POLITECNICO DI TORINO

POLITECNICO DI TORINO

Master Degree course in Management Engineering

Master Degree Thesis

Enhancing BPMN Exercise Evaluation: Expanded Solution Spaces and Advanced Validation Frameworks

Supervisors
Prof. Riccardo Coppola
Giacomo Garaccione

Candidate
ZAHIDE PINAR YAKICI

Academic Year 2024-2025

Acknowledgements

I would like to express my deepest gratitude to Professor Riccardo Coppola, my supervisor, for their continuous guidance throughout this thesis. Their insightful advice and expertise have been invaluable to my academic and personal growth.

I would also like to thank Giacomo Garaccione for their constant support, constructive feedback, and for sharing their knowledge with me during this work.

Special thanks to my real çekirdek aile, my mom, dad and my sister Deniz for being the best and funniest family there can ever be. At every step of my journey; you were there for me, always trying to push me one step forward, encouraging me to learn more every day, taught me to be curious with the correct questions to be asked and hug me with the warmest love possible.

I owe another special thanks to my grandparents; Babaannem, Ağa Dedem, Anneannem and Faruk Dedem, for raising me, always being there for me, being the "fun" parents and guiding me to become the person that I am today not only academically but in every aspect of my life.

I would also like to thank to my friends; my partner in crime for half of my life Sarin for always being sincere, caring and beautifully chaotic constant in my life, my sister (wife at heart) Doğa for being my home far away from home and reminding me what true friendship feels like. Also thanks to Ata, Alican, Alkın, and Katya for the endless fun, the deep talks, and the memories that I will always cherish. Your friendship has made every step of this journey more joyful.

To my colleagues, Elena and Agnese, and classmates, thank you for filling these years with laughter, support, and late-night teamwork. You have made even the hardest moments easier and reminded me how special it is to grow and learn together.

Finally, my Tommaso, thank you for always believing in me, heart-shaped snacks, being my biggest supporter, greatest motivation and number one fan.

Abstract

The recent advancements in Large Language Models (LLMs) have opened new possibilities for the automation of software and process modelling. This thesis investigates the capability of AI systems to generate and interpret Business Process Model and Notation (BPMN) diagrams, with the aim of assessing their sufficiency, syntactic correctness and structural coherence. Although BPMN offers a standardized visual language for describing business workflows, creating and evaluating these diagrams manually is still a demanding and error-prone activity.

To tackle this problem, four different AI models (ChatGPT, Copilot, Gemini and DeepSeek) were evaluated through a structured framework inspired by the COPE (Context, Objective, Prompt, Evaluation) methodology. Fifteen BPMN exercises, collected from diversified business and operational sectors, were analysed using two complementary scoring systems: one designed to quantify the relative difficulty of each exercise, and another developed to evaluate and compare the accuracy of AI-generated diagrams against reference solutions. Each exercise was solved, and the resulting diagrams were then evaluated both qualitatively and statistically through the Kruskal-Wallis and Wilcoxon Signed-Rank Tests.

The statistical analysis showed that Gemini achieved the highest score and most consistent performance, followed by ChatGPT, while Copilot and DeepSeek showed less reliable results. The qualitative analysis revealed that syntactic precision does not necessarily ensure semantic completeness, highlighting the significance of contextual comprehension when generating BPMN diagrams.

In conclusion, this thesis provides an empirical framework for evaluating the AI-generated process models and demonstrates that the reliability and understanding of AI-assisted process modelling can be improved by combining structured scoring and statistical validation.

Contents

1 Introduction 5

2 Background 7

2.1 Definition of Software Modelling . . . 7

2.2 Definition of Process Modelling . . . 9

2.2.1 Definition, Characteristics and Notions of Business Process Modelling 10

2.2.2 Common BPMN Elements . . . 11

2.2.3 Formats available for BPMN Diagrams . . . 13

2.2.4 Syntax and Semantic Errors in Business Process Modelling . . . 14

2.2.5 Evaluating BPMN Diagrams . . . 16

2.3 Large Language Models and Conversational Artificial Intelligence . . . 17

2.3.1 Introduction to Large Language Models (LLMs) . . . 17

2.3.2 Conversational Artificial Intelligence: Overview of Tools and Platforms . . . 18

2.3.3 Benefits and Limitations of Conversational AI Technologies . . . 19

2.3.4 Ethical and Societal Implications of AI Systems . . . 20

3 Applications of BPMN and AI in Process Modeling 23

3.1 BPMN Applications in Practice . . . 23

3.1.1 Applications Across Sectors . . . 24

3.1.2 Role of BPMN in Digitalization and Process Automation . . . 24

3.2 AI-Assisted BPMN Modelling . . . 25

3.2.1 Integration of AI Technologies . . . 25

3.2.2 Benefits and Opportunities . . . 26

3.2.3 Challenges and Limitations . . . 26

3.3 Previous Studies and Research Trends . . . 27

3.3.1 AI and Process Mining . . . 27

3.3.2 Natural Language Processing for BPMN Generation . . . 27

3.3.3 Large Language Models and BPMN Automation . . . 28

3.3.4 Evaluation Frameworks and Methodological Gaps . . . 28

3.3.5 Research Gap and Contribution of this Study . . . 28

4 Methodology 31
4.1 Study Framework and Dataset Preparation . . . 31
4.1.1 Exercises . . . 31
4.1.2 Scoring System . . . 34
4.2 Prompt Design . . . 36
4.2.1 The COPE Methodology: Contextualizing Prompt-Based Evaluation 38
4.3 Evaluation Methodology . . . 39
4.3.1 Scoring System for Diagram Evaluation . . . 39
4.3.2 Structure of Scoring System . . . 40
4.3.3 Explanation of Categories and Constraints . . . 41
4.4 Statistical Methods . . . 51
4.5 Limitations of the Evaluation Methodology . . . 52

5 Results and Discussion 55
5.1 Results . . . 55
5.1.1 Overview of the dataset . . . 55
5.1.2 Descriptive Statistics . . . 55
5.1.3 Statistical Analysis . . . 57
5.1.4 Summary of the Findings . . . 57
5.2 Discussion . . . 57
5.2.1 Interpretation of Quantitative Findings . . . 57
5.2.2 Qualitative Evaluation of Diagram Quality . . . 58
5.2.3 Comparative Observations . . . 59
5.2.4 Model Variability and Reliability . . . 60
5.2.5 Methodological and Practical Reflections . . . 61
5.2.6 Broader Implications . . . 61

6 Conclusion and Future Works 63
6.1 Conclusion . . . 63
6.2 Future Works . . . 64

Bibliography 65

List of Tables 71

List of Figures 73

A List of BPMN Exercises and Sources 75

Chapter 1

Introduction

This thesis presents a structured approach to evaluating how different **Large Language Models** perform in automatically generating **BPMN(Business Process Model Notation)** based on textual process descriptions in different forms. This methodology is based on practical and comparative frame work of 15 BPMN exercises. All of those exercises are identified by a scoring system to evaluate their difficulty levels in order to have a broad range of exercises. The exercises are solved by four different Artificial Intelligence models, **ChatGPT (GPT-5), Microsoft Copilot, Google Gemini and DeepSeek.** Each model was prompted individually in **.bpmn** form to turn to a scheme in Camunda Modeler and their outputs were systematically collected for evaluation.

To assess the quality of AI-generated diagrams, **a multi-criteria scoring system** was developed that is focusing on key dimensions such as **completeness, correctness, style and clarity, BPMN syntax and similarity to reference**.

This evaluation is not a one time output comparison but it is an **iterative prompting process** using **COPE method (Contextualizing Prompt-Based Evaluation)**. By doing this, it was possible to refine the prompts and gain a more clear understanding of how each model adapts to different task requirements.

The evaluation is conducted manually using a standardized matrix, ensuring consistency and transparency in how the scores were assigned.

This methodology enables a **fair** and **detailed** comparison across models and at the same time explores the potential of LLMs to support and automate structured process modelling tasks.

The remainder of this thesis is structured as follows: In Chapter 2 Large Language Models (LLMs) and Conversational Artificial Intelligence are introduced, together with the theoretical backdrop and definitions of important software and process modeling concepts and BPMN principles. Chapter 3 discusses the practical application of BPMN and AI in process modelling, highlighting integration opportunities, challenges, and recent research trends in AI-assisted modelling. The study's methodology is presented in Chapter

4, which includes information on the BPMN exercise dataset, the scoring schemes used, and the statistical methods employed for assessment. Chapter 5 reports and interprets the results, offering both quantitative and qualitative analyses of model performance, followed by a discussion of broader implications. Finally, Chapter 6 concludes the study by summarizing the main findings and suggests possibilities for future research.

Chapter 2

Background

This chapter provides the theoretical and conceptual foundations of this study. It introduces the fundamental principles of software and process modelling, outlines the role of BPMN in representing business workflows and examines the recent integration of artificial intelligence into process automation and model generation.

2.1 Definition of Software Modelling

The process of developing abstract, simplified representations of a software system or component is known as modelling [19]. Software models help engineers, designers, and both technical and non-technical stakeholders by transforming a complex system into a simplified, visual and structured format to have a better and easier understanding, planning, analyzing, and communicating the structure and behaviour of complex systems before implementation begins. Modelling creates a universal language that facilitates shared understanding for different roles in a project [11]. Essentially, software modelling allows us to answer essential questions:

- *What is the system supposed to do?*
- *How does the system organized?*
- *Who are the users and actors of the system?*
- *How does different components interact?*

These questions guide how we structure system knowledge during early stages of design.

In the past, writing code was the primary focus on software development, and only relatively less attention was paid on structured documentation and high-level design. This method became ineffective and prone to mistakes as the systems increased in size and complexity.

Nowadays, modern software engineering has a more model-driven approach, first building models to capture the requirements and structure of the system and then translating those models into working code.

Recent studies in Model-Driven Engineering (MDE) by France and Rumpe [19] have further reinforced the importance of software modeling, highlighting how high-level abstract models can bridge the gap between problem domains and software implementations. By describing systems from multiple perspectives and at varying levels of abstraction, MDE approaches enable systematic transformations from conceptual models to executable software, improving consistency, reducing errors, and supporting collaboration among stakeholders [17, 18].

Software modeling allows us engineers and designers to explore ideas, evaluate alternatives, and define the expected behavior and structure of the system in a clear and systematic way. It plays a fundamental role throughout the software development life cycle, from requirement analysis and architectural design to validation, documentation, and maintenance.

To ensure clear and consistent communication among such diverse stakeholders, visual modeling languages such as UML (Unified Modeling Language) and SysML (Systems Modeling Language) are widely used [22]. UML offers a collection of standardized diagram types to model both the static and dynamic aspects of software systems. SysML builds upon UML's foundations, but expands its capabilities to represent non-software elements, including hardware components, physical processes, and environmental constraints. This makes it particularly suitable for interdisciplinary system design. Both languages provide formal notations that help describe software architecture, logic, and behaviour across varying levels of abstraction.

Different types of software models can be used depending on what they represent [19, 30]:

- **Structural models** (e.g., UML class diagrams): Describes the organization of the system. It focuses on its components, data structures, and their relationships.
- **Behavioral models** (e.g., activity, sequence, or state diagrams): Focus on how the system behaves over time, how processes evolve, and how components interact dynamically.
- **Functional models** (e.g., data flow diagrams or use case diagrams): Focuses on what the system does with inputs and outputs.

Each of these model types plays a crucial role in supporting core engineering practices, such as:

- **Requirements Analysis:** This helps to better understand system limitations and user needs in the early stages of design.

- **System Design and Architecture:** Arranges system components and workflows.
- **Risk Reduction:** Helps to identify design flaws, inconsistencies in an early [11].
- **Documentation and Maintenance:** Provides a long-term reference for future development to support and facilitate upgrades, troubleshooting, and new developer onboarding [52].
- **Communication:** Enables stakeholders with different backgrounds to understand and discuss the system design effectively [11].

The choice of modelling types depends on several factors:

- **Project requirements:** Includes the system's domain, scope, and objectives.
- **Scalability:** The anticipated expected growth or evolution of the system.
- **Tool complexity:** The degree to which a modelling tool is easy to use, expensive, or adaptable [52].
- **Integration needs:** The way the model works with other platforms, systems, or formats.

As software continues to integrate with AI, embedded systems, cloud services, and automation, modelling is increasingly essential for ensuring reliability, traceability, and performance [57].

2.2 Definition of Process Modelling

Process modelling refers to the activity of representing the sequence of actions, decisions, and interactions that define how a specific task or system operates. It creates a visual and structured description of processes through diagrams so that the stakeholders would be able to analyse, communicate and improve how a process unfolds [38].

A **process** is a **chain of activies** that transforms a defined **input** to an expected **output**. These activities may include human actors, automated systems or a combination of both of them. By modelling a process; engineers and analysts are able to capture not only the flow of operations but also the rules, conditions, constraints and the roles that are crucial for the system's behaviour. It is useful to identify the inefficiencies, discovering bottlenecks and also evaluate alternative paths within a workflow [37].

Process modeling is widely used in business **process management**, **project management**, **software development** and **system optimization**. Regardless of its scope, its core value lies in providing a **shared and common representation** of how things work, making it easier for teams to have a common understanding, evaluation and redesign processes [36].

A process model typically includes the different components, such as:

- **Activities**: The units of work performed in the process.
- **Decision points(Gateways)**: Requirements or guidelines that determine which path is followed.
- **Events**: Triggers that start, interrupt, or terminate the process.
- **Flows**: Arrows that shows the sequence of actions and data/information exchange.
- **Actors**: Entities (people, roles, systems) involved in the process.

Process models offer value both for technical and collaborative aspects of system development. They contribute to clarity by making complex workflows more understandable and easier to follow. They also serve as a reliable tool for documenting current or planned processes, providing a reference that is both reusable and easy to interpret. Moreover, modelling supports analysis and refinement by helping to identify inefficiencies or unnecessary steps that may hinder performance. These models can also be used in simulation and validation scenarios, providing a way to validate process revisions prior to real-world usage. Finally, process models enhance communication by establishing a common framework that helps all stakeholders and engage with the system in a meaningful way.

Process models can represent both the current state (as-is) and a proposed expected version (to-be) of a system. This dual perspective supports iterative improvement cycles and data-driven decision-making, especially in complex systems where multiple actors interact with changing requirements.

Among the various modelling notations available, BPMN (Business Process Model and Notation) has has become a commonly used standard because of its balance of formal precision and intuitive visual representation. It allows both technical and non-technical users to engage with the process logic in a comprehensible way. BPMN diagrams support different perspectives such as individual activities, end-to-end processes, inter-organization interactions and they are frequently used in digital transformation initiatives and automated process evaluation tools.

2.2.1 Definition, Characteristics and Notions of Business Process Modelling

Business Process Modelling (BPM) is a specialized discipline within process modelling that concentrates on evaluating, analysing, and enhancing the workflows and operations within an organization. The main goal of BPM is to capturing the steps, roles, and rules underlying business operations. By doing this, is creates visual representations that not only reflect the day-to-day functioning of an organization but also serve as a basis for strategic improvement and transformation.

At its core, BPM is about representing the flow of tasks, decisions, and interactions that convert inputs into expected outputs as defined before. This is all about detailing

the sequence of activities, the conditions under which different actions are triggered, and the responsibilities of various organizational actors. By doing this, BPM bridges the gap between high-level business strategy and operational execution, ensuring that each process aligns with overall organizational goals [40, 43].

Business process modelling is characterized by a number of distinctive features that set it apart from other modelling practices [21]:

- **Visual Clarity and Accessibility:** BPM uses standardized graphical notations suc as BPMN (Business Process Model and Notation) to illustrate business workflows [36]. These models are designed to be immediately understandable by managers, analysts, and front-line employees alike, offering a clear picture of how tasks are coordinated [38].
- **Focus on Efficiency and Improvement:** By emphasizing the documentation of business activities, BPM supports the identification of redundancies, bottlenecks, or areas of unnecessary complexity. This detailed insight paves the way for targeted process redesign and continuous improvement initiatives.
- **Iterative and Adaptive Nature:** Effective BPM recognizes that business environments are dynamic. Models are developed iteratively, refined based on operational feedback and market changes, which in turn facilitates agile responses to new challenges.
- **Alignment with Business Objectives:** Beyond technical accuracy, BPM is fundamentally driven by the need to support strategic outcomes. Key performance indicators such as cost reduction, quality improvement, customer satisfaction, or regulatory compliance are ensured in every step of the process [20].

2.2.2 Common BPMN Elements

BPMN models are composed of elements that define the flow, structure and the interaction between the participants in a process. Those are not only arbitrary symbols but also these components form a formalized vocabulary that connects human understanding of organizational activities with the logic required for machine-readable process execution. In practice, they serve as the "building blocks" through which analysts model events, actions, and decision mechanisms, ensuring that processes remain both interpretable and executable across different tools and domains.The main BPMN elements are illustrated in the figure below, which provides an overview of the core components.For a more deep explanation, they are discussed individually, focusing on their purpose, notation, and role in defining the logic of a process model [60].

- **Events:**

  They are used to represent the moments that affect the progression of a process, either by initiating it, interrupting it or concluding it. They act as signals that indicate something of significant has taken place such as receiving a message, reaching a

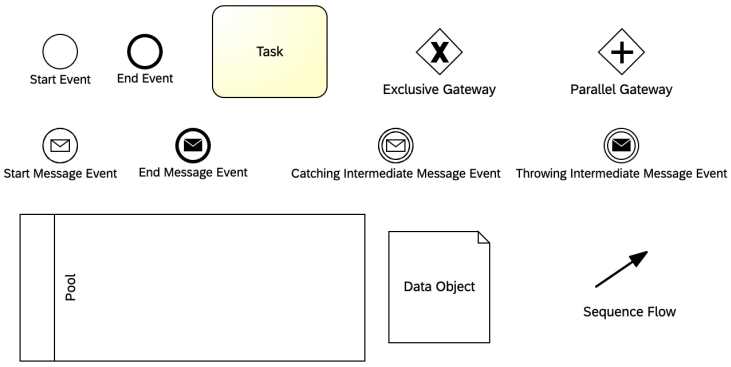

Figure 2.1. Common BPMN Elements

time limit or finishing an activity. The impact of each event on the process depends on both its position within the workflow and the type of trigger it represents [11,44].

Start Event mark the point at which a process begins. They can occur without any specific trigger or they can be triggered by a message, a timer, or an external condition. They are drawn as circles with thin borders and serve as the entry point for all subsequent activities

Intermediate Events occur during the execution of a process. They represent delays, waiting conditions or communication exchanges. Common examples can be **Catching Intermediate Message Event**, which pauses the flow until a message is received and **Throwing Intermediate Message Event**, which sends a message to another participant or process. They are crucial for modelling asynchronous interactions and exception handling.

End Events are illustrated with thick border circles. They signal the termination of a process or subprocess. Depending on their type, they may produce a message, signal completion or trigger compensation mechanisms in other processes. Event-based modelling allows BPMN to capture asynchronous interactions and exception handling mechanisms that are common in distributed systems [50].

- **Activities:**

  Activities represent units of work performed within a process, such as manual or automated tasks. The most common form is **task**, they are drawn as a rounded rectangle and they represent an steps that cannot be further decomposed. More complex behaviours can be represented through **subprocesses**, that represents a series of smaller activities within a hierarchical structure. This ensures scalability and modularity of process models, which are crucial when analysing or automating

workflows [11].

- **Gateways:**

  They are used to control the **splitting** and **joining** of flows. They determine decision points or synchronization mechanisms in the process. **Exclusive Gateway(XOR)**, is represented by a diamond with an "X", which indicates that only one of the outgoing paths will be selected based on a condition. **Parallel Gateway(AND)**, is represented by a "+" symbol, that activates multiple paths simultaneously and allows for concurrent activities. These elements are fundamental for modelling logical branching, synchronization and coordination among tasks. It also enables a clear representation of alternative and parallel behaviours.

- **Pools and Lanes:**

  They represent participants or organizational entities that are involved in the process such as departments, companies or systems. Each pool can be divided into lanes to specify internal roles or responsibilities. The interaction between pools occurs through message flows, which signifies communication between independent actors. This structural partitioning allows the model to have a collaborative process while maintaining clarity between internal and external operations [44].

- **Sequence Flows and Message Flows:**

  Sequence flow is represented by a **solid arrow** that defines the execution order of tasks and events within the same pool. It captures the control flow, ensuring the logical sequencing of process elements. On the other hand, **message flows** typically drawn as **dashed arrows** with open arrowheads. They represent communication between separate pools, highlighting an inter-organizational coordination.

- **Data Objects:**

  They provide information about data that is required or produced by activities. They indicate how data is created, transformed or consumed during process execution. Their inclusion make the process models more complete by integrating information flow, linking BPMN with data-centric process management [61].

2.2.3 Formats available for BPMN Diagrams

BPMN provides a standardized graphical language for modelling business processes. It is intended to close the gap by providing a common standard for workflow representation. One of BPMN's most powerful aspects is its flexibility in providing multiple diagram formats that each of them serving a specific modelling purpose and level of abstraction.

BPMN defines three main types of diagrams, each serving a specific modelling purpose [20, 37, 40, 43]:

- **Process Diagrams:** These are the most commonly used diagrams in BPMN. They describe the internal sequence of activities that occur in a single business entity or

participant. The diagram includes flow objects such as tasks, events, and gateways; connecting objects such as sequence and message flows; and pools and lanes that define who is responsible for each part of the process.

- **Collaboration Diagrams:** These diagrams extend process diagrams by focusing on the interactions between two or more participants. Each participant is represented by a pool, and message flows between them to capture the communication and data exchange. Collaboration diagrams are particularly useful for modelling inter-organizational workflows or multi-role coordination.

- **Choreography Diagrams:** Unlike process or collaboration diagrams that focus on the sequence of tasks, choreography diagrams describe how participants coordinate by exchanging messages. Each activity in a choreography represents a message exchange between two or more participants, showing who initiates the interaction and who responds. These diagrams are useful for high-level modelling of contracts, service-level agreements, or distributed systems.

These diagram formats, complement each other and can be used together to provide a more complete view of a business process.
Flexibility is one of the main values that makes BPMN a powerful tool in both technical and managerial contexts, it allows the same process to be viewed and analysed from different angles depending on the needs of a user.

2.2.4 Syntax and Semantic Errors in Business Process Modelling

BPMN is a useful technique to improve organizational clarity, communication and process efficiency [4]. However, the effectiveness of BPMN does not depend only on if a model is visually complete but also on if it correctly reflects the intended process logic. The identification and management of "errors" plays a central role in ensuring the quality and reliability of business process models [10, 24]. There are two type of errors, **Syntax Errors** and **Semantic Errors** [3, 15, 33, 36, 38, 42].

Syntax Errors

Syntax errors are structural errors that occur when the formal rules defined by the BPMN specifications are broken. BPMN has strict guidelines for how elements such as tasks, events, gateways, and flows can be combined, so when these rules are broken the diagrams may be rejected by BPMN modelling tools or can be misunderstood by stakeholders. Errors in modelling may lead to:

- Incorrect automation logic that results in implementation failures.

- Miscommunication between teams, especially in cross-functional or distributed environments.

- Compliance issues, particularly in regulated industries where precise documentation is critical.
- Inefficient process improvement, as flaws in the model and blocks an accurate analysis.

Typical syntax errors that can be given as example are:

- **Improper connections:** *Connecting a message flow between elements in the same pool, which is not permitted.* BPMN permits to use message flows for inter-actor communication. Using them internally interrupts the communication model and can and mislead both technical and non-technical stakeholders [15].
- **Missing or misplaced events:** *A process model that starts with a user task but has no start event or ends with a task without an end event.* Without clear start/end points, the process has no defined boundaries which makes simulation and execution ambiguous or impossible.
- **Incorrect usage of gateways:** *Using an exclusive gateway (XOR) to split the flow, but failing to include outgoing conditions for each branch.* Misuse of gateways leads to unclear or incorrect decision logic, which may cause multiple paths to execute unexpectedly or none to execute at all.
- **Unconnected elements:** *A user task floating in the diagram without any incoming or outgoing sequence flows.* Elements that are not connected to the main process flow serve no functional purpose and reduce the overall clarity of the model, potentially leading to misinterpretation by stakeholders and inconsistencies during validation.

Semantic Errors

Semantic errors are more subtle and involve a mismatch between what the model shows and what the process is supposed to represent. They occur when the process model fails to accurately represent the intended logic or real-world behaviour of the business process, although structurally correct. A semantic error arises when the diagram describes is not what the process is actually meant to do. This can lead to confusion, incorrect assumptions, or even process failures.

Typical semantic errors that can be given as example are:

- **Ambiguous process logic:** *An exclusive gateway (XOR) that splits into two branches, but without clearly defined conditions for each path.* Ambiguity can lead to unpredictable behavior during automation and uncertainty among users about which path will be followed.

- **Incorrect task allocation:** *Assigning responsibilities to the wrong pool or lane, which misrepresents who is in charge of a particular activity.* Incorrect task placement can confuse stakeholders, cause friction in cross-functional collaboration, and distort accountability within the organization [42].
- **Misplaced event types:** *Using a signal event where a timer or error event is more appropriate.* Incorrect event selection distorts the scope of the error or message, leading to execution errors or miscommunication during design reviews.

2.2.5 Evaluating BPMN Diagrams

Evaluating BPMN diagrams is not a trivial task. Their assessment requires a multi-dimensional analysis that takes into account both the **structural components** and **semantic correctness.** Structural components consist of **tasks, gateways and flows** and on the other hand semantic correctness consists **whether the diagram accurately represents the intended logic**. Over the years, there have been various proposals as an evaluation criteria to assess the quality and usefulness and they typically focused on **completeness, correctness, style and clarity, BPMN syntax and similarity to reference.**

In order to evaluate BPMN diagrams systematically, it has been highlighted in the literature in both structural and cognitive dimensions that affect the model quality and comprehension. In ("Factors of Process Model Comprehension-Findings from a Series of Experiments", Jan Mendling, Mark Strembeck, and Jan Recker (2012)) [38], it has been explored how the various characteristics of the process models influence the ability of users to correctly interpret them. Empirical findings from the study show that comprehension is impacted by factors such as model **complexity, structuredness, and degree of modularization**. According to these findings, models that are logically organized and well-structured increase understandability, clarity, readability, and naming consistency in model evaluation.

Additionally, the article ("Quality Metrics for Business Process Models" by Irene Vanderfeesten, Jorge Cardoso, Jan Mendling, Hajo A. Reijers, and Wil van der Aalst (2007)) [55] proposes a more formalized set of quality dimensions and measurable metrics for process models. With this framework; **syntactic correctness, completeness, understandability and modularity** are considered as **critical evaluation constraints**. These metrics support the correctness of BPMN syntax but also address how intuitively a model communicates its process logic with stakeholders. These works underline the importance of evaluating BPMN diagrams using a **comprehensive and multi-dimensional approach** that makes us able to look at the full picture.

In this thesis, rather than relying on formal mathematical methods, we adopted a **more practical and human-centered evaluation approach**. It is based on visually observable criteria and qualitative analysis, which reflects the way BPMN diagrams are reviewed and interpreted by stakeholders, technical teams, and AI systems more

accurately. This work offers a more practical and realistic framework for comparing AI-produced process models to the original solutions by emphasizing how understandable, accurate, complete, and clear the generated diagrams are. [3, 25, 35, 45, 46, 48, 55]

2.3 Large Language Models and Conversational Artificial Intelligence

The development of Large Language Models (LLMs) is a turning point in the evolution of Artificial Intelligence, because it shifts the paradigm from task-specific algorithms to general-purpose, language-driven systems. LLMs have significantly advanced ability of machines to process, generate and interact through human language with a fluency that used to be considered as unattainable [1, 53]. LLMs are capable of performing a wide range of natural language processing tasks with minimal task-specific tuning because they are built upon massive datasets and sophisticated neural architectures [36, 38].

Conversational Artificial Intelligence is one of the most impactful applications of LLMs considering that it is used to design machines to engage in dynamic, context-aware dialogues with users. With the conversational use of LLMs, a language which is more fluid and contextually appropriate is more present and it leads to more natural and human-like interactions compared to the systems in the past that relies on rigid decision trees and manually programmed responses.

2.3.1 Introduction to Large Language Models (LLMs)

Large Language Models (LLMs) represent a significant growth in the field of Natural Language Processing (NLP). These models are based on deep learning architectures that enable them to process and generate human-like text by learning from massive amounts of data. The term *large* describes both the quantity of data and the number of parameters included in the models.

LLMs are fundamentally built to predict what word should come next in a sentence, but because of their large size and advanced design, they are capable of doing much more than just completing sentences. They are capable of task such as translation, summarization, question answering, sentiment analysis and coding assistance without being explicitly programmed. This ability is known as emergent behavior, where complex skills arise from general-purpose training [6].

LLMs learn representations directly from raw text data, however the traditional rule-based systems or earlier machine learning models relied on feature engineering; the process of selecting, transforming or creating relevant input variables from raw data to improve the performance of machine learning models. With this ability, the generalization across the domains and tasks are more flexible. Furthermore, LLMs can be adapted to specific applications without retraining from scratch by using techniques such as pre-training and

fine-tuning which reduces the need for domain-specific datasets.

Recent generations of LLMs, have shown that scaling model size and training data leads to significant improvements in language understanding and generation [1]. As we can see from these capabilities, LLMs form the technological foundation of many modern conversationsal AI systems. As mentioned in the work by Brown et al. (2020) [6], the development of GPT-3 demonstrated that a sufficiently large and well-trained language model could exhibit few-shot or even zero-shot learning abilities, making them increasingly accessible for a wide range of use cases [53].

2.3.2 Conversational Artificial Intelligence: Overview of Tools and Platforms

Conversational Artificial Intelligence (AI) has evolved from simple rule-based dialogue systems to complex, language-aware platforms powered by Large Language Models (LLMs). These systems are designed to understand the natural language inputs, maintain a coherent dialogue over time, and respond in a way that it feels intuitive and human-like. A new generation of tools and systems has evolved to operationalize LLMs across industries and user scenarios as their size and capabilities have increased.

There are many examples of platforms that are used daily for various purposes:

- **Google Gemini:** It is the latest evolution in Google's AI infrastructure. It is developed by DeepMind and it is a multimodal LLM, which is able to understand and generate multiple types of input/output data such as code, images and structured data. It can support complex reasoning tasks and it is optimized for enterprise integration through Google Cloud and Vertex AI. Gemini also enables the development of assistants and agents that can perform nuanced tasks such as summarization, coding help, document parsing and as we are mentioning in the title, conversational interaction due to its rich contextual understanding.

- **Google Dialogflow:** It is another AI developed by Google, however it is serves with a different aim. It is a tool that has been designed specifically for structured conversational interfaces. Instead of relying on language models for dynamic generation, Dialogflow structures conversations by using intents, entities and context management. It is a tool that can be a good fit for use cases such as help desks, filling forms and guided troubleshooting, where precise answers and step-by-step support are more valuable than linguistic flexibility.

- **OpenAI ChatGPT:** It is a highly flexible and conversational platform. It has become the most noticeable example of general purpose conversational AI because of its abilities of human-like interaction, maintain context across turns and interactions with web browsing, plugins and file analysis. ChatGPT is designed around adaptability instead of pre-defined logic trees which gives it the flexibility and also making it suitable for personal assistants, creative writing, education and research support.

- **Anthropic Claude:** It is a model that mainly focuses on alignment and safety and consequently, introduced a safety-first idea to Conversational AI. Its cautious and explainable outputs make it especially appropriate for enterprise environments where safety, compliance and reasonings are critincal. It emphasizes alignment with human intentions, reducing hallucination and maintaining transparency in reasoning. It operates well in summarization, factual retrieval and high-level reasoning of the tasks.

- **DeepSeek:**It is a relatively newer LLM framework from China and it offers powerful language understanding and generation capabilities. It has trained with multilingual datasets with a focus on performance in both Chinese and English. It has been released in two primary forms of base models and instruction-tuned models. The base model is focused on general-purpose datasets to predict the next word in a sequence, learning grammar, syntax and factual associations. However it doesn't behave interactively and because of this reason, the base models followed a fine-tuning process as instruction tuning where they are trained on datasets consisting of prompts and responses. With this improvement, the model also gained the ability to better understand the user intent and generate more useful answers and made the model more conversational-friendly.

- **Copilot:** It is an AI-powered code assistant developed jointly by GitHub and OpenAI. It mainly assists developers by suggesting them contextually relevant code completions, generate functions and test cases so to reduce their routine workload and improving their productivity. Copilot's conversational interactions are mostly focused on code contexts by adapting developer's style and intent through interaction. It shows how LLMs are reshaping human-computer interfaces, especially in professional and domain-specific applications.

2.3.3 Benefits and Limitations of Conversational AI Technologies

Conversational AI technologies are transforming the way humans interact with machines. These systems are integrated into both consumer and enterprise environments thanks to their ability to generate human-like responses, utilizing natural language and providing context-aware assistance. On the other hand, conversational AI brings significant advantages but also introduces remarkable limitations and risks that must be carefully considered [53].

Benefits:

- **Scalability and Availability:** Conversational AI systems can handle thousands of simultaneous user interactions without any fatigue or delay, anywhere, during any time of the day. This scalability makes them ideal for information services, customer support and education where consistent and timely responses are crucial.

- **Accessibility and Inclusivity:** Conversational AI systems enable access to information and services for people who may face barriers with traditional interfaces such

as individuals with visual impairments, limited literacy or language barriers. Voice enabled assistants and multilingual models widen the accessibility to everyone.

- **Cost Reduction:** Automating repetitive tasks and routines with conversational AI systems can significantly reduce the operational costs for businesses and also for people in their daily lives.
- **Personalization:** Advanced models of conversational AI systems can generate responses based on user history, context and preferences. It makes interactions more meaningful, appealing and engaging. This is especially suitable for education, healthcare and e-commerce sectors.

Limitations:

- **Bias and Fairness Issues:** LLMs are trained on massive internet data, they can also reflect or even exaggerate societal biases in the training material. This reflects with ethical challenges in ensuring fairness, especially when the models are consulted in sensitive purposes such as recruitment, law or healthcare.
- **Inaccuracies:** One of the key technical limitations is the generation of incorrect or fabricated informations. This would sabotage the trust and may result in misinformation if it is not properly managed.
- **Lack of True Understanding:** Despite their potential, current models lack true comprehension and reasoning. Instead of using conscious interpretation, their outputs are based by statistical patterns, which may result in responses that are unclear or inappropriate for the setting.
- **Data Privacy and Security:** Conversational systems that handle personal or confidential data introduce privacy concerns. Improper handling of inputs or integration with insecure third-party systems may expose sensitive user information.
- **Dependence on Training Data:** The performance of conversational AI models is limited by the quality and scope of the data they are trained on. In domains with low digital representation, these systems may underperform or provide irrelevant information.
- **Over-Reliance and Misuse:** As conversational AI tools become more popular and universal, there is a growing concern about over-dependence on AI for tasks requiring human judgment, empathy, or ethical reasoning. Additionally, there is potential for misuse, such as generating fake content, impersonation, or misinformation

2.3.4 Ethical and Societal Implications of AI Systems

As artificial intelligence systems become increasingly integrated into daily life, their ethical and societal implications attracted attention from researchers, developers and policymakers. Conversational AI interacts directly with individuals and can influence decision-making processes. This direct interaction strengthen the idea of limitations that was

previously discussed. Issues such as bias, privacy risks, lack of transparency and misuse are not only technical challenges but also ethical since they shape decision-making, influence behavior, and may reinforce existing social inequalities [9, 39]. The ethical point of view of conversational AI extends beyond the system performance since it involves questions of value alignment, inclusivity and the prevention of harm. The European Commission's High-Level Expert group on AI has highlighted that, trustworthy AI must adhere to principles such as fairness, accountability, and transparency, especially when systems are deployed in sensitive domains like healthcare, education, or justice [13]. Regulating ethical standards at every stage of the AI lifecycle, from data collection to deployment, becomes more necessary as technology continues to improve and evolve. It is crucial to design these systems responsibly, apply clear regulations such as the EU AI Act and encourage cooperation between experts from different fields, to ensure that conversational AI respects human values and benefits society.

Chapter 3

Applications of BPMN and AI in Process Modeling

In this chapter, we explore how Business Process Model and Notation (BPMN) is applied in real-world contexts and how Artificial Intelligence (AI) is increasingly being employed to support and automate process modelling activities. With order we firstly explained the practical applications of BPMN across different industrial and service domains, followed by AI-assisted BPMN modelling and finally we have reviewed the previous academic studies and research trends to also motivate the analysis we have followed in the methodology chapter.

3.1 BPMN Applications in Practice

Business Process Model and Notation (BPMN) has become one of the most important standards for **documenting**, **analysing** and **automating** organizational workflows across industries. It has developed and maintained by the Object Management Group (OMG). The primary strength of BPMN is to **provide a shared visual language**, **unified graphical notation**, that bridges the gap between business stakeholders and technical developers. This fact enables both groups to describe processes in a structured and intuitive way [44].
This duality is precisely what makes BPMN an essential player of Business Process Management, which is a discipline that aims to analyse, optimize and automate processes to increase organizational **efficiency** and **transparency** [11].

The key strength of BPMN lies in its standardized vocabulary of elements such as events, activities, gateways and flows that allows complex processes to be expressed in a structured way to be interpreted consistently across different departments or organizations. In practical terms, BPMN acts as a bridge between business users, who focuses on the understanding of what the process does and also; IT Professionals, who are responsible for implementing it. Through its clear syntax and semantics, BPMN facilitates communication, reduces misinterpretation and enables the deployment of executable workflows

within process automation engines such as Camunda, Signavio, Bizagi, or IBM Blueworks Live.

3.1.1 Applications Across Sectors

The adaptability and flexibility of BPMN has made it a widely used standard across different industries and application areas [11, 44]. For instance, in the manufacturing industry, BPMN is used to model production sequences, material flow and quality control loops which is often used as a foundation for process simulation and automation. In service-oriented sectors, BPMN supports the optimization of customer interactions, from order placement sand fulfilment to feedback collection. Public administrations and regulatory institutions employ BPMN to model approval procedures, document workflows and compliance processes. As a consequence the traceability and transparency improves.In healthcare, BPMN diagrams help formalize complex procedures such as patient admission, medical examination and treatment pathways where clarity and coordination among different actors are essential. Similarly in logistics and transportation, BPMN is widely used to represent order management, delivery scheduling and customs clearance to help organizations identify bottlenecks and streamline international operations [11].
These examples show that BPMN is not limited to a single industry or process type but instead it provides a universal modelling framework that is capable of describing processes with varying complexity. Covering both human-driven collaborative workflows and fully automated operational processes. Its flexibility also allows it to be adapted to both routine and exceptional processes, and it enables organizations to model everyday administrative operations as well as high-risk, time-critical activities.

3.1.2 Role of BPMN in Digitalization and Process Automation

BPMN plays a central role in the context of digital transformation. As organizations shift toward data-driven management increasingly these days, BPMN provides the necessary **link between human decision-making and automated information systems**. The notation not only supports the documentation of current workflows but it also enables process execution and monitoring when it is integrated with Business Process Management Systems(BPMS). Through model driven architectures, BPMN diagrams can be automatically transformed into executable workflows where it is connected to real data streams and analysed to identify performance bottlenecks or compliance issues [62].

In this sense, BPMN acts as both a **design tool** and a **strategic instrument** for process optimization. It allows organizations to experiment with new process configurations, assess their impact on performance indicators and continuously improve operational efficiency. BPMN is also used as a reference model combined with modern process mining techniques, which real executing data can be compared and enabling data-informed redesign of business operations [54].

Artificial intelligence is assisting the development and improvement of BPMN models as digital technologies advance. Recent improvements in natural language processing, process mining and large language models have made it possible to generate BPMN diagrams automatically from textual process descriptions or event logs [5, 59].

AI-driven tools can also be used to determine what is lacking, make suggestions for improvements, and assess the rationality of current models. This development is part of a larger movement toward AI-assisted process modeling, where BPMN serves as an interface for intelligent systems that can reason, learn, and work with human modelers in addition to representing processes.

3.2 AI-Assisted BPMN Modelling

Artificial Intelligence (AI) is playing an expanding role in transforming how organizations design, analyze and optimize their business processes. Traditional process modelling, that was used to be a fully manual activity that requires an expert interpretation of textual descriptions or stakeholder interview, is now becoming a partially or fully automated through AI-driven tools. These systems combine techniques from natural language processing (NLP), process mining and machine learning to extract process information, detect relationships among activities and produce BPMN diagrams that are syntactically valid and semantically meaningful [11, 57, 59]. The introduction of AI into process modelling addresses one of the field's endless challenges: **The translating of unstructured knowledge into formal process representations.**

Business analysts typically describe processes in natural language, that can cause ambiguity, redundancy or domain-specific terminology [12]. AI models, especially those that are based on NLP and large-scale language architectures, can interpret these textual descriptions, identify processes entities such as tasks, events and gateways. As a consequence generate and initial BPMN structure automatically [29].

This automation does not only accelerates the modelling but also at the same time reduces the cognitive effort required from human experts, who can focus on validating and refining models rather than building them from scratch.

3.2.1 Integration of AI Technologies

Rule-based systems and process mining algorithms that recreated workflows from event logs were the foundation of early AI-assisted techniques. Although these methods are good at recognizing task sequences, their semantic comprehension is constrained. The recent emergence of Large Language Models (LLMs) represent a major leap forward. LLMs possess the capacity to reason over long textual contexts, identify process patterns and produce structured XML outputs that are compatible with BPMN standards. This capability allows them to act as intelligent assistants that are capable of generating, explaining or correcting BPMN models through conversational interfaces.

For instance, in the case that a user provides an unstructured textual description which outlines the sequence of actions, decisions, and interactions within a process; the model can generate a syntactically coherent BPMN diagram that identifies the relevant events, actors, and gateways. In more complex processes that involve multiple participants or simultaneous operations, AI systems can represent interdependent tasks and communication flows that would typically need human interpretation. This illustrates how AI-driven modeling may successfully support both human-oriented and system-oriented workflows, facilitating the shift from informal descriptions to standardized process representations.

3.2.2 Benefits and Opportunities

The adoption of artificial intelligence in BPMN modelling offers several concrete advantages that enhances both efficiency and reliability in process design. These benefits are; **Reduction of Modelling Time** where AI significantly accelerates the process of diagram creation by identifying the activities, events and decision points automatically from textual descriptions. This allows for rapid prototyping and iterative refinement of process ideas and reduces the manual workload traditionally required from human modelers. **Improved Accessibility**, because through natural language interaction, AI systems make BPMN modelling more accessible to non-expert, daily users. Individuals without technical backgrounds can describe processes in plain language and obtain coherent BPMN diagrams while broadening participation in process design and fostering organizational inclusivity. **Enhanced Consistency and Scalability** because models that are generated by AI tools follow standardized structures and formatting conventions and reducing the variability introduced by human subjectivity. This uniformity supports large-scale process documentation and ensures that multiple models across an organization to maintain a consistent level of quality and interpretation. **Error Detection and Optimization** because AI-assisted systems can automatically identify logical inconsistencies, redundant elements or missing connection within BPMN diagrams.

By flagging structural errors and suggesting improvements, these tools contribute to higher model quality and more robust process automation readiness.

3.2.3 Challenges and Limitations

Despite these advantages, current AI-based BPMN modelling still faces limitation problems. For example LLMs, may produce diagrams that are syntactically correct but semantically inconsistent, with misplaced gateways or events due to lack of domain context. Moreover, since AI outputs are highly dependent on prompt structure and input quality, small variations in the wording can lead to significantly different models. The probabilistic nature of model generation also introduces variability and reproducibility issues which make consistent evaluation difficult. Finally, concerns about how transparent and explainable AI-generated models remain important since the users must understand the logic behind these models when they are used in real business contexts [57].

In summary, AI-assisted BPMN modelling represents a rapidly evolving research area that **blends linguistic intelligence with process-engineering expertise**. It

promises faster and more accessible process design but it still requires systematic evaluation to determine **accuracy**, **reliability** and **domain adaptability**.

3.3 Previous Studies and Research Trends

Recent academic works have examined how artificial intelligence can enhance or automate the process modelling activities. While the previous section outlined the conceptual role of AI in BPMN generation, in this part we mostly focus on existing studies and methodological developments. The literature can be categorized into three research directions:

- **Data-Driven Process Discovery**
- **Natural Language-based Model Generation**
- **Large Language Model-assisted BPMN Automation**

Together these studies demonstrate the growing integration of AI in Business Process Management, yet they also reveal clear gaps in evaluation consistency and methodological robustness.

3.3.1 AI and Process Mining

Process mining represents one of the earliest intersections between artificial intelligence and Business Process Management, that focuses on reconstructing workflows from digital event logs [54].

These methods use data-driven algorithms to interpret process models by identifying sequences and dependencies between recorded events. Over time, AI and machine learning techniques have been incorporated into process mining to **improve prediction**, **anomaly detection** and **decision support** [14].

Even if process mining offers valuable insights into how processes are actually executed, it relies on structured event data and therefore it does not address conceptual modelling or translation of textual knowledge into BPMN diagrams. For this reason, it serves as an important precursor to AI-based BPMN generation but not as a direct competitor.

3.3.2 Natural Language Processing for BPMN Generation

A second research direction applies Natural Language Processing (NLP) to extract process information directly from textual descriptions. Early frameworks, employed syntactic and semantic rules to identify activities, actors and dependencies converting these linguistic features into BPMN constructs. While it was effective for small, well-structured texts, rule-based systems lacked generalization and scalability.

Recent studies have replaced deterministic parsing with neural and transformer-based architectures, that are capable of learning process structures from larger and more diverse textual datasets [2].

This evolution has made it possible to handle domain-specific terminology, ambiguous instructions and longer contextual dependencies that paves the way for more autonomous forms of BPMN generation. However, these approaches still depend heavily on high-quality training data and often require a post-processing to correct syntactic or logical inconsistencies.

3.3.3 Large Language Models and BPMN Automation

The introduction of Large Language Models has opened a new phase of research into AI-assisted process modelling. Unlike earlier NLP methods, LLMs can interpret complex instructions, reason across sentences and generate structured outputs in BPMN-compatible XML. Several recent studies [26, 32, 57] conceptualized large language models as intelligent process-design assistants that can automatically generate, refine and optimize BPMN diagrams.

LLMs can also integrate feedback loops through **prompt refinement** or **user interaction** to progressively **improve model accuracy**. Still, the research so far has focused on individual model evaluations or isolated case studies, also often in specific domains such as healthcare, logistics or education.

Only a few studies have attempted to compare multiple models or apply quantitative methods to validate results, which limits the generalizability of current findings.

3.3.4 Evaluation Frameworks and Methodological Gaps

One of the key challenges in this research field is the lack of standardized evaluation criteria. Different studies use heterogenous metrics that ranges from syntactic validity to readability or subjective accuracy that hinders cross-comparison [41].

To improve the methodological consistency, several frameworks have been proposed for evaluating AI-generated process models. These frameworks emphasize reproducibility and prompt design control, as well as alternative methods focusing on syntactic correctness, semantic adequacy, or expert-based validation [32, 56].

While these frameworks represent a step toward standardization and methodological consistency, most existing studiers still rely on qualitative or expert-based evaluations and often lack in quantitative validation to justify their conclusions. This ongoing limitations emphasizes the need for mixed-method evaluation approaches that integrate human judgement, structured scoring system and statistical verification, which is in the end an approach that is adopted in this study [51, 56].

3.3.5 Research Gap and Contribution of this Study

Despite the literature's clear progress in combining AI and BPMN, there are still a number of gaps:

- **Comparative analyses across multiple AI systems are rare, most studies examine only a single model or prompt configuration**
- **Evaluations often lack a diverse set of process domains, reducing the applicability of results to real-world BPMN scenarios**
- **Only few works include quantitative or statistical validation, leaving many findings in a descriptive level**

Building upon these observations, the current study uses a methodical and comparative approach to address the limitations in earlier researches. This study evaluates multiple AI systems across a diverse set of BPMN exercises that represent different organizational sectors and process types. To be able to assure methodological accuracy, We have employed a structured scoring mechanism, backed by quantitative validation, which will be covered in further depth in the upcoming chapter.

Chapter 4

Methodology

This chapter describes the used methodological framework to evaluate the capability of AI systems in generating BPMN diagrams. It explains the structure of the study, including the preparation of the dataset, the definition of the scoring systems, the experimental setup for testing different AI models, and the statistical methods used for analysis.

4.1 Study Framework and Dataset Preparation

To explore the capabilities of artificial intelligence in BPMN diagram generation, a structured **dataset of modelling exercises** was gathered as the foundation of this study with a goal of creating a diverse collection of scenarios. To better interpret the difficulty of exercises, we have introduced a custom **scoring system.**

4.1.1 Exercises

As mentioned earlier; the study began with the premise that evaluating the performance of artificial intelligence in BPMN diagram generation requires a **structured, meaningful, and diverse dataset of exercises**. Instead of relying on precompiled or standardized collections, a set of 16 BPMN exercises were gathered manually through online research and publicly available sources.

Each exercise was selected to reflect a different conceptual layer of process modelling, ensuring variation of conceptual and technical depth, to evaluate the capabilities of AI models in generating high-quality BPMN diagrams. The goal was to assemble a dataset which was also a representative of **real-life** scenarios.

The search process involved exploration of online academic resources, BPMN training platforms, educational blogs and community forums. When choosing the exercises, attention was paid to:

- **Complete Problem Description:** Exercises needed to include a sufficient description to enable BPMN diagram creation without interpretative assumptions. To guarantee clarity in the evaluation process, exercises with imprecise specifications or unclear modeling assumptions were eliminated.

- **Realistic Scenario:** The scenarios that were selected had a variety of domains such as retail, logistics, healthcare to allow better relatability to daily examples.

- **Well-Defined Solution:** Each exercise had a correct and valid BPMN solution that is provided directly with the exercise itself. It had also been used as a reference during the evaluation phase.

- **Coverage of BPMN Elements:** The dataset included tasks that require use of various core BPMN elements such as gateways, pools, messages. This made it possible to evaluate AI performance over the entire range of the notation.

The selected exercises reflect a wide range of conceptual and technical modelling challenges. Some of them have a basic process sequence and decision-making using gateways, some of them incorporate more advanced elements such as message flows, lane/pool interactions. This diversity allowed the evaluation to test the AI model's ability across different levels of BPMN complexity.

The following figure 4.1 provides an example of one of the BPMN solution, Hiring Process [23], also used for our evaluation:

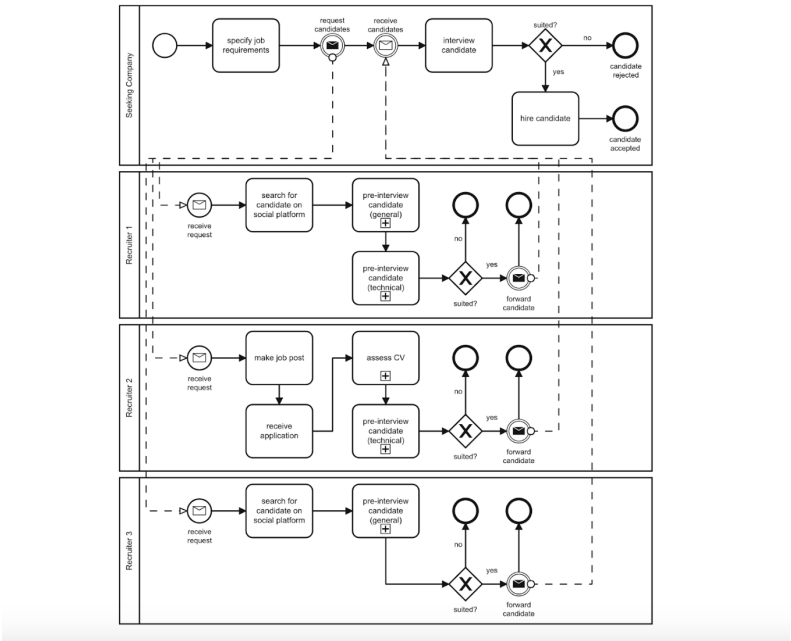

Figure 4.1. BPMN Exercise Solution Example, from Gdowska et al. [23].

Diversity of BPMN Scenarios in this Study

The BPMN exercises used in this study reflects this diversity of real-world applications [11, 44]. The dataset of 15 different processes was deliberately designed to contain multiple sectors and process categories, ensuring that the evaluation of AI-generated BPMN diagrams are not biased towards a specific domain or workflow structure [59].This scenarios include:

- **Service and Administrative Processes:**

  Table Service Collaboration, Signature Regulation, and Hiring Process. It emphasises coordination among human actors and decision-making steps.

- **Operational and Project-Oriented Processes:**

  Project Implementation, Job Scheduling, and Maintenance. It involves structured task sequencing, dependencies and feedback loops.

- **Industrial and Production Contexts:**

  Mine Blasting Process and Bill of Materials. It mostly focuses on material flow, safety controls and process dependencies.

- **Logistics and Supply-Chain Processes:**

  Placing an Order and International Transport of Goods. It highlights interactions between multiple organizations and cross-border activities.

- **Healthcare and Emergency Managements:**

  Hospital Treatment Process and Fire Extinguishing Collaboration Model. In these kind of cases, timing, coordination and event handling are critical to success.

- **Support and Travel-Related Processes:**

  Business Trip. It involves approval, resources allocation and repeating workflows.

This sectoral variety mirrors how BPMN is used in practice across industries, functions and organizational levels. It also strengthens the evaluation presented in this thesis by also exposing AI models to different modelling challenges such as:

- **Structured vs Unstructured Descriptions**
- **Human vs Automated Tasks**
- **Sequential vs Collaborative Activities**

By doing so, the dataset captures the heterogeneity of BPMN applications in contemporary organizations and provides a realistic test base for AI-based process modeling.

4.1.2 Scoring System

To ensure a systematic and objective classification of BPMN exercises according to their level of difficulty, we have created a unique scoring system. The aim of this system is to evaluate each exercise with a predefined set of constraints that represent both structural and semantic elements of process modelling.

By using this system, the qualitative indicators such as ambiguity, context and diagram size can be translated into a quantifiable score that allows consistent classification across all of the exercises.

Each exercise is evaluated using nine different constraints and each constraint is scored on a scale from 1 to 5, with 1 representing a low level of complexity and 5 representing very high complexity.

The final scores are also determines the difficulty level of the exercises, that is grouped into five different levels from very easy to very hard. This structured classification supports a more fair benchmarking of the generated solutions. The constraints and scoring criteria for BPMN exercise complexity evaluation are shown in Table 4.1.

To translate the cumulative scores from the evaluation we have concluded using the constraints, into an intuitive difficulty level, we have defined five categories:

- **Level 1-Very Easy (9-14 points):** Very short, clearly worded exercises with few BPMN elements. It usually involves a single pool, no loops or message flows, and minimal ambiguity.
- **Level 2-Easy (15-20 points):** It is slightly more complex than Level 1. It uses basic gateways, 1-2 pools, small diagram sizes, and limited interpretative effort required
- **Level 3-Medium (21-28 points):** It is a moderate-sized diagram including loops or message flows. It has some ambiguity or it can require interpretive assumption. It involves 2-3 pools and a wider range of BPMN elements.
- **Level 4-Hard (29-36 points):** There are more pools and lanes, more intricate logic, and a wider range of element kinds in this rather larger diagram. Moderate domain expertise can be necessary.
- **Level 5-Very Hard (37-45 points):** This level of exercise includes extremely complicated structures like several pools, nested loops, domain-specific vocabulary, sophisticated event handling, and considerable ambiguity. Significant modeling ability and contextual knowledge are required for these assignments.

After the scoring criteria and the difficulty levels established, the scoring system was applied to all of the exercises in the dataset. Each exercise was individually evaluated

Constraint	Description	1	2	3	4	5
Length of the exercise	Number of words in the textual description. Longer texts increase cognitive load and scenario complexity.	≤150	151–200	201–250	251–300	≥301
Number of BPMN Element Types	Variety of distinct BPMN elements (tasks, events, gateways, pools, subprocesses) needed to model the scenario.	1–2	3	4	5	≥6
Number of Gateways	Decision points in the process. More gateways increase control flow complexity.	0–1	2	3–4	5–6	≥7
Clarity of Wording	Degree of explicitness in the exercise text. Ambiguity increases modeling difficulty.	Fully explicit	Minor unclear point	Several implicit steps	Conflicting or missing info	Highly ambiguous
Domain Knowledge Needed	Amount of external knowledge required to understand the scenario.	Generic process	Simple scenario	Common business	Technical domain	Specialized domain
Expected Diagram Size	Total number of BPMN elements; larger diagrams imply greater complexity.	≤5	6–8	9–12	13–15	≥16
Number of Pools & Lanes	Number of participants or departments represented.	1	2	3	4	≥5
Cross-Boundary Interactions	Message flows across pools; more interactions increase coordination complexity.	None	1	2–3	4–5	≥6
Loops & Iterations	Presence of repeating or cyclic behavior in the process.	None	Simple loop	Structured loop	Nested loops	Complex or event-based loops

Table 4.1. Constraints and scoring criteria for BPMN exercise complexity evaluation.

across defined constraints and it resulted in a total score that reflects its overall complexity. This categorization is not only supported consistency during evaluation but also helped the understanding of how AI models perform across varying degrees of modelling difficulty. The analysis provide an objective and structured way to observe how AI performs across varying levels of complexity, revealing both its strengths and weaknesses in handling different structural and contextual challenges.

4.2 Prompt Design

The success of Large Language Models depends heavily on how the user formulates the prompts, which is also known as **prompt engineering**. Instead of simply asking a question, it requires to create inputs that guide the model to producing accurate, structured and relevant outputs particularly in domain-specific tasks like process modelling [47].

Prompt engineering can be considered as an alternative to fine-tuning, allowing pretrained models to be adapted for new tasks through input design without retraining the model itself [34]. As mentioned before, a prompt is more than a question. It is a structured instruction that sets context, format and expectations for the model's response. Prompts relies on **specificity**, **clear formatting**, **step-wise decomposition** and **iterative refinement** [58, 63].

In this study, prompt engineering played a crucial role evaluating the capabilities of different LLMs to generate accurate BPMN diagrams. One of the challenges was also to assess whether the models could understand a textual description but also if they could translate it into a **syntactically valid** and **logically correct** BPMN diagram in XML format compatible with Camunda Modeler.

This choice was not coincidental; during the early testing phases, it became evident that when AI models were asked to describe diagrams textually or generate them using natural language instructions, the results were incomplete, imprecise, or incorrect in structure. By explicitly requesting the output in **XML format**, the prompts helped to translate the response to a machine-readable structure and significantly improved the quality and consistency of the generated diagrams. Additionally, having XML as an output allowed the results to be directly validated and visualized with **Camunda Modeler** which ensured a more fair comparison and consistent evaluation across exercises.

To fully assess the capabilities of LLMs in BPMN generation, we have used four different AI systems in this study through their official web-based chat interfaces:

- **ChatGPT**
- **Copilot**
- **Gemini**
- **Deepseek**

These models were chosen based on their broad accessibility, high performance in structured tasks and their potential integration into real-world cases. Each model was prompted with the same problem descriptions to ensure a uniform evaluation framework.

The study initiated by using a **zero-shot prompting** approach, where the AI receives the task without any **prior examples** or **contextual training**. This strategy reflects realistic user scenarios where individuals typically expect a system or to understand and perform based on a single prompt. However, it became evident that with zero-shot prompts, it is insufficient to generate accurate diagrams [34].

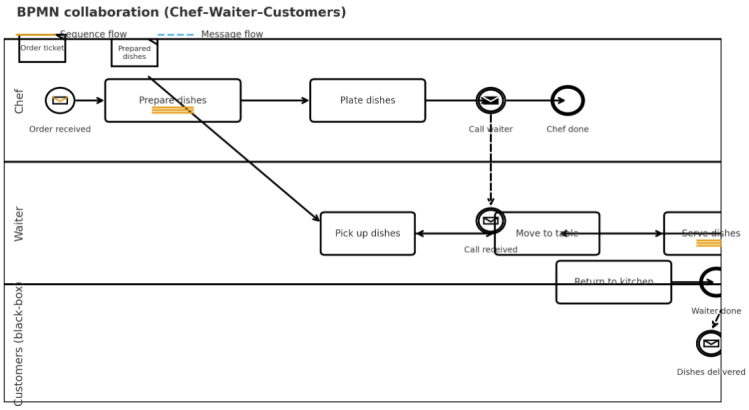

Figure 4.2. Zero-shot Prompting by ChatGPT

As shown in Figure 4.2, the ChatGPT generated BPMN diagram lacked structural completeness and semantic precision. While the main activities such as “Receive order”, “Prepare dishes”, and “Call waiter” were correctly identified, the model failed to capture the full interaction among all participants — particularly the message exchanges between the waiter and the customers. The outcome was syntactically correct but conceptually incomplete, indicating that the zero-shot prompting approach was insufficient for producing a coherent and contextually accurate representation of multi-pool workflows.

To overcome these limitations, the prompting methodology was refined iteratively. Rather than providing examples of correct solutions as it would have been done in few-shot prompting, the process involved giving **targeted feedbacks** based on the output of the previous response [63]. When a model returned as an incomplete or invalid diagram, new prompts were issued with precise instructions that corrected logical errors, clarified misunderstood process elements, or restructured malformed XML.

This iterative interaction helped to lead the model to produce **more coherent** and **valid** diagrams both logically and structurally. It has also reflected how each model adapted to incremental improvements. This feedback-driven refinement process enabled a fair comparison of the models’ capabilities.

4.2.1 The COPE Methodology: Contextualizing Prompt-Based Evaluation

As LLMs become increasingly integrated into domain-specific tasks such as business process modelling, structured evaluation frameworks are necessary to ensure consistent and interpretable assessments of their outputs. In this study we have used **COPE** methodology which stands for **Context**, **Objective**, **Prompt** and **Evaluation**. It has originally proposed to systematically analyze the behaviour of LLMs response to user inputs, provide a logical and repeatable structure for testing how different prompting strategies affect performance [7].

Each exercise followed the four stages of the COPE framework:

- **Define Context:** The AI model had sufficient background knowledge to structure the process logic because the relevant scenario was clearly described. The context was kept as close as possible to the **original question** without introducing assumptions, in order to test the model's capacity to reason from limited inputs.

- **Define Objective:** The expected output was defined explicitly in such way that it is syntactically valid, logical BPMN diagram represented in XML format, compatible with Camunda modeler. The goal was not only correctness in logic but also compatibility with BPMN syntax, enabling automated validation of the results.

- **Design Prompt:** The prompts used in this study were carefully created following established strategies from the field of prompt engineering. They were kept short and written **clearly** to avoid confusion. They were focused on the BPMN domain. Each prompt is directly asked the AI to generate a BPMN diagram in **XML** format so that it can be tested in the Camunda Modeler as it has been explained in the beginning of this section. If the AI misunderstood the task or produced incorrect results, the prompts were **revised and improved** step by step with the help of relative **feedbacks**, helping to guide the AI to **more accurat**e responses [34,47,58].

- **Evaluate Output:** Structural validity, in sense of whether the XML file could run and visualised, was tested via import into Camunda Modeler. The logical correctness was reviewed briefly with the reference diagram, with key factors such as task order, gateway usage, pool/lane consistency, and message flows taken into account.

Unlike approaches based on few-shot prompting or retraining, the COPE methodology enabled a **systematic evaluation of zero-shot outputs** and the **effectiveness of iteration** over time [58,63].

The use of COPE method brought a methodological perspective to the evaluation process and at the same time also contributed to have a **more reproducible and transparent experimentation protocol**. It has supported the identification of patterns across tasks, such as if certain BPMN elements had some consistent errors with the AI-generated outputs or if specific prompt formulations led to **higher XML validity rates.**

4.3 Evaluation Methodology

A crucial component of this study was establishing an **accurate and fair methodology** for evaluating the BPMN diagrams generated by using large language models and comparing them to the answers provided with the question itself. It was necessary to adopt a well-defined set of evaluation criteria to assess whether the different artificial intelligence systems can generate diagrams that are comparable in **accuracy, structure and clarity** to the answers.

After finalizing the prompt engineering strategy using also COPE framework, an online Google Drive was created for this thesis to store all the materials such as the question itself, the **PDF** and **.bpmn** version of the created BPMN diagram answers by AIs. The complete repository is publicly accessible and referenced in the Appendix at the end of this thesis.

Following the preparation phases, the generation of BPMN diagrams was conducted for each exercise **one-by-one** using the four selected LLM architectures.

For each exercises, the prompt was carefully formulated and submitted to the selected AI model. The output returned in XML format was saved and opened in Camunda Modeler, which ensured a consistent modelling interface for **visual inspection** and **standardized rendering** of each diagram. Where necessary, minimal manual corrections were made just to convert structurally correct diagrams into more readable diagrams, without altering the logic or structure of the original AI output.

To support a **systematic** and **objective** comparison between the generated diagrams and the expected solutions, a **custom scoring system** was developed. The scoring system which we will be explaining more detailed in the following sub-section allowed for a **multi-criteria assessment of each diagram.**

Finally, the evaluation methodology integrated a **manual comparison process**, in which AI-generated BPMN diagram was assessed using developed evaluation scoring system and benchmarked against the corresponding reference solution. This approach provided a structured means of quantifying the performance of each model and of identifying the areas of strength and limitation across different type of exercises.

4.3.1 Scoring System for Diagram Evaluation

To ensure a systematic and objective comparison among the "given answer" and diagrams generated by different LLMs, a custom scoring system was developed for this study. The scoring system is **comprehensive**, because it is a **multi-criteria** evaluation that captures the full quality spectrum of a process model. On the other hand it is also **replicable**, because it ensures that another researcher can use the same dataset and criteria and would obtain consistent results regardless of subjective interpretation.

To achieve this, all of the evaluation metrics were defined using precise, measurable

indicators and a uniform numerical scale from **0 to 5**. With this, there is a balanced comparison across diagrams generated by ChatGPT-5, Copilot, Gemini and DeepSeek even if each of these models operates under different architectures, training data and reasoning capabilities as we have briefly mentioned in the introduction chapter.

Establishing such a unified framework was essential to normalize performance differences among models and to provide a reliable basis for comparing their outputs against the reference BPMN solution.

Furthermore, the design emphasized **traceability** and **transparency**, meaning that every assigned score could be justified by a clear rule or observation derived from the BPMN 2.0 specification or from recognized process model quality literature. This approach reduced the risk of subjective bias and allowed the evaluation process to serve not only as a performance comparison but also as an analytical tool to identify where and how specific AI systems tend to make structural, syntactic, or interpretive mistakes in process modeling.

The scoring framework was inspired by several academic contributions on process model quality metrics:

- Gruhn & Laue (2006), who proposed core metrics for syntactic and semantic correctness [27].
- Mendling et al. (2010), who emphasized the impact of model understandability and layout readability [37].

The aim was not only to assess syntactic correctness but also to **evaluate the logical consistency and modelling clarity,** taking into account the types of errors commonly made by LLMs.

4.3.2 Structure of Scoring System

The scoring system was divided into five main categories, which all address a different aspect of BPMN quality: **Completeness**, **Correctness**, **BPMN Syntax**, **Style & Clarity** and **Similarity to Reference**.

Each category contains a set of constraints that is each graded on a scale from 0 to 5. The constraints were developed iteratively, with the help and inspiration from literature as well as qualitative insights gathered during creating of AI-generated diagrams.

To reflect the relative importance of different modeling dimensions, a **weighted system** was applied to the constraints. Specifically, **Completeness 20%** of the total score, **Correctness for 25%**, **BPMN Syntax for 15%**, **Style & Clarity for 15%**, and **Similarity to Reference for 25%**.

These weights were based on insights from literature and practical evaluations of previous works. Particularly, the high importance is assigned to Correctness and Similarity

to reference, which is also consistent with Dumas et al. [11] , who argue that execution **feasibility and behavioural equivalence are primary concerns** in model validation. The emphasis on Completeness reflects the need to **fully capture all required process steps and elements** as highlighted in Mendling et al. [38].

Each constraint's raw score was multiplied by its respected category weight, which produces a normalized score between 0 and 5. The weighted scores then **summed up** to calculate the final total score for each AI generated diagram, with a **maximum of 16 points**.

4.3.3 Explanation of Categories and Constraints

Each category and constraint was designed to isolate particular aspects of process model quality. It ranges from basic syntactic correctness to deeper functional equivalence. With this evaluation, not only we had a fairer comparison between AI outputs and reference solutions but also it leads us to a conclusion of strengths and weaknesses about different AI models, that we will deeply examine during the conclusion chapter.

Hence, each constraint is explained in detail including the reasoning behind its inclusion, how it contributes to overall model quality and how different scores from 0 to 5 were interpreted and assigned based on observable features in the generated diagrams.

1. **Completeness**

   It represents the degree to which a BPMN diagram captures all the essential components of the described process.

   It assesses if the AI generated models are able to include every relevant element required to represent the process described in the exercise. All of the activities, gateways and message or data flows that define the process from start to finish must be included in a complete diagram. Completeness criterion ensures that the model provides a fully traceable and meaningful workflow which can also avoid the gaps that could compromise its interpretability or execution logic.

   For completeness the subcriteria and their reasoning and scoring are introduced as:

   - **Activities Included:**
     This constraint evaluates if all the activies and tasks that appear in the reference solution are present in the AI generated solution.

     Possible question to ask: Are all tasks and activities that appear in the reference model present?

     Scores are distributed as:
     - 0 No relevant activities are present.

  - 1 Only a few relevant activities are included.
  - 2 Many core activities are missing or overly abstract.
  - 3 Some essential activities are missing or replaced with vague tasks.
  - 4 One minor activity is missing or inaccurately represented.
  - 5 All activities from the reference model are correctly included.

- **Gateways Used Properly:**
  Gateways are crucial to define decision points and parallel executions in BPMN. With this constraint it is possible to examine if the gateways are correctly representing the logic, synchronization and conditional paths. Misused or missing gateways may disrupt the process flow and lead to logical inconsistencies.

  Possible question to ask: Are decisions or parallel flows represented with the correct type of gateway?

  Scores are distributed as:

  - 0 All activities from the reference model are correctly included.
  - 1 Gateways used arbitrarily or inconsistently.
  - 2 Incorrect usage in most cases, logic is hard to follow.
  - 3 Multiple gateways used incorrectly, but overall logic is still interpretable.
  - 4 One gateway is misused or overly simplified.
  - 5 All decision/parallel flows use appropriate gateway types correctly.

- **Start/End Events:**
  Every process must begin and end with explicit events to ensure structural and logical closure. This constraint checks whether both start and end events are correctly placed and semantically appropriate. Missing or misplaced events indicate an incomplete or ambiguous workflow definition.

  Possible question to ask: Are both start and end events included?

  Scores are distributed as:

  - 0 No start or end events included.
  - 1 Only implicit start/end exists, poorly represented.
  - 2 Start and end events both misused or wrongly placed.
  - 3 Either start or end is missing or misused.
  - 4 Minor deviation (e.g., an extra end event with no impact).
  - 5 Correct use of one start and one or more end events, as per BPMN norms.

- **Data/Message Present:**
  This constraint focuses on whether data and message flows are represented wherever communication or information exchange is described in the problem statement. Including data and message events reflects the model's capacity to capture information exchange and inter-process interactions. Which are essential for representing the real-world process logic.

  Possible question to ask: If the process includes data exchange, are data/message objects shown?

  Scores are distributed as:
  - 0 No data or message flows included, despite need.
  - 1 Data/message elements mentioned but not correctly implemented.
  - 2 Data/message used incorrectly or confused with other symbols.
  - 3 Several elements missing, though intent is somewhat clear.
  - 4 One element missing or improperly attached.
  - 5 All data and message elements are present and correctly linked.

2. **Correctness**

It focuses on the logical soundness of a BPMN diagram and its ability to represent a valid and executable process model. It addresses how the necessary component interacts, if the flow is logically coherent and if the diagram can be expected to behave as intended during execution.

For correctness the subcriteria and their reasoning and scoring are introduced as:

- **Logical Flow:**
  It evaluates the clarity and consistency of the sequence flow from the start to the end event. A logically flowing diagram ensures that each activity leads to the next by forming a connected and interpretable path. Logical soundness is foundational to model interpretability and aligns with recommendations in prior work such as Mendling et al. (2010), which stresses that poor flow structure severely impairs comprehension and analysis of process behavior .

  Possible question to ask: Does the process flow logically from start to end without breaks or dead ends?

  Scores are distributed as:
  - 0 No logical structure can be inferred; flow is chaotic or disconnected.
  - 1 The logic is mostly broken; the model lacks coherent start-to-end progression.

  - 2 Multiple confusing or contradictory paths; logical sequence is hard to follow.
  - 3 Noticeable issues in logic (e.g., unclear task order, mild redundancy) but still interpretable.
  - 4 The process is mostly logical with minor issues (e.g., one unclear transition), but overall understandable.
  - 5 The process flows logically from start to end without any gaps, redundant loops, or inconsistencies.

- **Gateways Lead to Valid Outcomes:**
  It evaluates the correct application of gateways to direct the process based on decisions or concurrent paths. Properly configured gateways should preserve logical coherence by ensuring that every split has a corresponding and syntactically correct merge.

  Possible question to ask: Do gateways properly split and merge flows?

  Scores are distributed as:
  - 0 Gateways are absent, misused throughout, or lead to entirely invalid outcomes.
  - 1 Gateway usage severely disrupts process logic; most branches are flawed.
  - 2 Gateways frequently lead to incorrect, missing, or contradictory outcomes.
  - 3 Some incorrect or unclear gateway usage; some branches might be misinterpreted.
  - 4 Minor issues in gateway logic (e.g., wrong type used once), but model behavior remains valid.
  - 5 All gateways split and merge correctly, producing expected and valid paths.

- **Error-Free Token Flow :**
  This constraint considers the execution of the BPMN model using token game semantics, where tokens represent control flow in process execution. The goal is to identify any structural flaws that would result in errors, infinite loops or incomplete terminations.

  Possible question to ask: Would the model execute without token-based errors?

  Scores are distributed as:
  - 0 The model fails completely in terms of token execution (deadlocks, infinite loops, etc.).
  - 1 Most paths are token-infeasible; process likely fails in execution.

- 2 Several paths have critical token flow issues; model needs repair to work.
- 3 Some parts risk token mismanagement (e.g., tokens stuck or multiply flowing).
- 4 Slight risks of token misbehavior, but process would still execute reliably.
- 5 The model supports seamless token execution; no deadlocks, duplication, or loss.

3. **BPMN Syntax**

This category assesses if the BPMN diagrams follow the BPMN standard's syntactic criteria. While completeness and correctness focus on the logic and content of the process, This guarantees that the diagram uses the appropriate symbols and connections also at the same time label the elements in a readable and standard way. Syntactic errors may not necessarily prevent understanding but they can significantly reduce the diagram's readability and make it less user-friendly.

For BPMN Syntax the subcriteria and their reasoning and scoring are introduced as:

- **Correct Use of Shapes:**
  This constraint checks if the graphical elements such as tasks, start/end events, gateways are represented with the appropriate and standardized symbols.

  Possible question to ask: Are tasks, events, gateways used with the correct symbols?

  Scores are distributed as:
  - 0 BPMN shapes are absent or completely misused, making the diagram syntactically invalid.
  - 1 Most BPMN shapes are used incorrectly, leading to confusion or invalid process representation.
  - 2 Frequent misuse of BPMN elements; interpretation becomes difficult and may cause execution issues.
  - 3 Some incorrect or inconsistent shape use; process remains interpretable but partially noncompliant.
  - 4 Almost all shapes are correctly used; a few minor misuses (e.g., wrong event marker) that do not affect process meaning.
  - 5 All BPMN elements (tasks, events, gateways, etc.) are correctly used according to BPMN standards. Each shape type corresponds accurately to its intended function.

- **Connections Valid(No Semantic Violations):**
  It ensures that the sequence flows are drawn exclusively between valid BPMN elements that are allowed to be connected.

  Possible question to ask: Are sequence flows used only between valid elements?

  Scores are distributed as:
  - 0 Connections are entirely missing, illogical, or violate BPMN flow rules completely.
  - 1 The majority of connections are invalid or misleading, compromising process integrity.
  - 2 Multiple invalid connections leading to ambiguous or incorrect flow interpretation.
  - 3 Some connections are semantically invalid (e.g., message flow between tasks instead of pools), but overall flow is understandable.
  - 4 Minor connection issues (e.g., redundant flow or small ordering error) but the diagram remains semantically valid.
  - 5 All sequence flows and message flows are valid and logically consistent (e.g., flow from task -> gateway -> event is correct). No semantic violations.

- **Labels and Annotations:**
  This constraint evaluates the naming and description of the elements. Labels should clearly reflect the meaning of tasks and elements. At the same time annotations should improve the clarity of the diagram without causing confusion.

  Possible question to ask: Are elements properly labeled? Are any annotations used correctly?

  Scores are distributed as:
  - 0 No labels or annotations are provided, making process elements indistinguishable.
  - 1 Labels and annotations are mostly incorrect, vague, or irrelevant.
  - 2 Many labels are missing or misleading; annotations are confusing or incomplete.
  - 3 Several labels are missing, inconsistent, or unclear, slightly affecting readability.
  - 4 Labels are mostly clear and consistent; few missing or redundant annotations.
  - 5 All elements are clearly and accurately labeled. Annotations are meaningful, concise, and enhance understanding of the process.

4. **Style & Clarity**

   This category evaluates more on the visual side. It evaluates how visually readable, consistently named and clearly structured the process diagram is. It does not directly affect the functional execution of the model however these aspects are greatly influence user understanding, maintainability and model usability. A well-presented BPMN model ensures that the readers, either they are on the technical team or the stakeholder, can easily understand, comprehend and navigate the process without any confusion.

   For Style & Clarity the subcriteria and their reasoning and scoring are introduced as:

   - **Naming Consistency:**
     This constraint emphasizes the importance of consistent naming across similar tasks and elements helps maintain clarity and reduces ambiguity in BPMN diagrams. Readers may become confused by ambiguous task names that are inconsistent such as alternation between "Approve Request" and "Request Approval", especially when it is combined with inconsistent use of the verbs, objects and capitalization patterns. By switching to a more systematic naming format, the models become easier to read, analyse and validate.

     Possible question to ask: Are similar types of tasks labeled in a consistent and descriptive manner?

     Scores are distributed as:
     - 0 Most or all BPMN elements are unnamed, making the process very difficult to follow.
     - 1 Naming is highly inconsistent or misleading; readability and understanding are impaired.
     - 2 Many elements are either unnamed, use unclear labels, or show conflicting naming patterns.
     - 3 Some tasks or events are vaguely named or inconsistently formatted. Occasional ambiguity present.
     - 4 Minor inconsistencies in naming or formatting (e.g., some tasks lack verbs), but the meaning is still clear.
     - 5 All activities, events, and gateways are named consistently, following a clear and structured naming convention (e.g., verb-object format). No duplicate or ambiguous names.

   - **Layout Readability:**
     The visual layout of a BPMN diagram significantly influences its readability and its quality. A well-aligned and spaced model allows users to follow the process flow intuitively, without needing to change the position of the elements

or the direction of the arrows. On the other hand; misaligned symbols, overlapping flows or dense clustering of elements create a visual noise and disrupt understanding even when the syntax is formally correct.

Possible question to ask: Is the diagram visually clean, aligned, and non-overlapping?

Scores are distributed as:

- 0 Diagram layout is chaotic or broken, preventing any clear reading of the process.
- 1 Most of the diagram is difficult to read due to poor spacing, flow direction, or alignment.
- 2 Layout is confusing or disorganized. Overlapping elements and misaligned flows reduce clarity.
- 3 Some parts are visually cluttered or misaligned, requiring effort to follow the flow.
- 4 Mostly clear layout; occasional overlap or inconsistent alignment, but overall still readable.
- 5 Diagram is well-aligned, clearly structured, and easy to follow. No overlapping elements, and whitespace is used effectively..

- **Use of Lanes/Pools:**
  In collaborative processes involving multiple participants such as customers, departments and also external partners, the proper use of the pools and lanes is essential to represent interactions and responsibilities clearly. Pools in BPMN define separate participants and lanes indicate internal subdivisions. Using them incorrectly can result in confusion about who performs which task or where a message should flow.

  Possible question to ask: If multiple participants exist, are they properly modeled with pools and lanes?

  Scores are distributed as:

  - 0 No lanes or pools are used despite being necessary for multi-actor processes.
  - 1 Lanes/pools used inappropriately, leading to misunderstandings about responsibilities.
  - 2 Lanes or pools used incorrectly or inconsistently; process roles are not clearly separated.
  - 3 Lanes/pools used but not optimally (e.g., excessive splitting, unclear assignment of tasks).
  - 4 Pools and lanes are mostly correct; minor misplacement or redundancy in participants' roles.

  - 5 Lanes and pools are used accurately to distinguish participants and organizational roles. Clear separation of responsibilities.

5. **Similarity to Reference**

   This category focuses on how closely the generated BPMN model mirrors the reference solution in structure, composition and behaviour, while all the other categories were mostly about how much the AI generated models were reflects the original process definition. It is especially important, when using AI for automation, even slight deviations in structure or behaviour can lead functional mismatches or misinterpretations.

   For Similarity to Reference the subcriteria and their reasoning and scoring are introduced as:

   - **Structural Similarity:**
     This constraint refers to the preservation of the overall process organization such as order or the tasks, direction of flows and use of subprocesses or branches. The minor layout differences are acceptable but the generated model should follow a comparable structural logic to ensure that it reflects a similar workflow logic as the reference.

     Possible question to ask: Do they share similar structure in task order and flow?

     Scores are distributed as:
     - 0 No structural similarity; the process model is fundamentally different from the reference.
     - 1 The structure is mostly unrelated to the reference; only vague similarities in task grouping or sequence remain.
     - 2 Significant structural differences exist, potentially altering the flow logic or understanding of the process.
     - 3 Moderate structural deviations are present, but the overall process logic remains comparable to the reference.
     - 4 Minor differences in structure (e.g., flow arrangement, task grouping), but the overall process remains clearly aligned with the reference.
     - 5 The solution mirrors the structure of the reference model very closely, including task order, flow direction, and layout organization. No meaningful structural deviations.

   - **Element Match Ratio:**
     This constraint measures the percentage of element overlap between the AI generated model and the reference diagram. Each BPMN model consist of

standard elements such as tasks, start events, gateways, and accurate replication of these components is vital when evaluating how well an AI model has understood and reproduced a given process.

Possible question to ask: What percentage of elements (activities, events, etc.) are exactly the same?

Scores are distributed as:

- 0 No identifiable match with any elements from the reference model.
- 1 Only a few matching elements can be identified (less than 25%).
- 2 Less than half of the reference elements are present or correctly identified (25–49% match).
- 3 A moderate portion of elements match the reference (around 50–74%). Some key steps may be missing or mislabeled.
- 4 Most elements are present and correctly identified (around 75–89% match).
- 5 Nearly all key elements (tasks, events, gateways) match those in the reference model (90–100% match).

- **Functional Equivalence:**
  It considers if the AI generated process performs the same logical function as the reference, regardless of how it is structured. Two BPMN models can be different in sense of layout or optimization and yet, it can still lead to the same outcomes and outputs. This criteria is particularly relevant where creative but valid transformations are acceptable.

  Possible question to ask: Even if structurally different, does it achieve the same behavior/output?

  Scores are distributed as:

  - 0 The model fails to capture the process behavior or goal at all.
  - 1 Major functionality is missing or changed, and the model does not fulfill the process goal correctly.
  - 2 Several key functions or outcomes differ from the reference model; process behavior is partially correct.
  - 3 Functional gaps or simplifications exist, but the main objective of the process is still achieved.
  - 4 Minor functional differences exist (e.g., one path slightly optimized), but the core logic and outcome are preserved.
  - 5 Even if structurally different, the model achieves exactly the same behavior and outputs as the reference (behavioral equivalence maintained).

The following image, 4.3 represents the evaluation scoring system used in this study, showing the scores assigned to each constraint for one selected BPMN exercise as an example. The complete set of scores for all exercises is available in the shared Google Drive folder linked in the Appendix.

	max point	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5		
		0.2				0.25			0.15			0.15			0.25				
		Completeness				Correctness			BPMN Syntax			Style&Clarity			Similarity to reference				
Exercise Name	AI Model	Activities Included	Gateways used properly	Start/End Events	Data/Message Present	Logical Flow	Gateways lead to valid outcomes	Error-Free token flow	Correct Usage of Shapes	Connections valid (no semantic violations)	Labels and Annotations	Naming Consistency	Layout readability	Use of Lanes/Pools	Structural Similarity	Element Match Ratio	Functional Equivalence	Total score	Average score
Signature Regulation	ChatGPT	5	5	5	5	5	5	4	5	5	5	5	4	5	5	5	5	15.6	13.1625
	Copilot	4	5	5	5	5	5	4	5	5	5	3	3	5	5	5	5	14.95	
	Gemini	2	1	5	0	1	2	5	5	5	2	1	5	3	2	2	1	8	
	Deepseek	5	5	5	3	4	5	4	4	4	5	4	3	5	4	5	5	14.1	

Figure 4.3. Evaluation Scoring System Example-Signature Regulation

4.4 Statistical Methods

To evaluate and validate if the overserved performance differences among AI models were statistically meaningful, **non-parametric statistical tests** were used. The evaluation scores were expressed an ordinal scale of 0-5 and derived from the same set of BPMN exercises for all models, which made non-parametric methods particularly appropriate [8].

Unlike parametric approaches such as **ANOVA** or **t-test** which usually assume normally distributed interval data, non-parametric tests do not require such assumptions and they are more robust for ordinal datasets [16, 31].

For this reason, we have performed two complementary analyses: **Kruskal-Wallis** and **Wilcoxon Signed-Rank Tests.**

Firstly, we have started with **Kruskal-Wallis Test** which is the non-parametric equivalent of the one-way ANOVA. This test was used to determine if there are significant differences existed among the four AI models when considering their overall adequacy scores. The null hypothesis stated that all of the models performed equally, on the other hand the alternative hypothesis proposed that at least one model's distribution was different from the others. The Kruskal-Wallis test was performed with a significance level of alpha=0.05, which is corresponding to a 95% confidence interval [28].

Following this comparison, we have performed **Wilcoxon Signed-Rank Tests** to examine the differences between specific pairs or models. These tests were particularly useful to explore the performance relationships between **top two performing systems**, *ChatGPT vs.Gemini* and also **top-performing and lower-performing systems**, *Gemini vs.Deepseek*. The Wilcoxon test was selected because it is used for paired observations of ordinal data and does not assume normality [49].

A two-tailed configuration was adopted in all comparisons, because the analysis aimed

to detect any significant difference in either direction, rather than assuming the superiority of a specific model in advance [16].

All statistical tests were performed using and online computational platform and validated through descriptive inspection in Microsoft Excel. For all analyses the threshold of significance was set at $p < 0.05$ while, non-significant results ($p \geq 0.05$) were interpreted as evidence that the models performed comparably withing the dataset, whereas significant results indicated a consistence performance difference between results.

This methodological approach was chosen to balance statistical **robustness and interpretative clarity** while accounting for the relatively small dataset size of 15 exercises.

4.5 Limitations of the Evaluation Methodology

Even though the evaluation framework provided a systematic and objective way to compare the AI generated BPMN diagrams, we need to recognize certain limitations. These constraints do not undermine the validity of the findings but they help to **define the scope which the results should be interpreted**.

- **Sample Size and Data Diversity**

  Our analysis is based on 15 BPMN exercises. Although this number was sufficient to identify the general performance needs, it might not fully depict the variety of real-world process modelling scenarios. Larger and more varied datasets can increase the results' statistical power and make conclusions more applicable.

- **Subjectivity in Manual Scoring**

  Each AI output was manually evaluated using the predefined constraints and a 0-5 scoring scale. Even if the use of clear guidelines and weighted categories, the process is inevitably involves a degree of human judgement. Small differences in interpretation of completeness or correctness may influence the assigned scores. Future studies can reduce this limitation by including several evaluators and comparing their agreement to ensure a more consistent scoring.

- **Simplification of the Evaluation Framework**

  In order to keep the comparison consistent across models, the evaluation is focused on a limited set of measurable criteria. This simplification was necessary for quantitative analysis but it may not fully consider the qualitative aspects such as the semantic depth or the contextual adequacy of modelled processes. Complementary qualitative reviews could develop future analyses.

- **Statistical Constraints**

  The ordinal nature of the scoring system and the relatively small dataset led to the use of the non-parametric statistical methods of the Kruskal-Wallis and Wilcoxon

Signed-Rank Tests. While these tests are appropriate for ordinal data, they are generally less sensitive to small differences than the parametric methods. This might have reduced the probability of detecting subtle performance variations among different models.

- **Model Variability and Reproducibility**

  AI models such as ChatGPT, Copilot, Gemini and Deepseek can generate slightly different outputs even if they were prompted identically, as their responses depend on probabilistic generation processes and may vary across different model versions. Furthermore, each system has been trained with different objectives, datasets and architectures which makes them particularly effective in different domains or task types.

Despite these limitations, the adopted methodology provided a balanced compromise between analytical depth and practical feasibility to the analysis. It enabled a systematic comparison of AI based BPMN generation while maintaining the statistical and conceptual coherence. Recognising these boundaries would offer valuable guidance for future research that aims to refine evaluation metrics, expand datasets or incorporate automated validation tools.

Chapter 5

Results and Discussion

This chapter presents the outcomes of the evaluation conducted on BPMN diagrams generated by the for selected AI models of ChatGPT, Copilot, Gemini and DeepSeek. It reports firstly, quantitative findings obtained and followed by a qualitative discussion interpreting these results in relation to diagram quality, logical accuracy and model behaviour. The overall goal is to provide an integrated understanding of how each AI system performed across the evaluated dimensions.

5.1 Results

5.1.1 Overview of the dataset

The evaluation included 15 BPMN exercises, solved by four AI models of ChatGPT, Copilot, Gemini and DeepSeek. Each generated diagram was assessed through a multi-criteria scoring framework specially created for this study, that contains completeness, correctness, BPMN syntax, style&clarity and similarity to the reference solution. The scores are ranged from 0 to 5 across all criteria with different weights and aggregated results were computed for each model to determine their overall performance. The compiled data formed the basis for descriptive and inferential statistical analysis using non-parametric tests of Kruskal-Wallis and Wilcoxon Signed-Rank. Those tests chosen due to the limited sample size and the non-normal distribution of the scores.

5.1.2 Descriptive Statistics

Across the 15 BPMN exercises, the evaluation scores demonstrated consistent trends reflecting the varying capabilities of AI systems in managing process complexity. Due to the scoring systems that was created for understanding the difficulties of the exercises, **relatively simpler processes** such as Placing an Order or Business Trip **tend to receive higher scores** across all models, while exercises that involve **multiple actors, message flows or gateways**, such as International Transport of Goods or Fire Extinguishing Collaboration Model receive **noticeably lower values**. This pattern indicates that all models generally handled linear or single-pool workflows effectively but struggled to represent multi-pool interactions or parallel structures.

The descriptive analysis that is presented in the table 5.1 summarize the overall performance of all the four AI systems. **Gemini** achieved the highest mean score of **12.27** followed by **ChatGPT (11.30), Copilot (10.97), and Deepseek (10.38**). **Median values** are relatively homogeneous and they suggest a consistent **central tendency across all models**, it is ranging between 11.45 and 11.85. **Standard deviation** values indicate **moderate variability**, with Copilot showing the largest spread (3.76) and Gemini the smallest (2.35), meaning that **Gemini's responses were both accurate and stable across different exercises.**

Table 5.1. Comparison of AI model performance based on mean, median, and standard deviation.

Model	Mean Score	Median	Standard Deviation	Rank
Gemini	12.27	11.85	2.35	1
ChatGPT	11.30	11.50	3.27	2
Copilot	10.97	11.45	3.76	3
Deepseek	10.38	11.45	3.33	4

As illustrated in figure 5.1, the bar chart represents the average evaluation scores of all four models, accompanied by their respective standard deviations.This visual representation highlights how the results align with the descriptive statistics in table 5.1 , confirming Gemini's clear advantage in both mean performance and consistency. The relatively shorter deviation bar of Gemini further reinforces its stability across exercises, whereas the wider spreads observed in Copilot and ChatGPT indicate greater variability in their outputs.

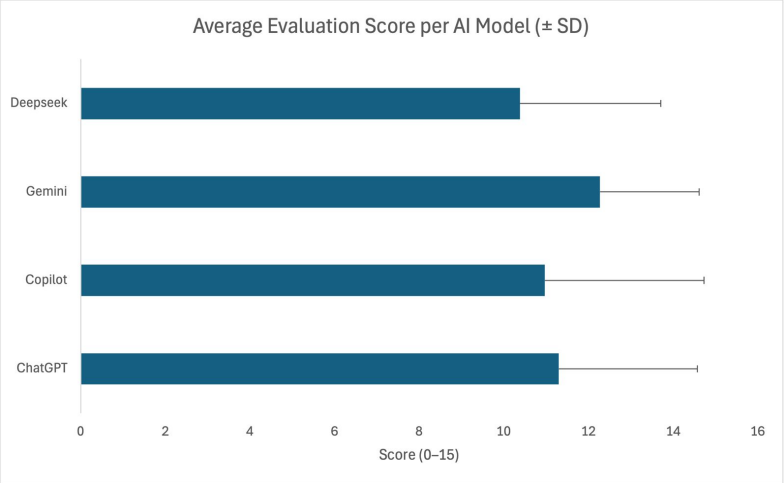

Figure 5.1. Average Evaluation Score per AI Model

5.1.3 Statistical Analysis

Kruskal-Wallis test was applied to the overall evaluation scores to determine if the observed performance differences were statistically significant. The analysis yielded an H statistic = 2.5322 and p = 0.46949, which is showing that the **differences among the four models are not statistically significant** at the 0.05 confidence level. This means that, despite visible variations in average scores, models' overall performance distributions do not differ significantly when considered collectively.

After that to investigate model-specific differences, Wilcoxon Signed-Rank tests were conducted for selected pairs. The **ChatGPT vs. Gemini** comparison (representing the two best performers) resulted in Z =−1.1359, p = 0.2543, indicating **no statistically significant difference.** However, the **Gemini vs. Deepseek** comparison (best vs. worst) yielded Z = −2.2718, p = 0.0232, **confirming a significant difference at p < 0.05.**

Table 5.2. Pairwise comparison results with Z-values and p-values.

Comparison	Z-value	p-value	Significance
ChatGPT vs Gemini	-1.1359	0.2543	Not significant
Gemini vs Deepseek	-2.2718	0.0232	Significant

5.1.4 Summary of the Findings

The descriptive results show that **Gemini achieved the highest overall performance and it follows by ChatGPT, Copilot, and Deepseek.**

Even if the Kruskal-Wallis test revealed no statistically significant global difference among the four AI models with H = 2.5322, p = 0.46949 the Wilcoxon Signed-Rank tests identified a significant difference between Gemini and Deepseek with $Z = -2.2718$, p = 0.023. This confirms Gemini's superiority over the lowest-performing model.

On the other hand, the difference between Gemini and ChatGPT was not statistically significant due to $Z = -1.1359$, p = 0.2543. This indicates that the two systems performed comparably in overall adequacy. These results provide a clear quantitative foundation for the following discussion, which explores how these numerical differences relate to the qualitative characteristics and behavioural patterns of each model.

5.2 Discussion

5.2.1 Interpretation of Quantitative Findings

Building upon the statistical results presented in the previous section, this discussion shows how the numerical differences amont the evaluated models relate to their internal reasoning and process representation capabilities. The overall results indicate that

Gemini demonstrated the strongest ability to maintain logical and structural consistency, especially in complex workflows that requires a clear coordination between process elements. **ChatGPT** showed **comparable conceptual understanding**, even if its outputs occasionally lacked peripheral details or presented simplified representations in multi-actor scenarios. **Copilot**, is **capable of generating coherent sequential logic**, but it is also tend to follow more rigid and rule-based structures so consequently this fact limited its adaptability to unconventional process designs.

On the other hand, **DeepSeek** has presented **greater fluctuations in accuracy and syntax**, especially when it faced with tasks that involve multiple interactions or decision gateways. These differences suggest that Gemini and ChatGPT are more capable of preserving context and managing hierarchical relationships within processes, where Copilot and DeepSeek show more sensitivity to contextual ambiguity and prompt complexity.

5.2.2 Qualitative Evaluation of Diagram Quality

While the quantitative analysis provided a numerical comparison of the performance, the qualitative analysis show deeper insights into "How?", the nature and structure of the BPMN diagrams produced by the evaluated AI systems. This analysis focuses on the completeness, syntactic correctness, logical coherence and interpretability. These facts assess not only accuracy but also the structural quality of the generated models.

- **Completeness**

  Differences among the AI systems were most apparent in the degree to which their diagrams captured the full scope of each process. The most comprehensive representations that often includes pools, lanes and message flows in multi-actor workflows were produced by Gemini. Also, ChatGPT covered the essential process logic but at the same time it tends to discard secondary flows or conditional branches to preserve clarity especially in simpler exercises. Copilot captured the main sequence of tasks reliably but rarely integrated optional or parallel activities. However DeepSeek, frequently discarded gateways or subprocesses which indicates a partial understanding of process hierarchies.

- **Syntactic Correctness**

  Compliance with BPMN standards differed among models. Copilot consistently used the appropriate notations and event categories, demonstrating the highest degree of syntactic consistency. Although Gemini mostly adhered to the standard, there were sporadic little variations in flow labeling. ChatGPT, balanced readability with formal accuracy whereas DeepSeek generated the highest number of syntax deviations such as the representation of message flows and end events.

- **Logical Coherence**

  When analysing task sequencing and causal flow, Gemini maintained the most consistent logic between activities, especially in multi-lane diagrams.In linear processes,

ChatGPT showed logical ordering; however, when working with loops, it occasionally introduced discontinuities.. Copilot ensured a strictly ordered sequence but lacked flexibility to reinterpret non-standard instructions. DeepSeek showed the weakest logical integrity with occasional broken or circular flows that disrupted process readability.

- **Interpretability**

  Each model was able to produce diagrams that remained understandable to readers familiar with BPMN notation even if they vary in clarity. Gemini and ChatGPT achieved the best balance between visual simplicity and informational richness, while Copilot prioritised structural correctness over readability. DeepSeek's outputs, due to missing or inconsistent elements, often required manual correction to be fully interpretable.

From this perspective, the qualitative evaluation demonstrates that syntactic precision does not ensure conceptual completeness by itself. Models like Copilot can generate technically correct diagrams that lack expressive depth, on the contrary Gemini and ChatGPT illustrate that contextual comprehension enhances both logic and interpretability. These findings complement the quantitative trends, and they confirm that effective BPMN generation depends on a model's ability to integrate formal syntax with semantic understanding which is a capability that is still uneven across current large language models.

5.2.3 Comparative Observations

Comparative analysis of the data shows clear behavioral patterns across the four AI systems. This reflects the underlying diversity of their architectures and training objectives. Rather than individual strengths or weaknesses, what emerges is a spectrum of modelling behaviours ranging from rule-based precision to contextual adaptability.

Gemini and ChatGPT are in the upper end of this spectrum. Both models demonstrated the ability to maintain coherent logic and align the generated diagram with the intended meaning of the process description. This proximity is not only qualitative but also supported by quantitative evidence. According to the Wilcoxon Signed-Rank test that was conducted specifically for these two models, the difference between their performance scores was not statistically significant, confirming that the two systems achieved comparable levels of adequacy and syntactic accuracy.

This finding emphasizes that the observed similarity does not come from subjective assessment but from measurable consistency across multiple exercises.

Although they perform similarly overall, there is a slight difference in how they approach process construction: ChatGPT relies more on sequential reasoning, generating diagrams that are semantically accurate but sometimes simplified in representation, while Gemini tends to construct processes more holistically, maintaining global structure and connections among activities. Copilot stands at the midpoint, serving as an example

of rule-governed reliability. Its strict adherence to BPMN conventions ensures syntactic precision, but this same characteristic limits its capacity to adapt when confronted with ambiguous or non-standard descriptions. In contrast, DeepSeek represents the opposite end of the behavioural spectrum. It displays a pattern of exploratory generation that producing varied yet inconsistent diagrams, where correctness sometimes emerges by chance rather than from stable internal logic.

Across all of the models, one common aspect is the reliance on linguistic coherence as the foundation of process construction. But there are significant differences in how much of this language reasoning translates into sound structural reasoning. While Copilot and Deepseek show that syntactic conformance by itself does not ensure semantic alignment, Gemini and ChatGPT show that greater contextual comprehension results in more integrated and interpretable diagrams.

Overall, the results show that BPMN production depends on the ability of large language models to capture and replicate relational and hierarchical logic, which is still a restriction for systems that are not process modeling specialists. It also requires more than just linguistic proficiency.

5.2.4 Model Variability and Reliability

The stability and reliability of the models' output across various process types is one of the study's key issues. Variability reflects how consistently a model can reproduce coherent results when faced with exercises of increasing complexity, while reliability refers to its ability to maintain logical and structural quality regardless of contextual changes.

Across the 15 BPMN exercises that we have used, the results show that Gemini maintained the highest degree of reliability. Its performance was relatively stable even in the more complex workflows. This indicates that the model could generalize its internal reasoning beyond simpler sequential processes. ChatGPT, even it is slightly more variable, also exhibited stable behaviour in most cases. Especially when the prompts were clearly structured. This consistency reinforces the earlier statistical findings that showed minimal difference between the models in Wilcoxon analysis, which suggests that they both rely on well-balanced reasoning mechanisms.

On the other hand, the results from Copilot and DeepSeek displayed a higher variability in their results. Copilot's stability was mainly due to its strict adherence to BPMN syntax, rather than genuine contextual understanding Consequently, although the process descriptions were drawn from standard BPMN logic, its outputs lacked flexibility, even if it consistently produced syntactically accurate diagrams.

DeepSeek, responses often changed substantially when identical prompts were submitted at different times, producing divergent interpretations of the same scenario. This suggests a weaker internal representation of process logic and a stronger sensitivity to prompt phrasing and sampling randomness.

From a methodological standpoint, this observed variability highlights a key limitation of using general-purpose language models for structured diagram generation. Even minor temporal or contextual variations can lead to different interpretations of the process since these models produce probabilistic outputs rather than deterministic rule-based systems.

As a consequence, repeatability remains as a major obstacle. Using multi-stage prompting strategies that assess and refine several candidate outputs before final selection or fine-tuning on specialized process-modeling data could increase the reliability of BPMN creation.

In summary, Gemini and ChatGPT demonstrate a relatively consistent behaviour across tasks while the performance of Copilot and DeepSeek remains more inconsistent. This highlights that stability is not only a matter of average accuracy but also a reproducibility and resistance to contextual noise. Which are factors that can be crucial for any future integration of AI models into process modelling workflows.

5.2.5 Methodological and Practical Reflections

From a methodological point of view,a systematic and transparent evaluation process allowed by the integration of a structured scoring framework and statistical validation methods. By designing a two layer approach of measuring exercise difficulty and then assessing AI-generated outputs, this study ensured that performance comparisons are based on objective criteria rather than subjective impressions.

One of the main methodological strengths is the variety of exercises selected. The inclusion of 15 BPMN scenarios from different business and operational contexts, allowed more comprehensive understanding of how AI systems behave across different process types and level of complexity. Furthermore, evaluating four different AI models under the same scoring framework provided a more fair basis cross-model comparison and reproducibility.

The evaluation's findings provided valuable insights regarding how AI can be integrated into process-modelling workflows. The results show that LLMs can assist domain experts by producing basic logical structures that can accelerate model development and standardization. However, some **AI-generated outputs lack stability and domain-specific reasoning, which emphasizes how crucial it is to maintain human oversight**.

5.2.6 Broader Implications

Beyond a technical evaluation of AI performance, the study's findings demonstrate that LLMs can capture and replicate important features of procedural logic even in the absence of specialized BPMN training. This suggest that AI systems could begin to function not only as tools for natural language understanding but also as assistive agents for structured reasoning and model generation.

However, the integration of AI into process modelling environments also raises questions of trust, transparency, and interpretability. While Gemini and ChatGPT produced logically consistent results, the way they reach these outcomes is still hard to interpret. This lack of clarity represents a challenge in areas such as business process management, where transparency and accountability are more of an obligation. As a result, AI-generated diagrams must remain understandable and verifiable by human experts to be trusted.

Another implication concerns ethical and organizational responsibility. **Sensitive or confidential internal processes**, **data exchanges**, and **decision-making processes** are frequently reflected in BPMN diagrams. It requires strong safeguards for privacy, data protection, and intellectual ownership to access and process such information. In this sense, the development of governance frameworks that define the ethical use of AI in process modelling will be as important as the technical progress itself.

Chapter 6

Conclusion and Future Works

6.1 Conclusion

This research examined the ability of Large Language Models to generate and interpret BPMN diagrams by combining structured scoring, qualitative evaluation and non-parametric statistical analysis. The goal was to determine whether these models can move beyond textual generation to represent the logical and contextual structure of business processes in a rational and standardized way.

The evaluation that was conducted across fifteen BPMN exercises revealed a consistent but distinctive pattern among the tested AI systems. **Gemini achieved the highest and most stable performance overall, then followed by ChatGPT, while Copilot and DeepSeek showed lower and more variable scores.** Although the Kruskal-Wallis test indicated that there is no statistically significant overall difference, the Wilcoxon Signed-Rank test confirmed that Gemini's performance was noticeably higher than DeepSeek's performance. Qualitative analysis supported these results, it shows that Gemini and ChatGPT maintained a more logical flow and structural consistency. Copilot produced rigid but syntactically correct diagrams and DeepSeek generated damaged or incomplete representations

These results suggest that, although current LLMs show increasing proficiency in formal process modeling, their comprehension is still **primarily syntactic**. The ability to comply with BPMN notation does not necessarily ensure semantic completeness or contextual accuracy. Gemini's success emphasizes the benefit of models trained on an extensive contextual data, while Copilot and DeepSeek's results reflect the limits of rule-based or code-focused reasoning.

From a methodological perspective, our study showed that the strengths and weaknesses of AI-generated BPMN diagrams may be successfully captured by combining statistical reasoning with structured evaluation. The adopted approach shown meaningful behavioural patterns and provided evidence-based results into how LLMs reason process structure instead of relying on isolated accuracy scores.

In conclusion, this study indicates that **AI models are not yet capable of generating BPMN diagrams in a completely autonomously**. However, they can **provide valuable support during the analytical and design phases of process modelling**. Their ability to create diagrams that are partially complete, structurally correct and generally interpretable represents a significant step toward automation in business process representation. Overall, the results highlight that the main challenge is enabling these systems to reason within context and to capture the full logic of the process, rather than simply recreating its syntax.

6.2 Future Works

This study opens several directions for further study in the generation and evaluation of BPMN diagrams produced by AI models.

A first natural next step would be to **expand the dataset** of BPMN exercises by including a wider range of business scenarios and problem domains. A **more diverse** dataset would allow a **stronger statistical validation** and provide a deeper understanding of how model performance changes with process complexity and domain specificity.

Even if this study already established a structured and measurable framework for evaluating BPMN generation, future work could focus on **enhancing the semantic reasoning capabilities** of language models. This would allow them to better understand process context instead of relying mainly on syntactic accuracy. Training or adapting models to learn relationships between activities, actors and message flows more consistently would represent a meaningful advancement. Another relevant improvement could be the **direct generation of BPMN diagrams without human refinement**. This would decrease the differences between logical reasoning and visual process by reducing interpretation errors introduced during the manual conversion of text outputs. Improving models so they can more reliably understand and represent the relationships between activities, actors, and message flows would mark a notable improvement.

Further research may also explore **cross-model collaboration**, combining models specialized in syntax compliance with another one that is optimized for contextual interpretation. Such hybrid system would come up with more reliable and semantically rich outputs.

Finally, extending the proposed evaluation framework to other modelling languages would help assess the generalizability of these findings and broaden the understanding of how language models interpret and formalize processes across different paradigms.

Bibliography

[1] Daniele De Bari, Giacomo Garaccione, Riccardo Coppola, Luca Ardito, and Marco Torchiano. Evaluating large language models in exercises of uml class diagram modeling. In *Proceedings of the 18th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM '24)*, pages 393–400, Barcelona, Spain, 2024. ACM.

[2] Paolo Bellan, Marco Dragoni, and Chiara Ghidini. Process Extraction from Text: State of the Art and Challenges for the Future. *CoRR*, abs/2110.03754, 2021.

[3] Weiyi Bian, Omar Alam, and Jörg Kienzle. Automated grading of class diagrams. In *2019 ACM/IEEE 22nd International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*, pages 700–707. IEEE, 2019.

[4] Narasimha Bolloju and Felix S. K. Leung. Assisting novice analysts in developing quality conceptual models with uml. *Communications of the ACM*, 49(7):108–114, 2006.

[5] Ana Bordignon, Lucinéia Thom, Thanner Soares Silva, Vinícius Stein Dani, Marcelo Fantinato, and Renato Ferreira. Natural language processing in business process identification and modeling: A systematic literature review. In *Proceedings of the 20th International Conference on Enterprise Information Systems (ICEIS)*, June 2018.

[6] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shaym, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1–22, 2020.

[7] Riccardo Coppola. Large language models: Prompt engineering. Course Material, 2024. DAUIN - Department of Control and Computer Engineering.

[8] G. W. Corder and D. I. Foreman. *Nonparametric Statistics: A Step-by-Step Approach*. Wiley, 2 edition, 2014.

[9] Kate Crawford. *Atlas of AI: Power, Politics, and the Planetary Costs of Artificial Intelligence*. Yale University Press, New Haven, CT, 2021.

[10] Remco M. Dijkman, Marlon Dumas, and Chun Ouyang. Semantics and analysis of business process models in bpmn. *Information and Software Technology*, 50(12):1281–1294, 2008.

[11] Marlon Dumas, Marcello La Rosa, Jan Mendling, and Hajo A. Reijers. *Fundamentals of Business Process Management*. Springer, second edition, 2018.

[12] Ali Nour Eldin, Nour Assy, Olan Anesini, Benjamin Dalmas, and Walid Gaaloul. Nala2BPMN: Automating BPMN Model Generation with Large Language Models.

In *Cooperative Information Systems - 30th International Conference, CoopIS 2024, Porto, Portugal, November 19-21, 2024, Proceedings, Demo Track*, volume 15506 of *Lecture Notes in Computer Science*.

[13] High-Level Expert Group on Artificial Intelligence European Commission. Ethics guidelines for trustworthy ai. Technical report, European Commission, 04 2019. Accessed: 2025-10-28.

[14] Joerg Evermann, Jana-Rebecca Rehse, and Peter Fettke. A deep learning approach for predicting process behaviour at runtime. pages 327–338, 05 2017.

[15] Reza Fauzan, Daniel Siahaan, Siti Rochimah, and Evi Triandini. Automated class diagram assessment using semantic and structural similarities. *International Journal of Intelligent Engineering and Systems*, 14(2):52–63, 2021.

[16] Andy Field. *Discovering Statistics Using IBM SPSS Statistics*. SAGE Publications, London, UK, 4th edition, 2013.

[17] Sarah Foss. Autoer: A system for the automatic generation and evaluation of uml database design diagrams. Master's thesis, The University of British Columbia, Okanagan, Canada, 2022.

[18] Sarah Foss, Tatiana Urazova, and Ramon Lawrence. Automatic generation and marking of uml database design diagrams. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education (SIGCSE '22)*, pages 626–628, Providence, RI, USA, 2022. ACM.

[19] Robert France and Bernhard Rumpe. Model-driven Development of Complex Software: A Research Roadmap. *Future of Software Engineering (FOSE '07)*, pages 37–54, 2007.

[20] Giacomo Garaccione, Riccardo Coppola, and Luca Ardito. Gamifying business process modeling education: A longitudinal study. In *Proceedings of the 28th International Conference on Evaluation and Assessment in Software Engineering (EASE 2024)*, Salerno, Italy, 2024.

[21] Giacomo Garaccione, Riccardo Coppola, Luca Ardito, and Marco Torchiano. Gamification of a bpmn modeling course: an analysis of effectiveness and student perception. In *Proceedings of the 18th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM 2024)*, Barcelona, Spain, 2024.

[22] Giacomo Garaccione, Riccardo Coppola, Luca Ardito, and Marco Torchiano. Gamification of conceptual modeling education: an analysis of productivity and students' perception. *Software Quality Journal*, 33(3), 2025.

[23] Katarzyna Gdowska, Maria T. Gomez-Lopez, and Joerg-Reiner Rehse, editors. *Business Process Management Workshops: BPM 2024 International Workshops, Krakow, Poland, September 1–6, 2024, Revised Selected Papers*, volume 534 of *Lecture Notes in Business Information Processing*. Springer, Cham, 2025.

[24] Pablo Gómez-Abajo, Esther Guerra, and Juan de Lara. Wodel: A Domain-Specific Language for Model Mutation. In *Proceedings of the 31st Annual ACM Symposium on Applied Computing (SAC 2016)*, pages 1968–1975, Pisa, Italy, 2016. ACM.

[25] Pablo Gómez-Abajo, Esther Guerra, and Juan de Lara. Automated generation and correction of diagram-based exercises for moodle. *Computers and Applications in Engineering Education*, 31(6), 2023.

[26] Michael Grohs, Maximilian Hutterer, Moritz Lang, and Daniel Haisch. Large language models can accomplish business process management tasks, 2023. Accepted at NLP4BPM workshop at BPM 2023.

[27] Volker Gruhn and Ralf Laue. Reducing the cognitive complexity of business process models. In *2009 8th IEEE International Conference on Cognitive Informatics*, pages 339–345, 2009.

[28] Myles. Hollander, Douglas A. Wolfe, and Eric Chicken. *Nonparametric statistical methods.* Wiley series in probability and statistics. John Wiley Sons, Inc., Hoboken, New Jersey, third edition / myles hollander, department of statistics, florida state university, tallahassee, florida, douglas a. wolfe, department of statistics, ohio state university, columbus, ohio, eric chicken, department of statistics, florida state university, tallahassee, florida. edition, 2013.

[29] Luca Hörner, Julius Köpke, Sebastian Rinker, Martin Matzner, Vladislav Janoušek, and Jan Mendling. Introducing BPMNGen: An LLM-based Conversational Framework for BPMN 2.0 Process Model Generation. In *Proceedings of the EMISA Forum*, 2025.

[30] IEEE Computer Society. Software engineering models and methods. *IEEE Computer Society Resources*, n.d. Accessed: 31/10/2025.

[31] Susan Jamieson. Likert scales: How to (ab)use them? *Medical Education*, 38(12):1217–1218, 2004.

[32] Wangfan Li and Carlos Toxtli. Automating automation: Using llms to generate bpmn workflows for robotic process automation. *ResearchGate*, 2024. Conference Paper.

[33] Ivar Lindland, Guttorm Sindre, and Arne Sølvberg. Understanding quality in conceptual modeling. *IEEE Software*, 11(2):42–49, 1994.

[34] Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, and Hannaneh Hajishirzi. Generated knowledge prompting for commonsense reasoning. *arXiv preprint arXiv:2110.08387*, 2022.

[35] Tomás Lopes and Sérgio Guerreiro. Assessing business process models: A literature review on techniques for BPMN testing and formal verification. *Business Process Management Journal*, 29(8):133–162, 2023.

[36] Jan Mendling. Managing structural and textual quality of business process models. In Philippe Cudre-Mauroux, Paolo Ceravolo, and Dragan Gašević, editors, *Lecture Notes in Business Information Processing*, volume 162, pages 100–111. Springer, 2013.

[37] Jan Mendling, Hajo A. Reijers, and Wil M. P. van der Aalst. Seven process modeling guidelines (7pmg). *Information and Software Technology*, 52(2):127–136, 2010.

[38] Jan Mendling, Mark Strembeck, and Jan Recker. Factors of process model comprehension-findings from a series of experiments. *Decision Support Systems*, 53(1):195–206, 2012.

[39] Brent D. Mittelstadt, Patrick Allo, Mariarosaria Taddeo, Sandra Wachter, and Luciano Floridi. The ethics of algorithms: Mapping the debate. *Big Data & Society*, 3(2):1–21, 2016.

[40] Madline Mößlang, Reinhard Bernsteiner, Christian Ploder, and Stephan Schlögl.

Automatic Generation of a Business Process Model Diagram Based on Natural Language Processing. In *Knowledge Management in Organizations (KMO 2024)*, volume 2152 of *Communications in Computer and Information Science*, pages 237–247. Springer Nature Switzerland, 2024.

[41] Meike Nauta, Jan Trienes, Shreyasi Pathak, Elisa Nguyen, Michelle Peters, Yasmin Schmitt, Jörg Schlötterer, Maurice van Keulen, and Christin Seifert. From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable AI. *arXiv preprint*, 2023.

[42] Oksana Nikiforova, Konstantins Gusarovs, Ludmila Kozacenko, Dace Ahilcenoka, and Dainis Ungurs. An approach to compare uml class diagrams based on semantical features of their elements. In *Proceedings of the 10th International Conference on Software Engineering Advances (ICSEA 2015)*, pages 147–154. IARIA, 2015.

[43] Quentin Nivon and Gwen Salaün. Automated generation of BPMN processes from textual requirements. In *Service-Oriented Computing – ICSOC 2024*, volume 15404 of *Lecture Notes in Computer Science*, pages 185–201. Springer Nature Singapore, 2025.

[44] Object Management Group (OMG). Business Process Model and Notation (BPMN) Version 2.0.2. Formal Specification formal/2013-12-09, Object Management Group (OMG), jan 2014.

[45] Anas Outair, Mariam Tanana, and Abdelouahid Lyhyaoui. New method for summative evaluation of uml class diagrams based on graph similarities. *International Journal of Electrical and Computer Engineering*, 11(2):1578–1590, 2021.

[46] Andreas Papasalouros. Automatic exercise generation in euclidean geometry. In Harris Papadopoulos, Andreas S. Andreou, and Lazaros Iliadis, editors, *Artificial Intelligence Applications and Innovations (AIAI 2013)*, volume 412 of *IFIP Advances in Information and Communication Technology*, pages 141–150. Springer, 2013.

[47] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. Technical report, OpenAI, 2019. Technical report.

[48] Dorsa Sadigh, Sanjit A. Seshia, and Mona Gupta. Automating exercise generation: A step towards meeting the mooc challenge for embedded systems. In *Proceedings of the Workshop on Embedded Systems Education (WESE 2013)*, pages 2:1–2:8, Montreal, Canada, 2013. ACM.

[49] David J. Sheskin. *Handbook of Parametric and Nonparametric Statistical Procedures*. Chapman & Hall/CRC Press, Boca Raton, FL, 3rd edition, 2004.

[50] Bruce Silver. *BPMN Method and Style, Second Edition, with BPMN Implementer's Guide*. Cody-Cassidy Press, second edition, 2019.

[51] Anna Szymański, Noah Ziems, Heather A. Eicher-Miller, Toby Jia-Jun Li, Meng Jiang, and Ronald A. Metoyer. Limitations of the llm-as-a-judge approach for evaluating llm outputs in expert knowledge tasks. *arXiv preprint arXiv:2410.20266*, 2024.

[52] Marco Torchiano, Federico Tomassetti, Filippo Ricca, Alessandro Tiso, and Gianna Reggio. Relevance, benefits, and problems of software modelling and model driven techniques: a survey in the italian industry. *The Journal of Systems and Software*,

86:2110–2126, 2013.

[53] Han van der Aa, Dominik Bork, Rainer Schmidt, and Arnon Sturm, editors. *Enterprise, Business-Process and Information Systems Modeling: BPMDS 2024 and EMMSAD 2024 Proceedings*, volume 511 of *Lecture Notes in Business Information Processing*. Springer Nature Switzerland, Cham, 2024.

[54] Wil van der Aalst. *Process Mining*. Springer Berlin, Heidelberg, 2 edition, 2016.

[55] Irene Vanderfeesten, Jorge Cardoso, Jan Mendling, Hajo A. Reijers, and Wil M. P. van der Aalst. Quality metrics for business process models. *Informatica*, 31(4):457–468, 2007.

[56] Athish Venkatachalam and Carlos Toxtli. Assessing ai-generated workflows: A multi-dimensional evaluation framework. *Preprint*, 2025. Presented at the BPMN 2025 Conference.

[57] Maxim Vidgof, Stefan Bachhofner, and Jan Mendling. Large language models for business process management: Opportunities and challenges. *arXiv preprint arXiv:2304.04309*, April 2023.

[58] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc., 2022.

[59] Sven Weinzierl, Sandra Zilker, Sebastian Dunzer, and Martin Matzner. Machine Learning in Business Process Management: A Systematic Literature Review. *arXiv preprint arXiv:2405.16396*, 2024.

[60] Mathias Weske. *Business Process Management: Concepts, Languages, Architectures*. Springer, Berlin, Heidelberg, fourth edition, 2024.

[61] Stephen A. White. Introduction to bpmn. *BPTrends*, jul 2004.

[62] Stephen A. White and Derek Miers. *BPMN Modeling and Reference Guide: Understanding and Using BPMN*. Future Strategies Inc., Deerfield Beach, FL, 2008.

[63] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 11809–11822. Curran Associates, Inc., 2023.

List of Tables

4.1 Constraints and scoring criteria for BPMN exercise complexity evaluation. 35

5.1 Comparison of AI model performance based on mean, median, and standard deviation. 56
5.2 Pairwise comparison results with Z-values and p-values. 57

List of Figures

2.1 Common BPMN Elements 12

4.1 BPMN Exercise Solution Example, from Gdowska et al. [23]. 32
4.2 Zero-shot Prompting by ChatGPT 37
4.3 Evaluation Scoring System Example-Signature Regulation 51

5.1 Average Evaluation Score per AI Model 56

Appendix A

List of BPMN Exercises and Sources

This appendix provides the list of BPMN exercises used in this study, together with the reference to the shared online repository containing all the materials used for analysis.

All exercises, their corresponding BPMN solutions, and AI-generated diagrams (ChatGPT, Copilot, Gemini, and DeepSeek) are available in the following online drive folder:

Google Drive Repository: `Open Repository`

The following lists report the BPMN exercises included in the dataset, each representing a distinct process scenario used for evaluation.

Exercises from Online Sources

- **Bill of Materials** - available online at WU Vienna Webtrainer.
- **Maintenance** - available online at WU Vienna Webtrainer.
- **Modifying Order Data** - available online at WU Vienna Webtrainer.
- **Placing an Order** - available online at WU Vienna Webtrainer.
- **Project Implementation** - available online at WU Vienna Webtrainer.
- **Signature Regulation** - available online at WU Vienna Webtrainer.

Exercises Adapted from Books or Academic Sources

- **Business Trip** - adapted from *Di Francescomarino, C., Burattin, A., Janiesch, C., and Sadiq, S. (Eds.). (2023).* ***Business Process Management Forum: BPM 2023 Forum, Utrecht, The Netherlands, September 11-15, 2023, Proceedings.*** *Springer, Cham.* DOI: 10.1007/978-3-031-41623-1

- **Mine Blasting Process** - adapted from *De Weerdt, J., and Pufahl, L. (Eds.). (2024).* ***Business Process Management Workshops: BPM 2023 International Workshops, Utrecht, The Netherlands, September 11-15, 2023, Revised Selected Papers.*** *Lecture Notes in Business Information Processing, Vol. 492. Springer, Cham.* DOI: 10.1007/978-3-031-50974-2

- **Fire Extinguishing Collaboration Model** - adapted from *Marrella, A., Resinas, M., Jans, M., and Rosemann, M. (Eds.). (2024).* ***Business Process Management: 22nd International Conference, BPM 2024, Krakow, Poland, September 1-6, 2024, Proceedings.*** *Lecture Notes in Computer Science, Vol. 14940. Springer, Cham.* DOI: 10.1007/978-3-031-70396-6

- **Hiring Process** - adapted from *Mueller, M., Simonet-Boulogne, A., Sengupta, S., and Beige, O. (2022). "Process Mining in Trusted Execution Environments: Towards Hardware Guarantees for Trust-Aware Inter-Organizational Process Analysis." In: J. Munoz-Gama and X. Lu (Eds.),* ***Process Mining Workshops (ICPM 2021 Workshops), Lecture Notes in Business Information Processing, Vol. 433****. Springer, Cham.* DOI: 10.1007/978-3-030-98581-3_27

- **Hospital Treatment Process** - adapted from *Gdowska, K., Gomez-Lopez, M. T., and Rehse, J.-R. (Eds.). (2025).* ***Business Process Management Workshops: BPM 2024 International Workshops, Krakow, Poland, September 1-6, 2024, Revised Selected Papers.*** *Lecture Notes in Business Information Processing, Vol. 582. Springer, Cham.* DOI: 10.1007/978-3-031-78666-2_5

- **Table Service Collaboration** - adapted from *Gdowska, K., Gomez-Lopez, M. T., and Rehse, J.-R. (Eds.). (2025).* ***Business Process Management Workshops: BPM 2024 International Workshops, Krakow, Poland, September 1-6, 2024, Revised Selected Papers.*** *Lecture Notes in Business Information Processing, Vol. 582. Springer, Cham.* DOI: 10.1007/978-3-031-78666-2_5

- **Job Scheduling** - adapted from *Gdowska, K., Gomez-Lopez, M. T., and Rehse, J.-R. (Eds.). (2025).* ***Business Process Management Workshops: BPM 2024 International Workshops, Krakow, Poland, September 1-6, 2024, Revised Selected Papers.*** *Lecture Notes in Business Information Processing, Vol. 534. Springer, Cham.* DOI: 10.1007/978-3-031-78666-2

- **International Transport of Goods** - adapted from *Di Francescomarino, C., Burattin, A., Janiesch, C., and Sadiq, S. (Eds.). (2023).* ***Business Process Management Forum: BPM 2023 Forum, Utrecht, The Netherlands, September 11-15, 2023, Proceedings.*** *Lecture Notes in Business Information Processing, Vol. 490. Springer, Cham.* DOI: 10.1007/978-3-031-41623-1_6

- **Research Process** - adapted from *Cabanillas, C., Garmann-Johnsen, N. F., and Koschmider, A. (Eds.). (2023).* ***Business Process Management Workshops: BPM 2022 International Workshops, Munster, Germany, September 11-16, 2022, Revised Selected Papers.*** *Lecture Notes in Business Information Processing, Vol. 460. Springer, Cham.* DOI: 10.1007/978-3-031-25383-6