produced is intrinsically linked to the user's ability to perform effective prompt engineering, an indispensable condition for fully exploiting the potential of these models.

2.3 Prompt Engineering

Prompt engineering has established itself as an emerging and strategic digital competency, defined by the formulation of clear instructions to guide LLM models and extract accurate

and auditable responses [Robertson *et al*., 2024; Fan *et al*., 2024]. Recent surveys synthesize prompting practices and evaluation considerations, offering a consolidated view of effective prompting patterns [Chang *et al*., 2024]. Furthermore, this practice differs from soft prompting, as it does not require retraining the model but focuses on crafting prompts, often anchored in HCI and communicative design [Oppenlaender, 2024]. Not surprisingly, prompt design is thus essential to reduce biases, mitigate misinterpretation, and ensure reliable outcomes [Sundberg and Holmström, 2024].

The applications of prompt engineering are broad and span science, business, entrepreneurship, and art. Research shows that, for example, with few-shot prompting, LLMs reveal emergent skills in tasks that smaller models do not perform [Wei *et al*., 2022]. In-context learning allows a few examples included in the prompt to guide the model toward the desired pattern without updating parameters [Brown *et al*., 2020]. In the business world, LLMs are already being used to build entrepreneurial narratives and enhance creative processes at low cost [Short and Short, 2023; Boussioux *et al*., 2024]. In more technical scenarios, the decomposition of problems into subtasks facilitates the selection of appropriate resources and tools by the models [Qin *et al*., 2024]. Thus, prompt engineering consolidates itself as a link between human creativity and the statistical power of LLMs, increasing efficiency in multiple domains.

Prompt errors can lead to hallucinated outputs, and reliability concerns motivate robust prompting practices and architectures that integrate external knowledge sources for better controllability [Yang *et al*., 2025]. Complementarily, clinical studies show that structured prompts, with explicit reasoning steps, significantly improve the performance of GPT-4 in classifying medical notes [Zhang *et al*., 2023]. Other approaches include prompt-chaining, where the output of one interaction feeds into the next, and role-playing, which places the model in specialized roles, expanding contextual richness [Boussioux *et al*., 2024]. Furthermore, prompt compression methods have been proposed to reduce context length while preserving task-relevant information [Gao *et al*., 2024]. Even jailbreak structures demonstrate how the prompt form can circumvent constraints and substantially alter model outputs [Liu *et al*., 2024].

Despite these advances, challenges persist. Generative models continue to be subject to producing "hallucinations," plausible but incorrect or biased content [Huang and Rust, 2024], and in many cases, function as veritable "black boxes," lacking transparency or auditability [AI Anatomy Map, 2020]. Given this, the need for human control is reinforced, whether through the systematic validation of outputs or through the use of AI training frameworks, where prompt evaluation and optimization can act as substitutes for direct human feedback [Sun *et al*., 2023]. Therefore, the sustainable competitive advantage associated with GenAI depends not only on the adoption of LLMs but, above all, on mastery of prompt engineering, articulated with critical verification and auditing practices [Robertson *et al*., 2024].

3 Methodology

This section details the methodological approach adopted, presenting it sequentially. Initially, the general design and operational objective (Section 3.1) and the tools and environment configuration (Section 3.2) were outlined. Next, the collection of sitemaps and the generation of artifacts (Section 3.3), the isolation of experiments (Section 3.4), and the standardized prompt (Section 3.5) were described. Finally, the completion of metadata (Section 3.6), the criteria for evaluating and replicating results (Section 3.7), and the description of the human verification of the outputs generated by the LLMs (Section 3.8) were discussed.

3.1 General design and operational objective

A 6×3 factorial experiment was conducted with 18 independent runs (one chat per condition), without control groups. The independent variable of the experiment was the combination of software at 6 levels and LLM model at 3 levels. The software in the sample were: INTEGRA (https://integra.ifrs.edu.br/), Salesforce (https://www.salesforce.com/br/?ir=1), HYPE Innovation (https://www.hypeinnovation.com/), IdeaScale (https://ideascale.com/), Viima/HYPE Boards (https://www.viima.com/) and Qmarkets (https://www.qmarkets.net/) and the models were ChatGPT 5.1 Thinking (https://chatgpt.com/), Gemini 3 Pro (https://gemini.google.com/) and DeepSeek-V3.2 (https://chat.deepseek.com/). In this experiment, the dependent variables were the protocol outputs, which include: a list of functionalities inferred from the sitemap, a benchmarking table, and an estimated price range/market value with justifications and sources. The objective was to estimate the market value through competitive analogy, based exclusively on functionalities inferred from the public sitemap of each solution and on the prices/business models of competitors with the same functional focus. An LLM-mediated approach was employed, structured to standardize inputs, outputs, and comparison criteria. A single experimental protocol was adopted, with a prompt developed according to Prompt Engineering principles and isolated environments (one chat per software and per LLM) to ensure comparability and reproducibility. From an operational standpoint, the LLMs were used in two distinct modes throughout the experiments. In the feature extraction phase, the model was instructed to work in "file-only" mode, that is, to consider exclusively the textual content of the PDF sitemap loaded for each software, disregarding its prior training knowledge of the software. The prompt explicitly stated this restriction, emphasizing that functionalities should be inferred only from the structure of menus, categories, and pages present in the artifact, in order to isolate the effect of the chosen data source and reduce the risk of "filling in the gaps" with information learned during training. In the competitive benchmarking and market price survey phase, automated web browsing was deliberately enabled. At this stage, the model was authorized to query the web to (i) identify competing software with comparable functional scope and (ii) locate official pages for plans, licenses, and price ranges. The instructions made it clear that estimates should be based on recent public sources, with explicit citation of URLs and,

whenever possible, indication of the plan type (per user, per instance, enterprise) associated with each value found. Table 1 summarizes the steps of the experimental process adopted, from the definition of the factorial design and the collection of sitemaps to the execution of the protocol in "file-only" mode, the benchmarking phase with web browsing enabled, and the consolidation of outputs for comparative analysis.

Table 1. Stages of the experimental process

Step	Description
1	Definition of the experimental design: 6×3 factorial experiment (6 software programs × 3 LLMs), totaling 18 independent runs, with one chat per condition and no control groups.
2	Sample selection: definition of the 6 software programs (INTEGRA, Salesforce, HYPE Innovation, IdeaScale, Viima/HYPE Boards and Qmarkets) and the 3 models (ChatGPT 5.1 Thinking, Gemini 3 Pro and DeepSeek-V3.2).
3	Collection of primary source data: obtaining the public sitemap for each software and generating the corresponding PDF of the sitemap, keeping the same file as a base for all executions of that software.
4	Standardization of the protocol: development of a single prompt and execution rules to ensure comparability (standardized inputs and outputs) and reproducibility (isolated environment via chat).
5	"File-only" stage (functionality extraction): In each chat, the model received the sitemap PDF and was instructed to infer functionalities exclusively from the file's content, without using prior training knowledge or web browsing.
6	Web-enabled stage (benchmarking and pricing): Next, web browsing was enabled for the model to identify comparable competitors and locate prices/plans in public sources, recording URLs and, when possible, the type of plan associated with the prices.
7	Generation of protocol outputs: for each execution, the following were produced: Highlighted functionalities (from the sitemap); Functional coverage (comparative); Cited competitive benchmark (model and ranges) and Value estimate (range and base).
8	Recording and consolidation: storage of outputs and logs from each chat, and subsequent consolidation of results into comparative charts for analysis between software and between models.

3.2 Tools and environment configuration

The LLM ChatGPT Thinking platform, based on OpenAI's GPT models (LLM family), was used, enabling file upload (PDF) and navigation features for collecting competitor prices and public documentation. Stable inference parameters appropriate for LLMs (without variations between runs) were employed. To control variables, new chats were opened without prior history, the same prompt (Section 3.5) designed by Prompt Engineering was applied to define context, role, steps, and output format, and full transcripts, evidence (links/captures), and generated artifacts were archived.

The ChatGPT Thinking platform from OpenAI was chosen because it combines wide public availability, accessible documentation, and the ability to integrate file reading (PDF) with controlled web browsing, which favors the reproducibility of the study by other researchers. At the time of the experiments, the interface used the GPT-5.1 Thinking model (GPT family, OpenAI), configured with the parameters described in Table 2, maintaining the platform's default exit token limit. Furthermore, when relevant to the experiment's objective (e.g., competitive benchmarking and price range collection), web browsing was explicitly used to search for information from public sources during execution, so that the evidence presented was obtained through queries performed at runtime and traceable in the chat logs. To mitigate the use of unverifiable prior knowledge, the protocol instructed the model not to rely on "memory" or inferences based solely on training, prioritizing exclusively (i) the content of the provided PDF and (ii) information confirmed by web pages consulted at the time of the experiment, whenever applicable. The runs reported in this article were performed in August 2025, in a web environment, with the interface language set to Brazilian Portuguese, and with automatic logs of date, time, chat identification, and model version archived as supplementary methodological evidence.

In addition to ChatGPT Thinking, the web interfaces of Gemini 3 Pro (Google) and DeepSeek-V3.2 were employed, using the same prompt protocol. These were selected because they represent distinct and widely accessible ecosystems, allowing for the evaluation of robustness and inter-model convergence under the same experimental design. Standard/balanced generation parameters and a single sample per run, without prior conversation history, were adopted on these platforms to approximate the behavior between the models and facilitate cross-comparison of estimated price ranges.

Table 2. Inference parameters adopted by the model

LLM	Temperature	top_p	n (samples)	Frequency penalty	Presence penalty
GPT-5.1 Thinking	0.2	1.0	1	0.0	0.0
Gemini 3 Pro	1.0	0.95	1	0.0	0.0
DeepSeek-V3.2	0.7	0.9–0.95	1	0.0	0.0

It should be noted, however, that the web interfaces of ChatGPT, Gemini 3 Pro, and DeepSeek-V3.2 do not expose all the internal details of the inference process, nor do they allow control over some fine aspects of the model, such as the full system prompt, cache policies, or the exact context limits (total number of tokens combining input and output). Thus, the values documented here correspond to the nominal configuration adopted by the authors, that is, to the configurable and observable parameters, but do not eliminate a residual margin of uncertainty associated with the fact that it is a managed cloud service, whose backend may undergo adjustments over time. For this reason, the contribution of this study should be understood as a replicable protocol in terms of flow, inputs, and external parameters of the model, even though it is not possible to completely fix all the internal inference parameters.

To ensure traceability and reproducibility, each experimental session was recorded by: (a) exporting the complete chat transcript, including sections where the tool indicates web searches and lists the consulted links; (b) noting, in a control spreadsheet, the date and time of the session, chat identification, name of the PDF file used, and main URLs actually

Table 3. Methodological procedures

Step	Description
1	The official URLs of the sitemaps for the six software programs were identified.
2	PDFs were generated from each URL (browser → "Print" → "Save as PDF"), fully preserving the route listing.
3	Named standardization was applied: sitemap_INTEGRA_YYYY-MM-DD.pdf, sitemap_SALESFORCE_YYYY-MM-DD.pdf, sitemap_HYPE_YYYY-MM-DD.pdf, sitemap_IDEASCALE_YYYY-MM-DD.pdf, sitemap_VIIMA_YYYY-MM-DD.pdf and sitemap_QMARKETS_YYYY-MM-DD.pdf.
4	Minimum metadata was recorded (capture date/time and exact URL).
5	The extraction of functionalities was restricted exclusively to the textual content of the respective sitemap PDF, as per the explicit instructions in the prompt.

used in price justifications; and (c) storing selected screenshots of the model configuration (name/version displayed in the interface) and the response blocks where the sources are summarized. In the human verification stage, the authors checked, on a sample basis, whether the values and descriptions presented by the model corresponded to the official pages opened in the same session, thus differentiating the use of prior knowledge from the LLM from information obtained by web browsing at runtime.

This procedure does not eliminate the fact that the model maintains a broad training background. It explicitly states when and for what purpose the Web was actually consulted, as well as what evidence was archived so that other researchers can, in the future, replicate the protocol with updated sitemaps and verify if they obtain price ranges and sets of sources comparable to those reported here.

3.3 Sitemap collection and artifact generation

Initially, the official URLs of each software's sitemaps were identified on their public pages, including Rede INTEGRA (https://redeintegra.mec.gov.br/sitemap.xml), Salesforce (https://www.salesforce.com/br/sitemap.xml), and HYPE Innovation (https://www.hypeinnovation.com/sitemap.xml), as well as IdeaScale (https://ideascale.com/post-sitemap.xml), Viima/HYPE Boards (https://www.viima.com/sitemap.xml), and Qmarkets (https://www.qmarkets.net/post-sitemap1.xml).

Next, each PDF was linked to its respective experimental chat, establishing that the extraction of functionalities by the LLM, according to the prompt conceived by Prompt Engineering principles, would occur exclusively on the textual content of the archived sitemap, prohibiting the incorporation of any other sources at this stage. Finally, the PDFs, transcripts, and generated tables were archived together, in order to allow independent verification and repetition of the procedure with the same input and instructions. Table 3 presents the activities performed.

The decision to use sitemaps as the primary source of information about the functionalities of the analyzed software was due to their being public artifacts maintained by the vendors themselves. They summarize the main functionalities, the navigation structure of the product and its modules in terms of menus, categories, and descriptive pages. This characteristic favors the reproducibility of the study, since any researcher can, in principle, retrieve the same sitemaps, convert them to PDF, and reapply the protocol with the same entries, without needing privileged access or in-depth prior knowledge of each solution.

In addition to sitemaps, potential sources of data published by vendors were considered, such as detailed technical documentation, application programming interface (API) guides, knowledge bases ("help centers"), user manuals, and marketing materials (brochures, videos, case studies). While these materials may offer a more granular view of specific capabilities, this data tends to be heterogeneous across vendors (in depth, format, and level of updating) and is not always available in a comparable way for all systems. In many cases, advanced technical documentation requires credentials or is distributed across multiple repositories, which would make it difficult to create a standardized data collection protocol. Strictly promotional materials, in turn, emphasize benefits and use cases, but do not always accurately describe the functional organization of the product.

The use of sitemaps, on the other hand, provides a structured and relatively neutral view of the application, anchored in how the vendor itself organizes the hierarchical structure of the system into sections for the end user. Still, it is recognized that this choice involves risks: (i) underestimation of capabilities, when relevant modules or functionalities do not appear clearly in the sitemap or are hidden in support areas, technical documentation, or internal interfaces; and (ii) overestimation, when the sitemap includes legacy pages, discontinued functionalities, or optional modules that are not offered in all commercial plans. To mitigate these risks, the protocol adopted in this study combined the reading of sitemaps with directed human verification on institutional pages, "product" sections, and pricing pages, discarding pages presented exclusively as institutional (blog, news, job postings, corporate communication) and flagging, in the analysis, cases where functionalities appeared to depend on additional modules or specific licenses.

Even with these precautions, the use of sitemaps as a functional proxy should be understood as an approximation of the capabilities available to the market, and not as an exhaustive inventory of the internal architecture of the systems. In particular, configurable resources, advanced integrations, or associated professional services (implementation, consulting, customizations) tend to appear partially or indirectly in these artifacts, which can impact the value estimate. This limitation is explicitly acknowledged as part of the exploratory design of the research and opens space for future work that systematically combines multiple sources, such as sitemaps, technical documentation, API guides, integration catalogs, and real commercial proposals, to build more complete representations of the functionalities and pricing models of the software analyzed.

3.3.1 Data collection period, experiment duration, and volume of data analyzed

The sitemap collection procedures and the execution of the experiment with LLM were carried out between August 2025 and December 2025. The PDF sitemaps were captured in two time windows, maintaining these same files as a reference for all experimental rounds, in order to avoid variations resulting from subsequent updates to the websites: (i) on August 13, 2025, for INTEGRA, Salesforce, and HYPE Innovation; and (ii) on December 8, 2025, for IdeaScale, Qmarkets, and Viima/HYPE Boards. The metadata recorded for each PDF (capture date and time and exact URL) were used as a basis for temporal traceability of the collections.

Each experiment was defined as an independent interaction session with the LLM used (ChatGPT 5.1 Thinking, Gemini 3 Pro, or DeepSeek-V3.2), consisting of loading a single sitemap PDF, sending the standardized prompt, and generating feature lists, comparative tables, and price estimates. These sessions were conducted in continuous blocks, on a single day using the software, with an approximate duration of 1 minute between sending the initial prompt and obtaining the final version of the results to be verified by the authors. There were no manual interventions to "edit" the model's responses during execution; human intervention focused on reading, checking, and recording the outputs at the end of each session.

In terms of data volume, the sitemaps converted to PDF presented the following order of magnitude at the time of capture: – INTEGRA: 1 PDF page, containing approximately 9 URLs; – Salesforce: 11 PDF pages, with approximately 1,043 URLs; – HYPE Innovation: 268 PDF pages, with approximately 642 URLs; – IdeaScale: 24 PDF pages, containing approximately 997 URLs; – Viima / HYPE Boards: 175 PDF pages, with approximately 308 URLs; – Qmarkets: 10 PDF pages, containing approximately 203 URLs.

In total, the LLMs analyzed 489 sitemap pages, totaling approximately 3,202 URLs of routes and institutional pages. This volume corresponds to the "functional outline" explored in the study, since the extraction of functionalities was deliberately restricted to the textual content of the archived sitemap PDFs, as described in Section 3.3, and supplemented by occasional queries to public product and pricing pages only during the benchmarking stages.

This volume corresponds to the "functional outline" explored in the study, since the extraction of functionalities was deliberately restricted to the textual content of the archived sitemap PDFs, as described in Section 3.3, and supplemented by occasional queries to public product and pricing pages only during the benchmarking stages.

3.3.2 Consistency between sitemaps and official documentation

Table 4 presents a comparison between the functionalities inferred from the sitemaps and those described in other public sources for the analyzed software. A consistently high degree of convergence is observed in the central functional categories: portals and platforms focused on innovation management are described, both in the sitemaps and in official documentation and user reviews, as solutions for capturing, evaluating, and monitoring ideas, projects, and portfolios; Salesforce, on the other hand, is repeatedly characterized as a CRM platform that integrates sales, customer service, marketing, and analytics. This convergence suggests that the sitemaps adequately capture the "functional outline" of the products, at least at the macro-domain level, defining domain groupings.

At the same time, it highlights recurring gaps when comparing sitemaps with detailed documentation and usage reports: advanced functionalities (e.g., in-depth analytics, configurable flows, low-code resources), specific integrations, professional services, and back-office routines tend to appear much more clearly on product pages, institutional materials, tutorials, and reviews than in the routes listed in sitemaps. In all cases, therefore, exclusive reliance on sitemaps would imply the risk of underestimating the functional scope and the value effectively delivered. For this reason, in this study, these artifacts are interpreted as a delimited parameter. The pricing results by analogy are discussed taking this limitation into account.

Table 4. Comparison between functionalities inferred via sitemaps and validation sources (documentation/evaluations)

Software	Validation sources (type)	Synthesis of sitemap alignment vs. other sources	Key risks of relying solely on the sitemap
INTEGRA (Integra Portal)	Institutional documentation, official news, and support materials about the Integra Portal and its use in the management of laboratories and innovation environments.	There is a strong alignment regarding Integra's role as a support environment for research, innovation, and management of laboratories/NITs; the sitemap sections dealing with laboratories, projects, and innovation management converge with the institutional descriptions and training materials.	Underrepresentation of details about internal flows (e.g., NIT operations, administrative routines, and indicators), which appear in videos, training, and practical use, but not always as clear routes in the sitemap; risk of underestimating back-office functionalities and administrative modules.
HYPE Innovation	Official product pages (innovation platform, idea management, and portfolio) and reviews on specialized portals.	High alignment across the macro-stages of ideation, selection, portfolio, and implementation, as well as in the innovation management modules highlighted in the sitemap, marketing materials, and reviews. Partial alignment in analytics, scouting, and integration capabilities.	Advanced features (analytics, scouting, deep integrations, consulting services) appear more prominently in product materials and reviews than in the sitemap; risk of missing important parts of the value proposition when looking only at the sitemap.
IdeaScale	Product overview pages and innovation suite; user reviews on review platforms.	High alignment in core functionalities for capturing, analyzing, and prioritizing ideas, campaigns, and integrations with collaborative tools (e.g., Teams, Slack), present in both the sitemap and the documentation and evaluations.	Professional services features, advanced settings, and plan details (e.g., community limits, SLAs) are not very visible in the sitemap, but appear in documentation and reviews; risk of underestimating associated services and configurable capabilities.
Viima / HYPE Boards	Viima product pages (idea management) and HYPE Boards review pages (formerly Viima)	High alignment in the functionalities for collecting and managing ideas, collaboration between users, and support for different use cases, present in the sitemap and emphasized in both documentation and reviews.	Some aspects mentioned by users, such as specific integrations (e.g., Jira), organizational scalability, and governance details of the innovation process, are not very evident in the sitemap; risk of not capturing all the flexibility and the ecosystem of integrations.
Qmarkets	Official product pages (feature set and overview) and reviews on software portals.	High alignment across major functional categories, such as customizable workflows, multiple workspaces, portfolio management, and evaluation tools, which appear both in the sitemap and on feature pages and reviews.	The sitemap tends to simplify the granularity of advanced features (analytics, APIs, integrations, customer success services) documented in detail on product pages and cited in reviews; risk reducing the platform to "idea management," ignoring broader modules of corporate innovation.
Salesforce (Sales Cloud / CRM Platform)	Official product pages (Sales Cloud and Salesforce platform) and reviews on review portals.	High alignment in core CRM modules (sales, customer service, marketing, analytics, and basic integrations), present in the sitemap sections and official descriptions and reviews. Partial alignment in more advanced platform features, automation, and low-code. (G2)	A wide range of add-ons, specialized clouds, low-code functionalities, and professional services is underrepresented in the sitemap; risk of underestimating the true scope of the solution and the impact of additional modules on pricing by analogy.

3.4 Isolation of experimental variables

Independent chat variables were created for each combination of software and LLM model, ensuring that the eighteen executions (6 software programs × 3 LLMs) remained isolated from each other. In each chat, only the PDF corresponding to the target software was loaded, avoiding context contamination between executions and preserving the state independence of the LLM. It was ensured that inferences in OpenAI's GPT models and the other LLMs used were based only on the respective sitemap and the standardized prompt.

3.5 Prompt

In each chat, the same prompt was applied, derived from Prompt Engineering principles (role definition, informational scope, verifiable reasoning steps, output format, and source usage restrictions), varying only SOFTWARE_NAME, COMPARISON_FOCUS, and SITEMAP_URL. The prompt was sent only once at the beginning of each session, instructing the LLM in OpenAI's GPT models to produce traceable lists, tables, and justifications.

This same basic prompt was used, without any alteration, on all three LLM platforms (ChatGPT 5.1 Thinking, Gemini 3 Pro, and DeepSeek-V3.2). Therefore, any differences in the suggested price ranges were mainly attributed to the characteristics of each model, and not to variations in the instructions provided. PROMPT: Estimate the market value of the software {{SOFTWARE_NAME}}, positioning it and comparing it directly with other solutions and platforms available on the market, focusing on {{COMPARISON_FOCUS}}. Context: The AI will act as a product analyst, tasked with evaluating the functionalities of the {{SOFTWARE_NAME}} software (publicly available through the sitemap) and performing a competitive benchmark. The ultimate goal is to determine an estimated market value for {{SOFTWARE_NAME}}, based on the prices and business models of competing software specializing in {{COMPARISON_FOCUS}}. Paths of AI (Steps to be Taken): Analysis and Extraction of {{SOFTWARE_NAME}} Features: Action: Analyze the sitemap ({{URL_SITEMAP}}) and specifically identify the modules and functionalities that {{SOFTWARE_NAME}} offers. Examples to look for: (list examples of features relevant to the focus of the comparison). Output: Create a clear and structured list of the features offered by {{SOFTWARE_NAME}}, serving as a basis for comparison. Market Research Focused on Software of {{COMPARISON_FOCUS}}: Action: Conduct market research to find commercial software (SaaS or licensed) that are direct competitors or analogous to {{SOFTWARE_NAME}} in the {{COMPARISON_FOCUS}} niche. Search terms should be specific: "Software {{FOCO_DA_COMPARACAO}} price" "Platform for {{COMPARISON_FOCUS}} cost" "Competitors {{SOFTWARE_NAME}}" "{{FOCO_DA_COMPARACAO}} software pricing" "Comparison {{FOCUS_OF_COMPARISON}} business" Research Focus: To identify established companies and products in the {{COMPARISON_FOCUS}} market and their pricing models. Comparative Analysis and Price Data Collection: Action: For each {{COMPARISON_FOCUS}} software found, the AI should collect the following information from public sources (official websites, reviews, articles): Software Name and Company Pricing Model (e.g., per user/month, per module, annual plan, etc.) - Values or Price Ranges (any publicly available cost information) Key Features (to validate the comparison with {{SOFTWARE_NAME}}) Output: A benchmarking table comparing {{SOFTWARE_NAME}} with the {{COMPARISON_FOCUS}} software found. Comparative Table of Features: Action: Create a detailed table comparing the identified functionalities of {{SOFTWARE_NAME}} (exclusively from the sitemap) with the functionalities of competing software (from public documentation). The table should highlight the similarities, differences, and strengths/weaknesses of each solution in relation to the functionalities. Output: A clear and organized comparison table of features. Cost Estimation by Market Analogy: Action: Based on the comparative data, formulate a market value estimate for a solution with the characteristics of {{SOFTWARE_NAME}}. - Annual Price Range: determine a likely minimum and maximum value (e.g., "An annual license for a platform with these features would cost between X and Y"). - Justification of the Estimate: clearly explain which competitor(s) served as the basis for the estimate and why the comparison is valid, considering the scope of the functionalities. Objective Delimitation of AI Advancement (What AI Will Deliver): - List of Features of {{SOFTWARE_NAME}}: a detailed list of relevant features identified. - Competitive Benchmarking Table: comparison with software from {{COMPARISON_FOCUS}}, highlighting functionalities, business models, and costs. - Comparative Features Table: comparison between {{SOFTWARE_NAME}} (sitemap only) and competitors (public documentation). - Market Value Estimate: estimated price range, with robust justification based on market analysis. - Research Sources: a list of all links to the sources used to collect competitor data. AI should NOT: - Utilize the Function Point Analysis (FPA) methodology. - Include software that is not primarily focused on {{COMPARISON_FOCUS}}. - Making assumptions about development or infrastructure costs. The focus is strictly on the market value of the final product.

With the aim of aligning the study with open science practices and facilitating its replication by other researchers, a public online repository has been structured on an open access platform (GitHub), hereinafter referred to as the Study Repository, entitled AI-Driven Software Pricing: An Integrated Approach with Prompt Engineering for Market Analysis, whose address is https://github.com/gregfermun/AI-Driven-Software-Pricing-An-Integrated-Approach-with-Prompt-Engineering-for-Market-Analysis. This repository brings together all non-confidential artifacts used or produced in the research, organized as follows: (i) complete prompts used in each case study, with the full text version in Portuguese (and, when applicable, in English); (ii) the PDFs of the sitemaps of the six software programs analyzed, accompanied by collection metadata (date, time, source URL and file hash); (iii) session log spreadsheets with the three LLM models used, containing for each execution the session identifier, corresponding software, date and time, model declared in the interface, configured inference parameters and indication of whether or not web browsing

was used; (iv) raw and consolidated output tables (lists of functionalities, competitor mappings, estimated price ranges and relative validation indicators), in open format (CSV/ODS); and (v) anonymized extracts of the transcripts of interactions with the model, preserving only the sections necessary for the protocol flow to be understood, without exposing any confidential commercial values.

The repository also includes a README file describing how to replicate the experiments, from downloading or updating the sitemaps to uploading the PDFs to the LLM platform, applying the prompts, manually checking the responses, and calculating the relative validation indicators. Additionally, a prompt protocol template has been made available that can be reused by other groups in different contexts, such as for exploratory pricing of other software, favoring not only the strict reproducibility of the results presented here, but also the reuse and adaptation of the approach in new case studies.

3.6 Filling in the metadata

Only the fields SOFTWARE_NAME, COMPARISON_FOCUS, and SITEMAP_URL were filled in, leaving the rest of the prompt unchanged, in order to preserve Prompt Engineering consistency and comparability.

Example applied to INTEGRA and replicated for other software: SOFTWARE_NAME = INTEGRA FOCUS OF COMPARISON = Innovation Management URL_SITEMAP = https://redeintegra.mec.gov.br/sitemap.xml

In other cases, FOCO_DA_COMPARACAO assumed the values “Innovation Management” (INTEGRA, HYPE Innovation, IdeaScale, Viima / HYPE Boards and Qmarkets) or “CRM / Sales” (Salesforce), in accordance with the functional domain described in Section 4.

In the [Chat-INTEGRA] prompt, the three substitutions above were used. For the other chats corresponding to the combinations between the six software programs and the three LLMs, the procedure was repeated, changing only the equivalent metadata (SOFTWARE_NAME, COMPARISON_FOCUS and SITEMAP_URL), keeping the rest of the prompt text identical.

3.7 Evaluation of results

The evaluation was structured around four axes: (i) conformity to the informational scope, verifying the anchoring of each functionality in identifiable sections of the sitemap PDF; (ii) price traceability, requiring a public source and licensing model; (iii) integrity of the outputs prescribed in the Prompt Engineering prompt; and (iv) semantic consistency and absence of hallucination, checking that the LLMs did not introduce elements without public evidence. Official sources were prioritized, and aggregators without a primary reference were rejected. Additionally, a specific axis of quantitative evaluation was defined, focused on comparing the price ranges suggested by the LLMs with the actual quotations obtained from suppliers, in order to quantify the relative deviation between these two references.

As a control, a technical replication of each experiment was performed (new chat, same prompt, and same PDF), comparing the similarity of the feature lists, the stability of the price ranges in terms of order of magnitude, and the coincidence of public sources. Relevant discrepancies were subjected to targeted re-evaluation (re-reading of sitemaps and pricing pages) and, when necessary, adjustments were recorded.

In the quantitative validation stage against market references, an annual market price (US$/year) was recorded for each software, along with a URL and access date. The price ranges suggested by the LLMs and the reference values were converted to a common basis (monthly to annual and, when applicable, currency conversion), maintaining the same monetary unit and time horizon. When the LLM output was a range, the midpoint (PM = (minimum + maximum)/2) was calculated. For each software/AI pair, two main indicators were then obtained: (a) the “within range” indicator, which verifies whether the market price (P_market) is contained within the estimated [minimum, maximum] interval; and (b) the “quotation” (P_market/PM $\times$ 100), which expresses, as a percentage, how much the market price represents in relation to the midpoint of the estimated range. Values close to 100% indicate greater alignment with the center of the range; Values below 100% suggest a midpoint above the market, and values above 100% suggest a midpoint below the market.

3.8 Human verification of results

The term “human verification” used throughout the article refers to a systematic step of manual checking carried out by the authors themselves, both with prior experience in innovation management, information systems analysis, and software evaluation. For each execution of the prompt protocol on the LLMs used, one of the authors assumed the role of primary reviewer and the other acted as secondary reviewer. The verification involved, firstly, checking whether the functionalities described by the model were actually present in the sitemaps and institutional pages of the software, preventing the model from extrapolating to undocumented functionalities. Next, the lists of competitors and the suggested price ranges were reviewed, checking the links, values, and licensing models in public sources (official pages, commercial materials, or price repositories).

Whenever human verification identified significant discrepancies, such as features not found in official documentation, clearly outdated prices, or a lack of links to verify the values, the case was recorded in a control spreadsheet, and the corresponding round was either discarded or repeated with minor adjustments to the prompt to reinforce instructions for the exclusive use of official sources and the mandatory listing of reference URLs. In these cases, only the outputs where both authors agreed on the adherence between the LLM description, the consulted public sources, and the proposed methodological protocol were retained in the study. However, a formal statistical calibration step between reviewers (e.g., calculation of agreement coefficients) was not conducted, which is recognized as both a limitation and an opportunity for future work, where a more granular and measurable human validation process for the results produced by LLMs could be structured.

4 Results and Discussion

This section presents the results obtained from applying the analogy pricing protocol to the six software programs analyzed, organizing the discussion into two complementary fronts. In Subsection 4.1, the findings are consolidated into a comparative synthesis, bringing together, in a standardized way, the inferred functional outline of the sitemaps, the comparative functional coverage, the competitive benchmarking, and the annual ranges estimated by the three LLMs, allowing for discussion of convergences and sources of variation between models. In Subsection 4.2, a quantitative evaluation of the adherence of the estimates to the market is carried out, comparing the ranges produced by the LLMs with annual reference prices and using simple performance indicators (range coverage and price relative to the midpoint), in order to qualify consistency, proximity, and direction of the deviation between the estimate and the observed price.

4.1 Comparative Synthesis

The results were consolidated in Table 5, which brings together, in an integrated view, analytical dimensions common to the six software programs evaluated: (i) the functional core inferred from the sitemaps, (ii) the comparative functional coverage, (iii) the competitive benchmarking (competitors and pricing models), (iv) the annual estimates in US$ generated by each LLM, (v) the degree of convergence between the models, and (vi) the main findings and sources of uncertainty by platform. This consolidation allows for a comparable interpretation of both the alignment patterns between the solutions and the differences introduced by variations in the assumed scope and billing unit, preparing the synthetic reading discussed in the following paragraph.

Table 5. Comparative summary (sitemap, coverage, benchmarking and annual estimates in US$)

Dimension	INTEGRA	Salesforce	HYPE Innovation	IdeaScale	Viima / HYPE Boards	Qmarkets
Functional core (sitemap)	Portal/hub; directories and catalogs (people, environments, laboratories, services, technologies); partnerships; institutional/FAQ.	Sales Cloud; Service Cloud; Analytics/Tableau; AI/Einstein; Integration (MuleSoft/Data Cloud); Industry Clouds (some outputs include Marketing/Commerce and/or Agentforce).	Idea management; CIP/continuous improvement; scouting/trends/foresight; open innovation/partnerships; governance/KPIs; integrations; culture/gamification (premium modules: boards, hackathons, AI and ISO).	Campaigns/challenges; idea gathering/communities; selection/screening; crowdsourcing/evaluation; whiteboards; templates/frameworks (project planning/portfolio appear in some outputs).	Boards/processes; capture and collaboration; evaluation/prioritization; visualization (Kanban/bubble); analytics; gamification; SSO/governance; integrations (MS 365/Teams, Jira, Power BI, API/REST); external crowdsourcing; mobile.	Ideation; continuous improvement; open innovation/crowdsourcing; scouting (tech/startups); trends/foresight; integrations; KPIs/governance; AI applied to innovation; portfolio/pipeline/roadmaps
Functional coverage (comparative)	Highlighted: directories/catalogs, technology/asset showcase, partnerships/matchmaking, and FAQ. Ideation, funnel/workflows, portfolio/projects, and analytics appear as likely/limited (not explicitly stated in the sitemap).	Convergent reading of "suite CRM enterprise" with functional amplitude and integrations; variation by tiers (SMB → enterprise core → premium with advanced modules).	Coverage of the "enterprise suite": ideas, portfolio, open innovation, trends, gamification, dashboards, automation, and APIs. Key differentiators: Innovation Partnerships, HYPE Boards/Hackathons, and ISO + native AI.	It adequately covers core ideation/campaigns and pipeline; analytics/ROI as partial/indicative data; and confirmed integrations/APIs/SSO outside the sitemap.	High coverage in core items (capture, templates, scoring, workflow, analytics, gamification, integrations, SSO, crowdsourcing). Divergence: automation/workflow not evidenced in Gemini.	Comprehensive suite: covering ideas, portfolio, CIP, scouting, trends, engagement, hackathons, integrations, analytics/KPIs, and AI.
Competitive benchmark (top competitors and model)	Anchors cited (SMB→enterprise): Viima, Accept, Planbox; suites enterprise: Qmarkets, HYPE, Brightidea; and, in one output, Wellspring/Skipso (TTO/ecosystem profile).	Recurring competitors: HubSpot, Microsoft Dynamics 365, Zoho, Pipedrive, Freshsales (mid-market) and Oracle CX / SAP CX / Adobe Experience Cloud (enterprise). Typical model: price per user/month + modules.	Top-tier providers: Qmarkets, Brightidea, Planview Spigit. Alternatives: IdeaScale, Planbox, Wazoku, InnovationCast, Sideways 6, Innovation Cloud. Models: annual license and/or per user.	Competitors: Brightidea, Planview Spigit, Planbox, Qmarkets, HYPE, Viima/HYPE Boards, Wazoku. Models: annual license and/or per user.	Competitors: IdeaScale, Qmarkets, Wazoku, Brightidea, Spigit, Sideways 6, Ideawake (in addition to Ideanote/Planbox). Models: per user and annual license.	Competitors: HYPE Boards/Viima, IdeaScale, Brightidea, Wazoku, ITONICS. Model: annual license (modular) + setup; prices in US$.
Annual estimate – ChatGPT 5.1 Thinking	US$ 8,000–25,000/year (alternative: US$ 20,000–60,000+/year if including ideation/portfolio/analytics).	US$1,200–3,600 per user/year (US$100–300/user/month).	US$ 50,000–100,000/year (annual license).	Approximately US$20,000–50,000/year (there are references to lower ranges and reports up to approximately US$72,000/year).	US$ 6,000–18,000/year (≈100 users) and US$ 50,000–75,000/year (≈1000 users).	US$ 33,000–55,000/year.
Annual Estimation – DeepSeek-V3.2	US$ 20,000–70,000/year.	US$ 2,400–7,200 per user/year (US$ 200–600/user/month).	US$360–600 per user/year (US$30–50/user/month) and an example total of US$36,000–60,000/year (100 users).	US$180–400 per user/year (US$15–33/user/month) — total depends on the number of users/innovators.	US$ 14,000–70,000/year (derived from tiers per user/month and volume calculation).	$30,000–55,000/year (modular).
Annual estimate – Gemini 3 Pro	US$ 130,000–280,000/year.	3 tiers (per user): (A) US$ 300/year (US$ 25/month) • (B) US$ 1,260–1,980/year (US$ 105–165/month) • (C) US$ 3,600–6,000/year (US$ 300–500/month).	US$ 50,000–90,000/year (annual license; ceiling increases with premium modules).	∼US$ 15,000–25,000/year (annual license).	US$ 6,000–15,000/year.	US$ 30,000–55,000/year (there is a reading of larger enterprise contracts on an exit).
Convergence between LLMs (reading)	Convergence in the "ecosystem/TTO profile" (catalogs + showcase + partnerships). The annual range varies mainly due to the hypothesis of scope expansion (ideation/portfolio/analytics).	Convergence in Sales/Service + Analytics + AI + Integration. Minor divergences in Marketing/Commerce and Agentforce; all reinforce the logic of tiers and modularity.	Convergence to tens of thousands of US$/year; premium modules (AI, ISO, hackathons, boards, partnerships) raise the price range.	Strong convergence in the core (campaigns/collection/selection/crowdsourcing + whiteboards/templates). Divergence: pipeline/portfolio (ChatGPT/DeepSeek) vs. more "ideation" focus (Gemini).	Convergence for order US$ 6,000–18,000/year (≈100 users), with possibility to scale by volume; focus on collaboration/visualization/analytics + integrations.	Convergence to enterprise level (∼US$ 30,000–55,000/year) per "multi-module suite" (portfolio + trend/scouting).
Short note (1–2 findings)	The greatest uncertainty stems from what is not explicitly stated in the sitemap (funnels/analytics). Same core, but price range depends on "how much of it is a suite" versus "hub ecosystem".	Integrations (MuleSoft/Data Cloud) appear as a cross-cutting differentiator. Price differences stem from included modules and tier (SMB vs. premium).	Innovation Partnerships is a rare differentiator and leans towards the premium market. The main difference is the emphasis on premium modules and the model (annual license vs. per user).	The per-user model changes the annual total depending on the scale. Direct annual estimates reinforce the mid-market/enterprise positioning.	Key point of divergence: evidence (or lack thereof) of automation/workflow in the sitemap. Values typically below enterprise suites, but increase with scale.	Specialized modules (portfolio + trend/scouting) support the enterprise level; variation lies in modularity and the scale of the contract assumed.

4.2 Evaluation of prices estimated by LLMs

To evaluate the results of the LLMs, Tables 6 and 7 consolidate, for six platforms, the comparison between the annual market price (US$/year) and the price ranges estimated by three LLMs. For each LLM, the estimated range, the midpoint (minimum + maximum of the estimated range divided by 2) as a measure of the range's central tendency, and two market adherence indicators are shown: the estimate of whether the market price is contained within the ranges estimated by the LLMs and the quotation (market price ÷ midpoint), which measures the market's proximity to the midpoints of the price ranges estimated by the LLMs (values close to 100% indicate greater alignment with the midpoint; values below 100% suggest a midpoint above the market; values above 100% suggest a midpoint below the market).

Under the criterion of interval coverage (market price contained within the range estimated by the LLM), ChatGPT shows superiority, covering 5 out of 6 cases (83.3%), followed by Gemini with 4 out of 6 (66.7%) and DeepSeek with 2 out of 6 (33.3%). In substantive terms, this pattern indicates a greater ability of ChatGPT to propose intervals compatible with the reality of prices observed for the analyzed set (n=6), while DeepSeek presents a higher frequency of intervals that do not cover the market. Convergence between models is also identified in cases where all LLMs cover the market, as well as cases where coverage is achieved by only one model, highlighting heterogeneity by platform and suggesting sensitivity of performance to the pricing structure and software domain.

In summary, considering both the coverage rate and proximity to the midpoint, ChatGPT stands out as the most accurate LLM in the examined set, with Gemini showing intermediate performance and DeepSeek evidencing greater misalignment, mainly due to overestimations of the midpoint. These results are particularly useful for applied research purposes, as they show that evaluation should not be limited to "within/outside the range": the rating allows for the interpretation of the accuracy and the direction of the deviation (under/overestimation). Given the limited sample size (n=6), it is recommended to treat the evidence as diagnostic and expand the empirical base in subsequent studies, testing the robustness of the observed pattern in a larger set of software, different licensing regimes, and multiple price sources.

Table 6. Evaluation of prices estimated by LLMs

Software / billing unit	Market price (US$/year)	ChatGPT Estimative	Gemini Estimative	DeepSeek Estimative	Source of market price (link, website, date of access)	LLMs with correctly estimated range
INTEGRA. By an organization independent of the number of users.	US$ 13,309.00	US$ 8,000.00–US$ 25,000.00. LLM estimative includes market price: Yes	US$ 23,000.00–US$ 50,000.00. Does the LLM estimative include the market price?: No	US$ 20,000.00–US$ 70,000.00. Does the LLM estimative include the market price?: No	Official Gazette of the Union (DOU) – Extract of Contract No. 74/2021 (UASG 158141 – IFRS), Section 3, 12/16/2021, p. 85: `https://pesquisa.in.gov.br/imprensa/servlet/INPDFViewer?captchafield=firstAccess&data=16%2F12%2F2021&jornal=530&pagina=85`. (Accessed: December 22, 2025). Exchange rate (PTAX average purchase rate – Ipeadata/BCB, series 38590): `https://www.ipeadata.gov.br/ExibeSerie.aspx?module=M&serid=38590`. (Accessed: December 22, 2025). Conversion of the contract value from BRL to USD/year via PTAX (average purchase rate).	ChatGPT
HYPE Innovation. Medium-sized companies (∼100 users).	US$ 58,560.00	US$ 50,000.00–US$ 100,000.00. LLM estimative includes market price: Yes	US$ 50,000.00–US$ 90,000.00. LLM estimative includes market price: Yes	US$ 36,000.00–US$ 60,000.00. LLM estimative includes market price: Yes	Kangaruu – "7 innovation management software in 2024" (HYPE: 50,000/year): `https://kangaruu.io/best-innovation-management-software/`. (Accessed: December 22, 2025). ECB – Euro foreign exchange reference rates (EUR→USD): `https://www.ecb.europa.eu/stats/shared/pdf/eurofxref.pdf`. (Accessed: December 22, 2025). Market value based on public estimate (EUR) converted to USD.	ChatGPT; Gemini; DeepSeek
IdeaScale. Medium-sized companies (∼100 users).	US$ 17,499.50	US$ 60,000.00–US$ 70,000.00. LLM estimative includes market price: No	US$ 15,000.00–US$ 25,000.00. LLM estimative includes market price: Yes	US$ 90,000.00–US$ 270,000.00. LLM estimative includes market price: No	TrustRadius – IdeaScale Pricing (Advance: US$ 34,999 for 2 years): `https://www.trustradius.com/products/ideascale/pricing`. (Accessed: December 22, 2025). Annualized price (÷2) based on the value published on a biannual basis (Advance edition).	Gemini
Viima. Medium-sized companies (∼100 users).	US$ 12,087.61	US$ 6,000.00–US$ 18,000.00. LLM estimative includes market price: Yes	US$ 6,000.00–US$ 15,000.00. LLM estimative includes market price: Yes	US$ 12,000.00–US$ 60,000.00. LLM estimative includes market price: Yes	Viima (HYPE Boards) – Pricing (Pro: US$ 499/month; additional licenses: 7/month per user): `https://www.viima.com/pricing`. (Accessed: December 22, 2025). ECB – Euro foreign exchange reference rates (EUR→USD): `https://www.ecb.europa.eu/stats/shared/pdf/eurofxref.pdf`. (Accessed: December 22, 2025). Annual price calculated with monthly fee + additional licenses (assumed 162 users).	ChatGPT; Gemini; DeepSeek

Continued on next page

Table 6. Evaluation of prices estimated by LLMs (continued)

Software / billing unit	Market price (US$/year)	ChatGPT Estimative	Gemini Estimative	DeepSeek Estimative	Source of market price (link, website, date of access)	LLMs with correctly estimated range
Qmarkets. Medium-sized companies (∼100 users).	US$ 46,848.00	US$ 33,000.00–US$ 55,000.00. LLM estimative includes market price: Yes	US$ 30,000.00–US$ 55,000.00. LLM estimative includes market price: Yes	US$ 70,000.00–US$ 200,000.00. LLM estimative includes market price: No	Kangaruu – "7 innovation management software in 2024" (Qmarkets: 30,000–50,000/year): `https://kangaruu.io/best-innovation-management-software/`. (Accessed: December 22, 2025). ECB – Euro foreign exchange reference rates (EUR→USD): `https://www.ecb.europa.eu/stats/shared/pdf/eurofxref.pdf`. (Accessed: December 22, 2025). Public range in EUR; midpoint was adopted to obtain a single annual value and convert to USD.	ChatGPT; Gemini
Salesforce. Per user.	US$ 1,200.00	US$ 1,200.00–US$ 1,200.00. LLM estimative includes market price: Yes	US$ 3,600.00–US$ 6,000.00. LLM estimative includes market price: No	US$ 2,400.00–US$ 7,200.00. LLM estimative includes market price: No	Salesforce – Sales Pricing (Starter Suite: US$ 25/user/month): `https://www.salesforce.com/sales/pricing/`. (Accessed: December 22, 2025). Price calculated for 4 users on the Starter Suite plan (US$ 25/user/month).	ChatGPT

Table 7. Quotation (market price ÷ midpoint) by LLM

Software	Market price (US$/year)	ChatGPT — Midpoint (US$)	ChatGPT — Rate (%)	Gemini — Midpoint (US$)	Gemini — Rate (%)	DeepSeek — Midpoint (US$)	DeepSeek — Rate (%)
INTEGRA	US$ 13,309.00	US$ 16,500.00	80.70%	US$ 36,500.00	36.50%	US$ 45,000.00	29.60%
HYPE Innovation	US$ 58,560.00	US$ 75,000.00	78.10%	US$ 70,000.00	83.70%	US$ 48,000.00	122.00%
IdeaScale	US$ 17,499.50	US$ 65,000.00	26.90%	US$ 20,000.00	87.50%	US$ 180,000.00	9.70%
Viima	US$ 12,087.61	US$ 12,000.00	100.70%	US$ 10,500.00	115.10%	US$ 36,000.00	33.60%
Qmarkets	US$ 46,848.00	US$ 44,000.00	106.50%	US$ 42,500.00	110.20%	US$ 135,000.00	34.70%
Salesforce	US$ 1,200.00	US$ 1,200.00	100.00%	US$ 4,800.00	25.00%	US$ 4,800.00	25.00%

From the point of view of accuracy and plausibility, there is no single "true price" for each solution, since commercial proposals vary according to the negotiated scope, discounts, and contractual context. Even so, as an external verification, formal quotes were requested from the suppliers of the three software programs analyzed after the experiment was conducted. Although the specific figures are not disclosed for confidentiality reasons, the comparison indicated that the quoted values fell within the ranges estimated/suggested by the protocol with the LLMs, which suggests good adherence in terms of orders of magnitude and reinforces the usefulness of the approach to support initial planning and negotiation phases.

Regarding stability and accuracy, the technical replication of experiments with the same prompt, the same sitemap PDF, and constant inference parameters allowed for the evaluation of the robustness of the outputs. The replicated rounds were compared according to (a) similarity of feature lists, (b) stability of price ranges, and (c) coincidence of the public sources used. As a control, each experiment was technically replicated (new chat, same prompt, and same PDF), comparing the similarity of feature lists, the stability of price ranges in terms of order of magnitude, and the coincidence of the cited public sources; discrepancies were subjected to a directed rereading of the sitemaps and pricing pages, which could result in discarding or adjustment. In practice, the observed differences remained within these levels, indicating that, given the same instructions and inputs, LLM tends to produce relatively stable, albeit non-deterministic, estimates.

Regarding the complexity and effort of application, the LLM-based approach differs substantially from the traditional pricing and cost estimation methods discussed in the theoretical framework, such as algorithmic models (e.g., COCOMO and variations) and methods based on functional metrics (such as Function Points). While these techniques demand structured historical data, parameter calibration, and often detailed measurements of the software's functional size, the protocol presented here operates with significantly lighter inputs: public sitemaps, institutional documentation, and pricing pages available on the web. This reduces the barrier to entry and the time required to obtain an initial price band, albeit at the cost of less granularity in terms of development effort, deployment cost, and TCO (Total Cost of Ownership).

Finally, analyzing the results from a perspective of transparency and reproducibility, the approach benefits from the use of a single LLM model, a standardized prompt, the isolation of chats, and the archiving of artifacts (PDFs, transcripts, and tables). Each step from functional extraction to price benchmarking is described in a traceable way, allowing other researchers to reproduce the procedure with the same inputs or adapt it to new contexts. In contrast, many traditional methods for estimating effort and cost, although mathematically well-defined, rely on proprietary calibrations, non-public databases, or poorly documented fine-tuning in organizational environments, which hinders external replication.

In summary, the exploratory evaluation indicates that the analogy pricing protocol, mediated by LLM, is capable of producing estimates consistent with market price ranges, with acceptable stability between runs and reduced application costs, especially useful in the initial phases of analysis and comparison of alternatives. On the other hand, the approach does not replace traditional estimation methods based on effort and functional complexity. Rather, it positions itself as a complement for rapid screening and definition of price bands. These bands can later be refined by techniques such as Function Points, COCOMO, or hybrid models based on machine learning. A formal evaluation comparing average errors, variance, and application costs between these methods, in a larger set of cases with known prices, remains an agenda item for future work.

5 Conclusion

This study aimed to propose and test a software pricing protocol by analogy, mediated by language models and based on prompt engineering and public evidence (official sitemaps and open sources), in order to produce auditable and comparable price ranges and verify their adherence to real market quotations. To operationalize this general objective, the work consolidated official sitemaps in PDFs and extracted, in a standardized way, the functional outline of each solution; conducted competitive benchmarking in public sources, identifying pricing models and orders of magnitude practiced; estimated, for each software, value ranges using three LLMs under the same prompt and with isolated chats; evaluated the estimates using two simple indicators (range coverage and percentage quotation); and recorded artifacts and evidence (PDFs, transcripts, and spreadsheets) to allow replication and scrutiny by third parties.

Applied in a standardized way to three models (ChatGPT 5.1 Thinking, Gemini 3 Pro, and DeepSeek-V3.2) and six software programs (INTEGRA, Salesforce, HYPE Innovation, IdeaScale, Viima/HYPE Boards, and Qmarkets), the protocol generated price ranges consistent with the functional positioning and billing models observed, preserving traceability through the systematic archiving of inputs and price evidence. The results support the idea that pricing by analogy, when guided by a single prompt and verifiable sources, can produce market-aligned estimates in terms of order of magnitude, provided it is interpreted as an exploratory tool aimed at screening and initially directing decisions.

The range coverage assessment (annual market price within/outside the range) showed superior performance for ChatGPT (5/6 software programs with contained market price), followed by Gemini (4/6) and DeepSeek (2/6), suggesting differences in inter-model consistency even under the same protocol. Additionally, the percentage quote (market price ÷ midpoint) proved useful for qualifying the accuracy and interpreting the direction of the deviation (underestimation or overestimation), including in cases where the observed price was marginally outside the estimated range. Together, these two simple indicators reinforce the protocol's usefulness as a pragmatic mechanism for external validation, while also making explicit the variations in robustness between models.

The article's contributions are twofold. Methodologically, the work consolidates a reproducible workflow that combines chat isolation, source restriction in the "file-only" stage, metadata recording, and evidence preservation, increasing the

transparency and auditability of LLM use in market analysis. Empirically and managerially, it delivers a comparative synthesis of six platforms, articulating functional outlines, price bands, and public sources, which can support preliminary positioning decisions, definition of minimum viable scope, prioritization of integrations, and screening of alternatives before formal contracting and negotiation processes.

Relevant limitations are acknowledged: some official websites do not make software prices readily available; reliance on public data may under/over-represent functionalities and prices; catalog changes and commercial policies over time affect comparability; and the sample size is small (n=6). As a future agenda, it is proposed to expand the sample and sources, incorporate automatic evidence-checking mechanisms, and systematically compare the approach with traditional methods (e.g., Function Points, COCOMO, and hybrid models), evaluating application cost, variance, and performance in scenarios with observable prices, in order to more precisely define where pricing by analogy mediated by LLMs offers greater practical gain and where methodological complementarity is required.

Declarations

Authors' Contributions

Each author made substantial contributions to this manuscript. All authors have reviewed, edited, and approved the final version of the manuscript.

Competing interests

The authors declare that they have no competing interests.

Acknowledgements

The authors thank the Instituto Federal do Rio Grande do Sul (IFRS) for institutional support and the reviewers for their valuable comments that helped improve this article.

Funding

This research received no external funding.

Availability of data and materials

The datasets generated and/or analyzed during the current study are available in the GitHub repository: https://github.com/gregfermun/AI-Driven-Software-Pricing-An-Integrated-Approach-with-Prompt-Engineering-for-Market-Analysis

References

Agrawal, A., Jain, N., and Sheikh, A. (2016). Software cost estimation using artificial neural networks. In *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. DOI: 10.1109/ICACCI.2016.7732254.

AI Anatomy Map (2020). The AI anatomy map. Retrieved December 23, 2025, from `https://anatomyof.ai/`.

Alauthman, M., Ghanem, W., and Al-Dhaqm, A. (2023). A systematic literature review for just-in-time defect prediction. *International Journal of Systems and Software Science and Computational Intelligence*, 14(1):1–19. DOI: 10.4018/IJSSCI.328359.

Ali, S., Almajali, S., and Tahat, L. (2023). Artificial intelligence and ChatGPT: A review of the challenges and opportunities of AI-generated text. *IEEE Access*, 11:100774–100789. DOI: 10.1109/ACCESS.2023.3316530.

Baur, C., Groh, A., and Jung, F. (2014). Value-based pricing in digital services: A strategic pricing framework. *Journal of Business Research*, 67(5):976–982. DOI: 10.1016/j.jbusres.2013.08.007.

Bodendorf, F., Lutz, M., and Franke, J. (2021). Valuation and pricing of software licenses to support supplier–buyer negotiations: A case study in the automotive industry. *Managerial and Decision Economics*, 42(7):1686–1702. DOI: 10.1002/mde.3336.

Boussioux, L., Lai, Y., Malik, H., Menick, J., Nguyen, A., and Zoph, B. (2024). The cost of using AI for writing. *Organization Science*, 35(5):1589–1607.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901.

Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., Chen, H., Yi, X., Wang, C., Wang, Y., Ye, W., Zhang, Y., Chang, Y., Yu, P. S., Yang, Q., and Xie, X. (2024). A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3):1–45. DOI: 10.1145/3641289.

De Cremer, D., Mollick, E., and Bahadoor, S. (2023). How to use generative AI to augment your work. *Harvard Business Review*.

Dell'Acqua, F., Eling, M., Gaur, V., Lakhani, K., and Nori, H. (2023). Navigating the jagged technological frontier: Field experimental evidence of the effects of AI on knowledge worker productivity and quality. *Harvard Business School Working Paper*, (24–013).

Fan, Y., Liu, Y., Zhang, W., and Chen, H. (2024). A communication theory perspective on prompting engineering methods and measures for AI text generation: Proposal for a research agenda. *International Journal of Human-Computer Interaction*. DOI: 10.1080/10447318.2024.2316402.

Gao, J., Cao, Z., and Li, W. (2024). SelfCP: Compressing over-limit prompts via the frozen large language model itself. *Information Processing & Management*, 61(6):103873. DOI: 10.1016/j.ipm.2024.103873.

Harmon, R., Demirkan, H., Hefley, B., and Auseklis, N. (2009). Pricing strategies for information technology ser-

vices: A value-based approach. *Journal of Service Science*, 2(2):33–50. DOI: 10.1287/serv.2.1_2.33.

Henrickson, L. and Meroño-Peñuela, A. (2023). Prompting meaning: A hermeneutic approach to optimizing prompt engineering with ChatGPT. *AI & Society*. DOI: 10.1007/s00146-023-01737-4.

Hoc, T., Brule, E., and Treco, E. (2023). Transfer learning in deep models for software effort estimation. *Journal of Systems and Software*, 196:111563. DOI: 10.1016/j.jss.2022.111563.

Holmström, J. and Carroll, N. (2024). How organizations can innovate with generative AI. *Business Horizons*. DOI: 10.1016/j.bushor.2024.02.010.

Huang, A. H. and Chang, K.-W. (2023). Fine-tuning and in-context learning with large language models for prompt engineering: A comparative analysis of performance and cost. *Findings of the Association for Computational Linguistics: ACL 2023*. DOI: 10.18653/v1/2023.findings-acl.67.

Huang, M.-H. and Rust, R. T. (2024). Generative artificial intelligence in marketing: A framework for research and applications. *Journal of Marketing*, 88(1):53–77.

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., and Fung, P. (2023). Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):Article 248. DOI: 10.1145/3571730.

Jørgensen, M. and Shepperd, M. (2007). A systematic review of software development cost estimation studies. *IEEE Transactions on Software Engineering*, 33(1):33–53. DOI: 10.1109/TSE.2007.256943.

Kietzmann, J. and Park, C. W. (2024). Written by ChatGPT: AI, large language models, conversational chatbots, and their place in society and business. *Business Horizons*, 67(5):453–459. DOI: 10.1016/j.bushor.2024.06.002.

Korzynski, P., Mazurek, G., and Haenlein, M. (2023). Leveraging large language models for open source intelligence. *Entrepreneurship and Business Economics Review*, 11(3):131–152. DOI: 10.1007/s40821-023-00226-1.

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., and Neubig, G. (2023). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35. DOI: 10.1145/3560815.

Liu, X., Zhang, Y., Chen, W., and Liu, Y. (2024). Jailbreak and adversarial prompt injection attacks: Understanding security vulnerabilities in LLMs. *arXiv preprint*.

López-Martín, C. (2015). Predictive accuracy comparison between neural networks and statistical regression for development effort of software projects. *Applied Soft Computing*, 27:434–449. DOI: 10.1016/j.asoc.2014.10.030.

MacRae, M. (2023). How generative AI is changing software development. *MIT Sloan Management Review*.

Oppenlaender, J. (2024). Prompt engineering for text-based generative AI: A (literary) perspective on prompt modifiers. *International Journal of Human-Computer Interaction*. DOI: 10.1080/10447318.2024.2431761.

Qassem, M. and Saleh, A. (2023). Impact of machine learning techniques on software effort estimation. *International Research Journal of Innovations in Engineering and Technology*, 7(1):68–74.

Qin, Y., Hu, S., Lin, Y., Chen, W., Ding, N., Cui, G., Zeng, Z., Huang, Y., Xiao, C., Han, C., Fung, Y. R., Su, Y., Wang, H., Qian, C., Shi, R., Zheng, R., Liu, Z., Zhou, J., Zhang, P., Sun, M., and Liu, Z. (2024). Tool learning with foundation models. *ACM Computing Surveys*, 57(3):1–38. DOI: 10.1145/3704435.

Rankovic, N., Miskovic, V., and Jovanovic, M. (2021). A hybrid ANN approach for software cost estimation. *IEEE Access*, 9:153737–153748. DOI: 10.1109/ACCESS.2021.3127958.

Rashid, M., Riaz, M. R., Ahmad, S., and Khan, S. (2025). A systematic literature review on software cost estimation models: Evolution and emerging trends. *Alexandria Engineering Journal*, 102:162–170. DOI: 10.1016/j.aej.2025.02.064.

Robertson, J., Prado, M., and Nielsen, D. (2024). Prompt engineering: The art and science of asking better questions. *Business Horizons*, 67(4):409–418. DOI: 10.1016/j.bushor.2024.03.008.

Saljoughinejad, S. and Khatibi, V. (2018). A comparative analysis of COCOMO-based estimation models. *Software Quality Journal*, 26(2):399–421. DOI: 10.1007/s11219-016-9339-5.

Santaella, L. (2023). Artificial intelligence and daily life: From background algorithms to generative systems. *Journal of Digital Studies*, 2(1):1–22.

Short, J. and Short, T. (2023). Real or fake? How artificial intelligence can enhance corporate communication. *Journal of Business Venturing Insights*, 20:e00315. DOI: 10.1016/j.jbvi.2023.e00315.

Sun, Z., Wang, X., Tay, Y., Yang, Y., and Zhou, D. (2023). Recitation-augmented language models. *arXiv preprint*.

Sundberg, L. and Holmström, J. (2024). Prompt engineering: The art of asking the right questions. *Business Horizons*, 67(5):561–570. DOI: 10.1016/j.bushor.2024.04.014.

Verner, J. M., Sampson, J., and Cerpa, N. (2008). What factors lead to software project failure? In *2008 Second International Conference on Research Challenges in Information Science*, pages 71–80. DOI: 10.1109/RCIS.2008.4632095.

Villalobos-Arias, L., Quesada-López, C., Martínez, A., and Jenkins, M. (2020). Evaluating hyper-parameter tuning using random search in support vector machines for software effort estimation. In *Proceedings of the 16th ACM International Conference on Predictive Models and Data Analytics in Software Engineering*, pages 31–40. DOI: 10.1145/3408301.3408305.

Wang, B., Min, S., Hou, X., Chen, L., Hu, S., Chen, J., Zhang, W., Zhou, J., Peng, J., Zhao, Y., Hao, J., and Zhang, J. (2023). Towards understanding chain-of-thought prompting: An empirical study of what matters. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1740–1762. DOI: 10.18653/v1/2023.acl-long.153.

Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D., Chi, E. H., Hashimoto, T., Vinyals, O., Liang, P., Dean, J., and Fedus, W. (2022). Emergent abilities of large language models. *Transactions on Machine Learning*

Research.

Wen, J., Li, S., Lin, Z., Hu, Y., and Huang, C. (2012). Systematic literature review of machine learning based software development effort estimation models. *Information and Software Technology*, 54(1):41–59. DOI: 10.1016/j.infsof.2011.09.002.

Yang, L., Zhang, S., Wang, Y., and Li, Y. (2025). Robust prompting practices and architectures for LLM controllability with external knowledge integration. *AI and Ethics*, 5(1):89–105. DOI: 10.1007/s43681-024-00456-3.

Zhang, D., Liu, Y., Li, X., and Wang, H. (2024). Mixed data classification of clinical notes in electronic medical records. *Journal of Biomedical Informatics*, 149:104571. DOI: 10.1016/j.jbi.2023.104571.

Zhang, Y., Peng, N., Li, X., and Wang, C. (2023). Improving GPT-4 performance on clinical note classification with structured prompts and explicit reasoning steps. *Journal of Biomedical Informatics*, 146:104504. DOI: 10.1016/j.jbi.2023.104504.