Fraunhofer IEM

HEINZ NIXDORF INSTITUT
UNIVERSITÄT PADERBORN

Master's Thesis

Computer Science

Generative AI-Driven Approach for Systems Engineering Competency Assessment

submitted by

cand. Derik Roby

Matr.-Nr. 6961670

Examiners:

Prof. Dr.-Ing. Roman Dumitrescu
Dr. Christian Koldewey
Ulf Könemann
Oliver von Heißen

Paderborn, 21.03.2025

Masters in Computer Science, Matrikelnummer: 6961670

Generative AI-Driven Approach for Systems Engineering Competency Assessment
on: 21.03.2025

FRAUNHOFER IEM
Zukunftsmeile 1
D-33102 Paderborn,Germany

Non-disclosure notice

This work remains closed to the public due to confidential data and information.

Derik Roby

Ulf Könemann

Affidavit:

I hereby declare in lieu of an oath that I have prepared this thesis independently and without unauthorised external assistance, that I have not used any sources or aids other than those specified and that I have marked the passages taken from the sources used as such, either verbatim or in terms of content.

Paderborn, 21.03.2025

Summary

This thesis presents the research, analysis, and development of a Systems Engineering Competency Assessment Web App, followed by its evaluation and validation. The product aims to provide competency level gap analysis for professionals based on established industry standards. Additionally, the web app can comprehensively assess users across all competencies and provide optimal matching role suggestions.

The system logic for performing gap analysis and role suggestions is derived from established frameworks and concepts in Systems Engineering, Software Engineering, and Mathematics. Modern technologies like Generative AI are also employed to solve complex challenges like role-process mapping, which enables users to take a survey by describing their tasks in natural language. Based on the user inputs, the system calculates the required competency levels. App users also receive individualized Generative AI-powered assessment feedback.

The system development incorporated the best practices in software engineering and utilized modern tools and technologies for the successful realization and future-proofness of the research. The developed artefact is scientifically evaluated and validated for its alignment with the established framework and user-friendliness. The Generative AI-assisted evaluations performed for validating the product also highlight promising aspects of using generative AI in research.

While challenges remain, the work presented provides clear directions for future improvements. The work provides a strong foundation for further research and development, not only limited to Systems Engineering competency assessment but also exploring how software engineering techniques could be employed in combination with domain knowledge and novel technologies like Generative AI to address complex problems that were previously complex to solve.

Contents

Page

List of Figures

1 Introduction

The complexity of modern systems has increased significantly, evolving from mechanical structures to mechatronic and, more recently, cyber-physical systems. These advancements require collaboration across multiple disciplines, including engineering, natural sciences, information technology, and human factors. Systems Engineering (SE) enables this interdisciplinary coordination through structured methodologies, fostering clear communication and a comprehensive understanding of system components, thereby ensuring the efficient development of complex systems [DAR+21].

Managing complexity is a crucial aspect of developing modern systems. Systems Engineering provides a structured framework that addresses these challenges and facilitates their resolution. [BSF+20] demonstrate how Systems Engineering plays a vital role in overcoming both technical and organizational hurdles encountered throughout the system lifecycle. Similarly, the findings of [HTB+19] highlight that implementing Systems Engineering within organizations is essential for ensuring the consistent delivery of value in projects. Furthermore, BEASLEY emphasizes that SE helps organizations identify risks, optimize resource allocation, and enhance overall project success [Bea17]. This is particularly important in the development of complex products, where minimizing unforeseen issues and ensuring high-quality results are critical to improving project outcomes.

Integrating Systems Engineering into business processes further enhances organizational efficiency and performance. Studies show that organizations applying Systems Engineering methodologies achieve higher success rates, particularly in cost-intensive and highly competitive industries [AMK12]. This highlights Systems Engineering's role in helping businesses streamline operations and maintain a competitive edge.

Furthermore, Systems Engineering provides tools and methodologies that support strategic planning and execution, particularly in distributed teams. ZEC et al. highlight that collaborative Systems Engineering methodologies enhance coordination across teams, ensuring all stakeholders are aligned on project scope, technical requirements, and expected outcomes. This collaborative aspect is crucial for organizations relying on cross-functional teams to drive innovation and stay competitive [ZSD+14].

From these insights, it is evident that Systems Engineering is essential for managing complexity, improving project outcomes, enhancing organizational efficiency, and fostering collaboration across disciplines. Its structured methodologies and strategic integration enable organizations to develop high-quality systems while optimizing resources and mitigating risks, making Systems Engineering a critical discipline in today's increasingly sophisticated technological landscape.

To effectively perform Systems Engineering and develop advanced systems, engineers must possess specific competencies and qualifications. These competencies are essential for carrying out the necessary processes for the successful realization of advanced systems [INC23c]. They are typically gained through formal education as well as on-the-job learning, including both practical experience and dedicated professional training [DAR+21]. A survey conducted by DUMITRESCU et al. emphasizes the urgent need for new training models to equip employees with the skills needed to work with advanced systems. The study also shows that a skilled systems engineer brings significant value to a company by enhancing system understanding, identifying errors early in the development process, streamlining tasks, and supporting the creation of innovative, customer-oriented solutions [DAR+21]. Furthermore, competencies in Systems Engineering enable engineers to manage the complexity of advanced systems by ensuring that each decision is traceable and transparent through systematic documentation—a capability that is crucial for companies aiming to develop innovative systems efficiently [DAR+21].

Introducing Systems Engineering in a company often requires thorough training, especially for new employees who may not yet have the essential Systems Engineering skills. While these employees might have learned the theory during their university studies, applying this theory at work can be challenging because each company has its own unique processes, methods, and tools. Therefore, effective training should be customized to address these specific needs, often including coaching and organizational change management techniques [DAR+21].

Training needs within a company vary depending on employees' roles and responsibilities [SEB24]. SHEARD identified 12 key roles in Systems Engineering, each requiring different competencies [She96]. However, these predefined roles don't always align with the diverse and evolving needs of modern organizations. To bridge this gap, KÖNEMANN et al. introduced a more flexible role clustering approach that better reflects the different stakeholders involved in Systems Engineering [KWA+22]. Their work provides a structured way to map roles within a company, define responsibilities across various ISO processes, and determine the specific competencies needed for each role. This framework helps organizations align their training programs with actual job requirements, ensuring employees develop the right skills to support effective Systems Engineering practices.

Wright emphasizes the importance of competency assessment in organizations, highlighting its role in effectively managing evolving skill requirements. As jobs and the products being developed continuously evolve, organizations must regularly assess employee competencies to ensure their workforce remains equipped with the necessary skills. This ongoing evaluation enables companies to adapt to industry changes, bridge skill gaps, and maintain efficiency in an environment where technological advancements and market demands are constantly shifting [Wri24].

KÖNEMANN et al.'s [KWA+22] adaptable framework provides a structured methodology for assessing Systems Engineering competency levels within organizations. Their approach establishes a clear relationship between the competencies required for various roles, the processes these roles are responsible for, and the proficiency levels necessary to execute each process effectively. By defining target competency levels for different roles, the framework ensures that employees develop the skills to perform their assigned responsibilities successfully.

The proposed Systems Engineering competency assessment system in this master thesis leverages the work of KÖNEMANN et al. to enables organizations to evaluate employee competencies based on their roles and responsibilities. By assessing the alignment between required and actual competency levels, organizations can identify skill gaps and areas for improvement. This insight allows for targeted training initiatives, ensuring employees develop the necessary expertise to perform their roles effectively. Additionally, the system could also support broader organizational analysis since competency data is gathered at the individual level. Companies can use this data to assess overall competency distribution across teams, optimize workforce planning, and refine role assignments.

KÖNEMANN et al.'s [KWA+22] framework provides a structured approach to assessing competencies based on predefined role clusters. However, organizations often have unique internal role definitions that do not directly align with these clusters [INC23b], making competency assessment more challenging. To address this, a mechanism is needed to map employees' tasks and responsibilities to relevant ISO processes. By leveraging the relationships between ISO processes, activities, and tasks outlined in [?], employees' work activities can be systematically linked to the corresponding ISO processes. Once these mappings are established, KÖNEMANN et al.'s framework can be applied to derive the required competency profile. In the new age of Large Language Models (LLMs) and Generative AI [VSP+23], automating complex tasks such as process mapping has become feasible [WM24]. These models excel at processing natural language [RWC+19] and have demonstrated advanced reasoning capabilities [PWV+24], making them well-suited for handling the variability and complexity in systems engineering competency assessment. In this thesis, we investigate how LLMs can be used to automate process mapping by analyzing employees' tasks and aligning them with the relevant ISO processes. By combining LLM-driven process mapping with KÖNEMANN et al.'s competency framework, we aim to develop a system that efficiently determines competency requirements based on actual work activities.

Beyond competency assessment, KÖNEMANN et al.'s [KWA+22] framework could also be extended for a data-driven approach to role recommendation. Since the framework defines the competency levels required for each role, it can be used to match individuals with the roles they are best suited for based on their assessed competencies. By comparing an individual's current competency profile from the assessment with the required levels for different roles provided by [KWA+22], a recommendation system could be developed to suggest the best roles for the individual. This would allow employees to explore career progression paths within their organization while also helping companies optimize role assignments based on skill fit. Integrating such a recommendation system into competency assessment would enhance workforce planning, ensuring that employees are placed in roles where they can perform effectively while also identifying key areas for upskilling.

2 Problem Statement

2.1 Problem Analysis

Assessing the right competencies within an organization is essential for ensuring employees have the skills needed to support the company's goals. Research shows that performance improves when there is a strong alignment between the competencies of an individual and their job requirements [BMA20] [JKM13]. In systems engineering, the INCOSE Systems Engineering Competency Framework is widely recognized as a key model to define the necessary competencies that Systems Engineers should have [sec23].

However, implementing this framework in practice presents several challenges. BEASLEY emphasizes that the effectiveness of Systems Engineering competency models depends on their adaptability to different organizational contexts [Bea13]. KASER et al. [KFZ+10] highlight that the Systems Engineering Competency Framework(SECF) was originally designed with an emphasis on objective assessment methods, primarily based on practices observed among systems engineers in the UK. Consequently, applying this framework in diverse organizational environments can be complex, requiring significant adaptation.

The Systems Engineering Competency Assessment Guide (SECAG) extends this framework by providing structured guidelines for assessing Systems Engineering competencies while allowing customization for specific organizations. SECF defines 37 competencies, each structured into four proficiency levels. SECAG provides with approximately ten indicators per level and multiple sub-indicators for each of the competencies defined in SECF [INC23b]. These indicators span various dimensions, including knowledge, experience, behaviors, and evidence that practitioners at each of the proficiency level should display, offering a comprehensive basis for competency assessment. However, for many organizations, this level of granularity can be overwhelming. Tailoring the framework to specific roles requires assessors and competency designers to manually navigate through extensive competency indicators, a process that is not only time-consuming but also susceptible to subjectivity and inconsistencies in assessment.

Addressing these complexities and limitations, KÖNEMANN et al. [KWA+22] proposed a structured and more simpler approach for mapping roles to competencies and processes in Systems Engineering. Their framework not only simplifies this mapping but also defines the specific competency levels that different stakeholders should attain within an organization. The required competency levels are systematically derived from the ISO processes that each Systems Engineering role is responsible for, ensuring a clear and logical connection between job responsibilities and skill requirements. This structured approach makes the framework adaptable, as modifying role-process assignments automatically adjusts the corresponding role-competency values, allowing organizations to tailor it to their specific needs.

While KÖNEMANN et al.'s [KWA+22] framework provides a structured and adaptable method for mapping roles to competencies and processes, it remains largely at a theoretical level. There is currently no complete practical implementation, software, or scalable tool that automates this approach, making it challenging for organizations to efficiently apply it in real-world settings. Without an automated system, conducting Systems Engineering competency assessments remains a time-intensive process that requires significant manual effort. This lack of practical implementation limits the framework's accessibility, especially for organizations that need to perform large-scale assessments with limited time and resources. To fully realize the potential of this structured approach, there is a need for a scalable software solution that can automate competency assessments, streamline role mapping, and provide actionable insights for workforce development.

In a previous master's thesis by JAVAID, a survey methodology was developed that allowed individuals to select their role, complete a survey, assess their competencies, and compare the results with the target competencies for that role [Jav23]. The implementation utilized the techniques outlined in KÖNEMANN et al.'s [KWA+22] work to define target competencies for each role cluster. However, the implementation has several shortcomings that could be further improved and enhanced.

Firstly, the survey questions designed to assess each competency exhibit common pitfalls, as mentioned by [CP05]. Improving these questions is essential, and it is also necessary to scientifically validate whether the set of questions accurately captures the intended competencies.

Currently, the methodology can only assess the competencies of employees who can identify their roles from a predefined set of role clusters. However, in real-life scenarios, these predefined roles may not always align with the specific organizational context of a company, as companies often define roles in unique ways that do not directly map to established role clusters [DAR+21, INC23b]. According to the Helix project [HWL17], "a systems engineering role is defined as a specific set of related systems engineering activities." The Helix project derived an updated set of systems engineering roles by examining the activities performed by employees, highlighting the importance of task-based role definition [HWL17]. Therefore, a similar flexible approach needs to be introduced, allowing employees to select tasks and responsibilities that reflect their actual work within the company. These tasks can then be mapped directly to relevant ISO processes, which, in turn, will inform the appropriate competency profile based on the established framework [KWA+22]. This ensures that the assessment is tailored to each company's specific organizational needs.

Furthermore, the Helix research revealed that employees often take on responsibilities from more than one systems engineering role within organizations [HWL17]. This creates a challenge for existing competency assessment methodologies, which typically assess employees based on a single predefined role [Jav23]. To address this issue, a more flexible approach is needed, one that allows systems engineers to be evaluated across multiple roles. This will ensure that the competency assessment accurately reflects the diverse range of tasks and responsibilities employees undertake in their day-to-day work.

Once a structured competency assessment system is established and the expected competency levels for each role are clearly defined [KWA+22], its application can extend beyond evaluation. The assessment results can be leveraged to match employees to the roles that best align with their skills by comparing their competency profiles against predefined role competency mappings. This would enable organizations to make more informed role assignments, ensuring that employees are placed in positions where they can perform effectively. This approach would enhance workforce efficiency while also providing employees with clearer career progression paths and targeted skill development opportunities.

Finally, a competency assessment system could also be expanded to collect and store employee competency results, allowing for deeper organizational and team-level analysis. This would give companies valuable insights into their workforce's strengths, skill gaps, and areas that need improvement. By examining this aggregated data, organizations can make better decisions on workforce planning, identify training needs, and develop targeted skill-building programs. This approach not only helps in closing competency gaps but also supports long-term strategic planning, ensuring the organization remains competitive and well-equipped to handle evolving challenges [SSD+24].

2.2 Field of Actions

Based on the issues in the Problem Analysis 2.1 section, there is a need for a more flexible and comprehensive systems engineering competency assessment framework or system.

Based on the findings in 2.1, two key fields of action have been defined: **improving the survey methodology for systems engineering competency assessment** and **enhancing the overall competency assessment process**. These action fields address the existing approaches' limitations and aim to provide a more accurate, tailored, and scalable solution for assessing systems engineering competencies. The following table 2-1 outlines these fields of action along with a brief description of the proposed improvements.

Table 2-1: Identified Fields of Action from Problem Description

No	Field of Action	Description
1	**Improve Survey Methodology for SE Competency Assessment**	Refine and validate survey questions to ensure they accurately assess systems engineering competencies. Include comparisons with established frameworks like INCOSE SECAG to benchmark improvements.
2	**Enhance Competency Assessment**	Adapt the competency assessment process to reflect the unique needs of organizations. This includes mapping tasks to relevant ISO processes, identifying the required competencies for those processes, supporting assessments across multiple roles, suggesting optimal roles based on competency results, and providing both individual and organizational-level competency analysis.

2.3 Objectives

The primary objective of this thesis is to create a verified and validated systems engineering competency assessment solution that allows for company-specific tailoring. This methodology will utilize the techniques developed by KÖNEMANN et al. [KWA+22] to determine the ideal target competencies for each systems engineering role. The aim is to combine these techniques with an implementation that supports organizational flexibility, enabling employees to map their tasks and responsibilities to relevant ISO processes, which can then be used to identify the appropriate competencies. By doing so, the solution will derive the necessary competencies for each role based on the processes employees engage in, facilitating tailored competency assessments. Additionally, this solution will address the need for multi-role assessments and organizational-level competency analysis, providing companies with valuable insights into both individual and overall competency standing. The primary objectives are listed below:

2.3.1 Survey Methodology Improvement

The following objectives focus on improving the effectiveness and accuracy of the survey questions used in the competency assessment:

- **Develop a refined set of questionnaires** Improve upon the limitations of the survey questionnaires from JAVAID [Jav23] by designing a new set using scientifically grounded methods, such as cognitive and psychometric science. These refined questionnaires will accurately capture the competency levels of respondents in alignment with the competency levels defined by KÖNEMANN et al. [KWA+22].
- **Validate Competency Level Definitions:** Assess the competency level definitions provided by KÖNEMANN et al. [KWA+22] against the standards outlined in the INCOSE Systems Engineering Competency Framework (SECF) [sec23] and the Systems Engineering Competency Assessment Guide (SECAG) [INC23b] to ensure consistency with industry standards and alignment with established evaluation criteria.
- **Validate Through Expert Feedback:** Ensure the final set of questions correctly assesses the intended competencies by conducting expert reviews and making iterative refinements based on their feedback.

2.3.2 Competency Assessment Enhancement

This enhancement aims to develop a Systems Engineering Competency Assessment framework with features that better assess and analyze competencies in Systems Engineering. The new system should be designed to incorporate the identified enhancements from the problem analysis section and should focus on tailoring competency assessment to both individual and organizational needs:

- **Assess Competencies with Known Roles:** Evaluate the competencies of individuals who can identify their roles from a predefined set of SE role clusters from KÖNEMANN et al. [KWA+22].
- **Assess Competencies with Unknown Roles:** Develop a matching mechanism that maps the tasks and responsibilities provided by individuals to relevant ISO processes for those who cannot identify to a predefined SE role. Once the ISO processes are identified, the required competencies should be determined, and the survey proceeds to assess these competencies.

- **Suggest Ideal SE Roles:** Based on the competency assessment results, recommend the most suitable roles for employees, aligning their skill sets with the roles that best match their competencies and responsibilities.
- **Handle Multiple Roles:** Enable the system to account for individuals performing more than one SE role. Also, calculate a competency profile that reflects the combined requirements of these roles.
- **Provide Aggregated Competency Analysis:** Offer both individual and company-level aggregated competency analysis, highlighting each level's strengths and weaknesses. This analysis should compare actual competencies against ideal target competencies, providing actionable insights for individual and organizational development.

By achieving these objectives, this thesis aims to create a more flexible and comprehensive systems engineering competency assessment framework, addressing the shortcomings of the previous work and providing valuable insights for both individual employees and organizations.

Figure 2-1 illustrates the Systems Engineering (SE) Competency Assessment Framework, which is designed around two key objectives: Competency Assessment Enhancement and Improved Survey Methodology. The core framework, based on the principles of KÖNEMANN et al. [KWA+22], enables users to select a role cluster, which then determines the target competencies relevant to their role. The system facilitates competency assessment by allowing users to evaluate their skills against predefined benchmarks. Users can review their individual results, while administrators gain insights into team and organizational competency standings, supporting data-driven decision-making for workforce planning and development.

The top section introduces enhancements that extend the framework's capabilities. These include:

- Support for multiple role assessments allows individuals to evaluate themselves against multiple roles.
- Assessment for unknown roles, enabling users to determine where their competencies align without pre-selecting a specific role.
- Optimal role recommendations, where the system suggests the most suitable roles based on assessment results.

The bottom section focuses on improving the survey methodology used to assess competencies. This enhancement involves developing a new set of survey questionnaires based on scientific principles from cognitive and psychometric research. Experts validate these questionnaires to ensure they accurately assess Systems Engineering competencies. This framework integrates competency assessment and survey methodology improvements to provide a structured, scalable, and scientifically validated approach to assessing SE competencies at both individual and organizational levels.

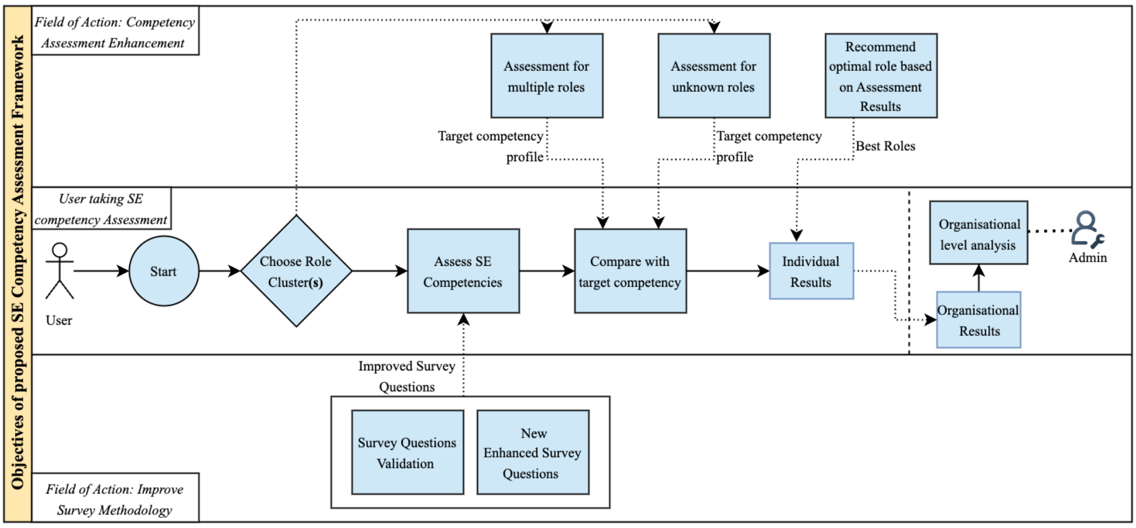


Figure 2-1: Overview of the SE Competency Assessment Framework, illustrating the two primary fields of action: Competency Assessment Enhancement and Survey Methodology Improvement. This diagram outlines key steps for mapping user tasks to ISO processes, supporting multi-role assessments, providing aggregated results, and refining survey questions to improve survey effectiveness.

3 Foundation

In order to effectively understand and implement the work presented in this thesis, it is essential to establish a strong theoretical foundation. This section introduces the key concepts, frameworks, and studies that underpin the research, providing the necessary background for competency assessment in Systems Engineering (SE). Additionally, it covers relevant software engineering concepts, Generative AI methodologies, and scientific techniques that contribute to the successful implementation of the Systems Engineering Competency Assessment Framework.

3.1 INCOSE Systems Engineering Competency Framework

A competency framework provides a structured model for defining the essential skills, knowledge, and behaviors required for individuals to excel in their roles within an organization or a specific domain [CIP19]. According to Holt and Perry (2011), competency is "a measure of an individual's ability in terms of their knowledge, skills, and behavior to perform a given role" [HP11].

In the field of Systems Engineering (SE), the INCOSE Systems Engineering Competency Framework (SECF) [sec23] defines the core competencies necessary for designing, developing, and managing complex systems. The International Council on Systems Engineering (INCOSE) [Int23a] developed and published SECF to provide a standardized competency model that ensures consistency in skill expectations, professional development, and Systems Engineering practices across industries.

The SECF categorizes competencies into five key areas:

- **Core Competencies –** Fundamental skills required across all engineering roles.
- **Professional Competencies –** Skills related to collaboration, communication, and ethical considerations.
- **Management Competencies –** Leadership and project management skills for overseeing engineering efforts.
- **Technical Competencies –** Domain-specific engineering knowledge required for system design and implementation.

- **Integrating Competencies –** Skills necessary for integrating systems, processes, and teams effectively.

Each competency within these categories is further defined by multiple proficiency levels, which help identify an individual's expertise in a specific area. SECF outlines competency levels ranging from "Awareness" (basic understanding) to "Expert" (highest level of proficiency). To provide clear differentiation between levels, SECF defines competency level descriptions along with specific behavioral indicators that professionals at each level should demonstrate. Typically, each level includes 8 to 10 indicators, which serve as measurable criteria for evaluating an individual's competency progression.

Additionally, SECF provides guidelines on how organizations can tailor the framework to meet their specific requirements and needs. This adaptability allows companies to align competency expectations with their unique engineering processes, workforce structures, and business goals.

3.2 KÖNEMANN et al.'s Stakeholder-Specific Systems Engineering Competencies

KÖNEMANN et al. introduce a framework for identifying stakeholder-specific competencies in Systems Engineering (SE) [KWA+22]. Their research focuses on defining competencies that align with specific roles and responsibilities within an organization, ensuring that competency assessments are relevant to the actual needs of different stakeholders. The framework is structured around five key entities:

3.2.1 Role Clusters

To capture all relevant stakeholders and roles in Systems Engineering, KÖNEMANN et al. validated job titles from industry and grouped them into 14 role clusters. Each role cluster serves as a grouping or aggregation of similar roles found in various industries. This provides a structured way to categorize job roles in Systems Engineering.

Figure 3-1 illustrates a sample of these role clusters, demonstrating how different positions within an organization can be mapped into broader categories.

3.2.2 Competencies

According to GELOSH et al., organizations should tailor their competency models based on their unique requirements and workflows, rather than adopting a universal model

TABLE II. ROLE CLUSTERS

Role Cluster Name	Description
Customer	represents the party that orders or uses a service (= development order). The customer has influence on the design/technical execution of the system.
Customer representative	forms the interface between the customer and the company. The roles in this cluster form the voice for all customer-relevant information required for the project.
Project manager	is responsible for the planning and coordination on the project side. The roles assume responsibility for achieving the project goals and monitoring the resources (time, costs, personnel) within a time-limited framework and also have a moderating role in conflicts and disputes.

Figure 3-1: Sample of role clusters from KÖNEMANN et al.'s [KWA+22]

Figure 3-2: KÖNEMANN et al.'s extended INCOSE competency framework [KWA+22]

[GHS+17]. While the INCOSE SECF provides a comprehensive list of 37 competencies, KÖNEMANN et al. propose a simplified and extended competency model consisting of 5 competency areas and a total of 16 competencies.

This model is considered simplified because it selects only 16 out of the 37 competencies defined in SECF, focusing on the most relevant areas for practical application. It is also extended, as it introduces new competencies, such as agile competencies, which are essential in modern development processes.

Figure 3-2 provides an overview of the simplified and extended competency model proposed by KÖNEMANN et al., highlighting the key competencies used in their framework.

3.2.3 Role Process Matrix

The Role-Process Matrix establishes a mapping between role clusters defined by KÖNEMANN et al. and the ISO 15288 processes [**?**] used in Systems Engineering. This matrix provides a structured way to identify the level of involvement each role cluster has in performing different SE processes, ensuring that competency assessments align with actual responsibilities in an organization.

The matrix assigns one of four distinct values to indicate the degree of involvement a role has in a given process:

- **0** – The role is not involved in performing the process.
- **1** – The role supports the process but is not primarily responsible for its execution.
- **2** – The role is responsible for performing the process.
- **3** – The role is responsible for designing the process within the organization.

These values were determined by analyzing industry practices, studying corporate structures, and validating the framework with industry partners as part of [KWA+22]. By providing a clear role-to-process mapping, this matrix helps organizations identify which competencies are required for each role based on their actual process responsibilities.

In Figure 3-3, the top-left matrix represents the Role-Process Matrix, where individual role clusters are mapped to their respective levels of responsibility within different ISO 15288 processes.

3.3 Process Competency Matrix

The Process-Competency Matrix establishes a mapping between ISO 15288 processes and the competencies described in Section 3.2.2. This matrix defines the level of importance of each competency in executing a given ISO 15288 process, ensuring that competency assessments align with the actual skill requirements needed for process execution.

The matrix assigns one of three values to indicate the relevance of a competency to a specific process:

- **0** – The competency is not needed to perform the process.
- **1** – The competency is supportive, meaning it is helpful but not essential.
- **2** – The competency is mandatory, indicating it is crucial for successfully executing the process.

To develop this matrix, KÖNEMANN et al. utilized competency mappings from the INCOSE Systems Engineering Competency Framework (SECF) along with insights gathered from industry workshops. These workshops involved industry professionals who helped validate and refine the competency requirements for different processes.

In Figure 3-3, the top-right matrix represents the Process-Competency Matrix, where each ISO 15288 process is mapped to the competencies required for its successful execution.

3.4 Role-Competency Matrix

The Role-Competency Matrix defines the competencies required for each role cluster to effectively perform its assigned ISO 15288 processes. KÖNEMANN et al. derive this matrix by combining the Role-Process Matrix and the Process-Competency Matrix, multiplying their respective values and selecting the maximum value for each role-competency combination [KWA$^+$22].

By applying this method, the matrix assigns a competency level to each role based on Bloom's Taxonomy:

- **Knowing (1)**– Basic awareness of the competency.
- **Understanding (2)** – Ability to comprehend and explain concepts.
- **Using (3) –** Capability to apply the competency in practice.
- **Mastering (6) –** High-level expertise in the competency.

This matrix helps determine the level of competency each role requires in various areas, ensuring that professionals have the appropriate skills for their responsibilities. For example, a role may need only "Knowing" level proficiency in one competency while requiring a higher level, such as "Using" or "Mastering", in another.

In Figure 3-3, the bottom matrix represents the Role-Competency Matrix, which is derived from the Role-Process Matrix and the Process-Competency Matrix. This final mapping provides a structured competency profile for each role, helping organizations define skill expectations and design targeted training programs.

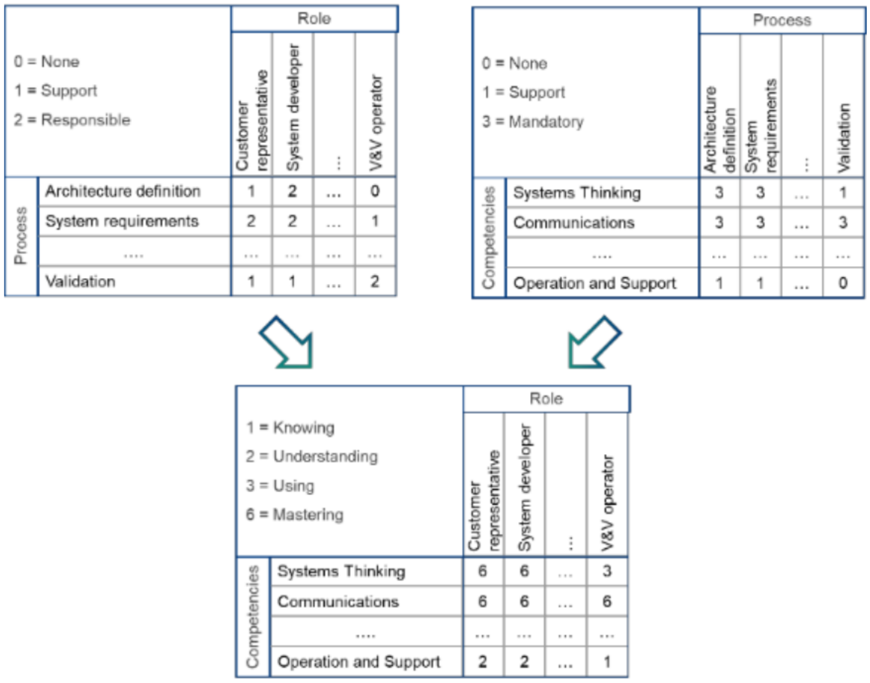


Figure 3-3: Derivation of the Role-Competency Matrix from the Role-Process and Process-Competency Matrices. The top-left matrix represents the Role-Process Matrix, which maps roles to their level of involvement in ISO 15288 processes. The top-right matrix represents the Process-Competency Matrix, which maps ISO 15288 processes to the competencies. The bottom matrix is the Role-Competency Matrix, derived by multiplying the values in the Role-Process Matrix and the Process-Competency Matrix, then taking the maximum value for each role-competency pair. This matrix provides a structured way to determine the competencies required for each role.[KWA+22]

3.5 Retrieval Augmented Generation

Generative AI (Gen AI) is a powerful technology that enables computers to produce text, images, videos, and other creative content in response to human input [Lv23]. According to LIM et al., Gen AI models process user prompts and generate responses based on their pre-trained knowledge, aiming to provide the most relevant output [LGP+23]. However, Large Language Models (LLMs) [VSP+23], which power most Gen AI systems, often face challenges such as hallucinations—where the model generates incorrect or misleading information [XJK24].

To mitigate this issue and enhance response reliability, Retrieval-Augmented Generation (RAG) has been introduced by LEWIS et al. as a method to combine the capabilities of LLMs with external knowledge sources. RAG allows AI models to retrieve relevant, factual data from predefined sources before generating a response, ensuring that the output is more accurate and grounded [LPP+21].

3.5.1 How RAG Works

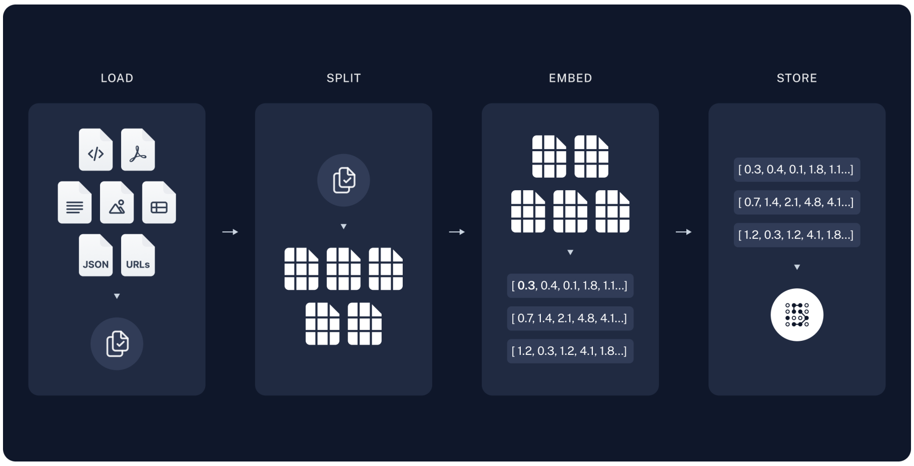


Figure 3-4: Indexing process in Retrieval-Augmented Generation (RAG)[LPP+21], as illustrated in LangChain website, a widely used LLM application development framework[Lan].

Langchain, a widely used Python-based LLM application development framework, explain RAG in two main stages [lan23]:

1. **Indexing: Building a Knowledge Base** Before an LLM can use retrieval-based augmentation, a structured vector store must be created, containing relevant knowledge in a searchable format. A vector store is a database of numerical representations (vectors) of text that allows for efficient similarity searching. The steps involved in building a vector store include:
   - **Data Collection:** Gathering all relevant documents, texts, or structured data that might be useful for answering user queries. This ensures that the model has access to reliable external knowledge.
   - **Chunking:** Splitting the collected data into smaller, meaningful text segments. Instead of storing full-length documents, chunking allows the system to retrieve only relevant portions of the data when needed [VNK24].

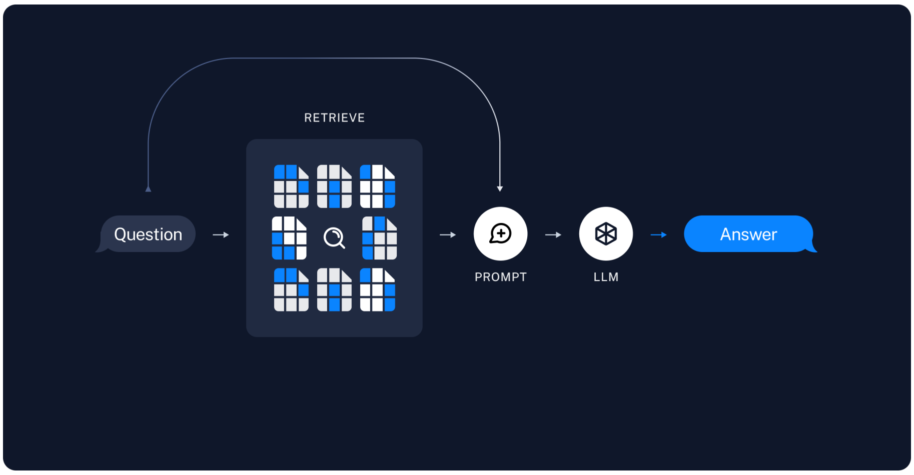


Figure 3-5: Retrieval and generation process in Retrieval-Augmented Generation (RAG) [LPP+21], as illustrated in LangChain, a widely used LLM application development framework[lan23].

- **Embedding:** Converting text chunks into vector representations—numerical embeddings that capture the semantic meaning of the content. This transformation is necessary for efficient similarity-based retrieval [MCC+13].

- **Storing in a Vector Database:** The embedded chunks are stored in a vector database (such as ChromaDB [chr25] or FAISS [DGD+25]), where they can be quickly retrieved based on their relevance to user queries.

2. **Retrieval and Generation** Once the knowledge base is set up, the RAG pipeline consists of two key processes:

   - **Retrieval:** When a user submits a query, the system searches the vector store to find semantically similar text chunks using similarity search techniques. Instead of relying solely on the model's pre-trained knowledge, the retrieved documents serve as a factual reference for the AI's response.

   - **Generation:** The retrieved documents are combined with the user's query and passed as input to the LLM. The model then generates a response that is grounded in the provided documents, significantly reducing the risk of incorrect or misleading answers.

3.5.2 Why RAG is Effective?

LLMs excel in understanding and reasoning [BMR+20] and demonstrate domain generalization and adaptability across various fields [RWC+19]. However, integrating RAG ensures context-aware question answering, grounding responses in real-world knowledge and reducing hallucinations [LPP+21].

In this thesis, RAG plays a crucial role in organizational tailoring of KÖNEMANN et al.'s framework[KWA+22]. One key objective is to automate task-to-ISO-process mapping by leveraging LLMs with relevant ISO 15288 documentation [**?**]. When users provide task descriptions in natural language, the system retrieves ISO process documentation as context, enabling the LLM to identify and map the tasks to the appropriate ISO processes.

3.6 Basic Working of a Web Application

In software development, a web application consists of three main components: frontend, backend, and database.

The frontend is the user interface (UI) that users interact with through a web browser or mobile browser. It is typically built using frameworks like Vue.js [Vue14] or React[Met13].

When a user interacts with the frontend, their input is sent to the backend, which processes the request, executes application logic, and retrieves or updates data stored in the database. The backend is commonly implemented using frameworks such as Flask (Python based) [Fla] or Node.js (JavaScript) [Nod09], while databases like PostgreSQL [Pos96] or MySQL [MyS95] store and manage application data.

This interaction between the frontend, backend, and database ensures seamless functionality in web applications, enabling users to perform various operations efficiently.

4 Related Works

4.1 INCOSE Systems Engineering Competency Framework (SECF)

The INCOSE Systems Engineering Competency Framework (SECF) [sec23] is the standard competency model for Systems Engineering (Section 3.1). In addition to defining competencies, SECF provides guidelines for assigning competencies to roles using ARCIFE models (responsibility model derived from RACI [Ins17]), which identify the required competency levels for each role. SECF also outlines methods for tailoring roles to organizational needs, stating that companies have unique competencies and processes and implement Systems Engineering differently [sec23].

However, KÖNEMANN et al. identified several limitations in SECF [KWA+22]:

- Agile competencies are not considered despite their importance in modern development.
- No structured method exists to map competencies to stakeholders, making role definitions inconsistent.
- Professional competencies are not linked to specific processes, limiting their practical application.
- Processes like support and qualification are not considered, leaving gaps in role coverage.

To address these gaps, KÖNEMANN et al. developed an improved stakeholder-specific competency framework that refines SECF by integrating missing competencies, establishing a structured role-to-process mapping, and aligning professional competencies with specific ISO processes [KWA+22].

4.2 KÖNEMANN et al.'s Stakeholder-Specific SE Competencies

The stakeholder-specific Systems Engineering competency model proposed by KÖNEMANN et al. builds upon existing competency frameworks, addressing the limitations of

SECF and LESLIE [Les16] competency models [KWA+22]. Since this thesis is based on KÖNEMANN et al.'s framework, the fundamental concepts have already been explained in the foundation section (Section 3.2).

While the model provides a structured approach for mapping roles to competencies and ISO 15288 processes, it remains conceptual. Therefore, we need an implementation mechanism to apply it in real-world scenarios. Additionally, the method focuses solely on role mappings to ISO 15288 processes without accounting for organizations that use alternative role definitions, such as RACI models [KWA+22].

This thesis aims to address these gaps by implementing the framework practically and extending its applicability to organizations that use custom role models. By integrating automated role process mapping mechanisms, the proposed solution ensures that competency assessments align with both ISO 15288 and other organization-specific processes.

4.3 INCOSE's Systems Engineering Competency Assessment Guide

The INCOSE Systems Engineering Competency Assessment Guide (SECAG) provides a structured method for assessing the 37 competencies defined in the Systems Engineering Competency Framework (SECF) [INC23b]. SECF categorizes competencies into five proficiency levels ranging from Awareness to Expert, with approximately 7-10 indicators per competency level.

SECAG expands on SECF by defining knowledge, experience, and behaviours associated with each competency level indicator. Additionally, it provides examples of activities and behavioural evidence that individuals should demonstrate at each level. This has already been introduced in the problem statement (Section 2.1).

Despite its structured approach, SECAG has significant challenges:

- **High Complexity –** SECAG builds upon SECF's 37 competencies, each with five proficiency levels, resulting in 185 competency entries. Since each level includes 7-10 indicators, this results in approximately 1,541 indicators in total[INC24].
- **Extensive Data Representation –** SECAG further extends these indicators by defining knowledge, activities, and professional attitudes for each entry, resulting in 4,751 additional entries. It also provides evidence examples, bringing the total dataset to over 3,081 evidence points. Managing this manually is highly complex and inefficient[INC24].

- **Static Representation –** SECAG and similar frameworks we discussed are mostly static; they are currently documented using spreadsheets and textual documents, making navigation, searching, and customization for different organizations difficult and impractical. This static representation makes it hard to update and manage, especially as feedback is incorporated and the framework evolves[INC24].

- **Lack of Software Implementation –** Due to the above reasons, tailoring SECAG for organizational roles and job descriptions requires significant time, resources, and expertise in competency assessment. Due to its complexity and high cost of manual implementation, no standardized software solution exists.

This thesis addresses these challenges by developing an automated system that supports Systems Engineering competency assessments, making the process scalable, efficient, and easily adaptable for different organizational needs.

4.4 INCOSE Systems Engineering Competency Assessment Guide SysML Model

To address the limitations of SECAG discussed in the previous section 4.3, AMENABAR et al. developed a digital model-based tool using Systems Modeling Language (SysML) [Sys25] to implement SECF and SECAG[INC24]. This model formalizes relationships and traceability within the competency framework by defining links between competency indicators and evidence. It also enables organizational tailoring of roles and job descriptions while aligning with SECAG. Additionally, the model integrates with INCOSE's Professional Development Portal (PDP) [INC22] for competency management.

Despite its advantages, this approach presents several challenges:

- **Complexity –** The digital model remains difficult to navigate due to the inherent complexity of SECAG.

- **Requirement of SysML/MBSE Knowledge -** Users need expertise in SysML or Model-Based Systems Engineering (MBSE) to utilize the tool effectively.

- **Tooling Limitations –** The implementation relies on specialized SysML tools like Cameo Systems Modeler, which require licensing and additional costs and may have integration challenges with other enterprise tools.

- **Lack of a User Interface for Self-Evaluation –** No graphical user interface (GUI) is provided for individual competency assessment, requiring additional development efforts.

- **Maintenance and Adoption Effort –** Organizations need to invest in training, change management, and ongoing maintenance to transition from traditional documentation- and spreadsheet-based approaches.

These limitations highlight the need for a more accessible, user-friendly, and scalable solution for competency assessment and organizational tailoring, which this thesis aims to address.

4.5 Systems Engineering Competency Methodology and Functional/Domain Competency Assessment Tool at BAE Systems Inc.

DANO developed a validated competency assessment tool for evaluating Systems Engineering and functional domain expertise at BAE Systems Inc [Dan19][BAE24]. The tool was designed as a self-assessment system, implemented using spreadsheets and Python, to measure engineers' competencies in both lifecycle-based SE skills and specialized domain-specific competencies tailored to the business area in which they operated.

The competency model was based on the INCOSE Competency Development of SE Practitioners model [RC03] and was adapted to BAE Systems' specific requirements by modifying the set of competencies to align with internal business needs.

The assessment methodology is as follows:

- A rubric/scoring guide was developed with capability levels ranging from 1 to 5, indicating the degree of training and experience required for a given role.

- Competency expectations were mapped one-to-one to the company's Global Grades (GG) system. For example, a junior engineer (GG10) was expected to meet a lower capability level than a senior engineer (GG13 or GG15).

- Engineers self-assessed their competency levels across multiple competencies. The lowest-scoring competency (a ”weakest link” approach) determined the overall competency level, ensuring that critical gaps were identified and prioritized.

- The final assessment result was compared with the individual's Global Grade. If the assessed competency level was below the expected level for their grade, it indicated the need for additional training, mentoring, or job rotation to close the gap.

The tool was tested twice with participants, and feedback was used for iterative improvements. Results showed that the tool effectively highlighted competency gaps, supporting individual development and managerial oversight. It also provided actionable insights into training and experience shortfalls, making it a useful resource for career progression and skills management.

This methodology has some limitations that we should consider:

- **Broad Competency Definitions –** Some participants found the domain competency descriptions used for the self-assessment are too generic[Dan19].
- **Reliance on Self-Assessment –** Engineers might overestimate or underestimate their skills, affecting accuracy.
- **Limited Sample Validation –** The beta tests involved a small group of participants, limiting generalizability.
- **Adapting to Other Organizations –** The tool relied on the BAE System's internal Global Grades, making it difficult to apply in other organizational settings.
- **Dynamic Industry Needs –** As technology evolves, the tool requires regular updates to keep its competency models and training recommendations relevant.

This research provides useful insight into competency assessment methodologies but remains highly specific to BAE Systems. In contrast, this thesis aims to develop a more adaptable and scalable approach that can be applied across different organizations with varying competency models and role definitions.

4.6 INCOSE Certification Programs

The International Council on Systems Engineering (INCOSE) offers Systems Engineering Professional (SEP) certification programs to validate individual competencies and skills in Systems Engineering [INC25]. These certifications include:

- **Associate Systems Engineering Professional (ASEP) –** Entry-level certification assessing foundational SE knowledge.

- **Certified Systems Engineering Professional (CSEP) –** Certification for experienced practitioners, requiring demonstrated SE knowledge and application.
- **Expert Systems Engineering Professional (ESEP) –** The highest certification level, requiring INCOSE membership, professional references, and an interview assessing technical leadership experience. References may also be interviewed as part of the evaluation.

While these certifications help individuals improve their professional profiles and career prospects, they have limitations:

- **Organizationis not considered –** The focus is on individual certification without addressing how organizations implement SE competencies.
- **Role-Process relationships are not considered –** The certifications do not define how competent an individual should be to perform specific processes (example: ISO 15288) at work.
- **Roles are not factored into competency validation -** There is no structured mapping between certification levels and the competencies required for different SE roles.

This thesis addresses these gaps by developing a structured competency assessment framework that evaluates individuals in the context of their roles, responsibilities, and organizational processes.

4.7 Survey Methodology by JAVAID

The Master's thesis by Javaid [Jav23] implemented KÖNEMANN et al.'s competency framework [KWA+22] as a web application. However, as discussed in the problem analysis chapter 2.1, this implementation was incomplete and had several limitations:

- **Assessment limited to a single role per person –** Users might be performing more than one role in an organization.
- **No organizational tailoring –** The system lacked features for customizing roles or competencies to fit specific organizational structures.

- **Survey-based approach with unvalidated questionnaires –** The competency assessment relied on a survey, but the questionnaires were not scientifically validated, raising concerns about assessment accuracy.

These shortcomings highlight the need for a more comprehensive solution that supports multi-role assessments, organizational tailoring, and validated competency measurement, which this thesis aims to address.

5 Design

In this section, we provide a detailed overview of the design of our web application for assessing Systems Engineering (SE) competency, addressing the limitations and challenges identified in previous sections. We examine the System's Functional Architecture from the user's perspective, focusing on how individuals assess the application. Next, the section explains the survey paths available to the users: Role-based assessment, Task-based assessment, and Full comprehensive assessment. Following this, we shift to the admin perspective, detailing the preparatory steps required to configure the system before surveys can be conducted. This includes an in-depth discussion on data models that administrators must configure, ensuring the system is set up to capture role-specific and organization-specific competency requirements. We then present the complete SE competency assessment workflow, outlining how users interact with the system and what happens in the system while users complete their evaluations. The final section discusses the design of the survey questionnaires by explaining how they are structured to ensure accurate competency measurement and incorporating validated scientific approaches to improve assessment reliability.

5.1 System Functional Architecture: User Perspective: Performing the Assessment

The SE competency assessment web application is designed to evaluate the competency levels of Systems Engineers based on their roles and responsibilities within an organization. The system provides users multiple pathways to assess their competencies, ensuring flexibility and adaptability to diverse organizational structures and individual career trajectories. Figure 5-1 illustrates the functional overview of the system from the perspective of a user taking the survey. It depicts the core processes and interactions involved in competency assessment.

The SE competency assessment process begins when the user initiates the survey and selects one of three available pathways. In the Role-Based Competency Assessment, users select one or more roles from the predefined role clusters. The system determines the required competency levels for performing the selected roles. The user's responses are

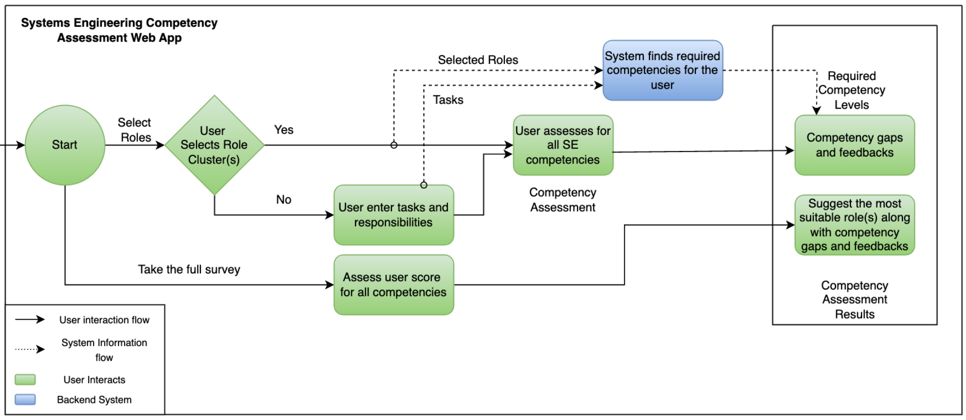


Figure 5-1: Functional Overview of the Competency Assessment System from User Perspective

then evaluated against these predefined requirements, allowing for a direct comparison of recorded competencies with those necessary for the selected role(s).

In the Task-Based Competency Assessment, users who cannot directly map their job role to the predefined role clusters provide details about their job tasks and responsibilities, categorized by their level of involvement (e.g., responsible for, supporting, or designing). The system then maps these tasks to relevant ISO processes using an intelligent inference mechanism, deriving the necessary competencies for performing these tasks, against which the user is assessed.

In the Full Competency Assessment, users comprehensively evaluate all competencies within the framework. Based on their survey responses, the system analyzes their competency profile and suggests the most suitable roles, identifying those that best align with their demonstrated competencies. The following sections provide a detailed explanation of each of these three assessment pathways.

5.1.1 Survey Paths and and their Logic

5.1.1.1 Role-Based Competency Assessment

In this pathway, users have the option to select one or more roles from a predefined set of 14 role clusters from the KÖNEMANN et al.'s methodology[KWA+22]. If a user's current role at work aligns with one or more of these predefined clusters, they can select the corresponding role(s) for competency assessment. The system determines the required competencies for each role using the Role–Competency Matrix, which specifies the necessary competency levels for effectively performing a given role (refer Section 3.4).

For users selecting multiple roles, the system identifies the competency requirements necessary to perform the selected roles. This ensures that individuals who take on multiple responsibilities receive a comprehensive evaluation that accurately reflects their professional requirements.

5.1.1.2 Task-Based Competency Assessment

This survey pathway is for a user who cannot identify their role within the 14 predefined role clusters. These users can input their job tasks, responsibilities, and level of involvement. The involvement levels are categorized as *Not Applicable, Supporting, Responsible, or Designing*. Based on this input, an intelligent mechanism should be utilized to map the tasks to the relevant ISO processes.

Once the system identifies the ISO processes performed by the user, it utilizes them in combination with the Role-process matrix and the Process-competency matrix to derive the required competencies for performing this unknown role using the Role-Competency Matrix derivation explained in section 3.4.

This approach ensures that the competency assessment remains flexible and inclusive. It accommodates users from diverse organizational backgrounds where predefined role titles may not accurately reflect their actual responsibilities [DAR+21] [INC23b]. Additionally, some organizations may use role names that differ entirely from the role cluster names defined in the system[SH22]. In such cases, the task-based assessment allows users to map their work responsibilities to competency requirements without relying on predefined role classifications. This ensures that the evaluation remains relevant and accurate, regardless of the specific terminology used within an organization.

5.1.1.3 Comprehensive Competency Assessment

In this survey path, the system can assess all competencies for users seeking a complete SE competency evaluation without requiring prior role selection. Users are evaluated across the entire set of 16 competencies defined in the system. Based on the assessment results, the system determines the user's best role(s) by comparing their competency profile with the Role-Competency matrix. This approach ensures that users receive role recommendations aligning with their demonstrated competencies. It also provides insights into potential roles that best match their skills and knowledge.

This assessment method is particularly valuable for organizations, as it helps assign roles to new employees or restructure teams based on individual competency profiles. Using this data-driven approach, organizations can make informed decisions about workforce planning and ensure employees are placed in roles where their skills are best used.

5.2 System Functional Architecture: Admin Perspective: Preparation for the Assessment

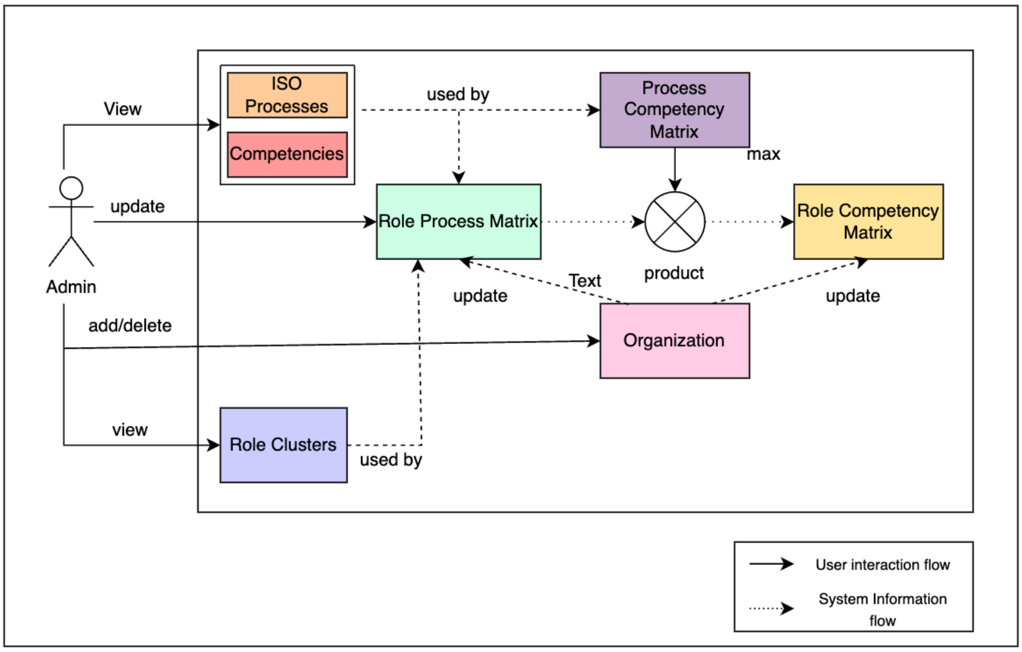


Figure 5-2: Functional Overview of the Competency Assessment System from Admin Perspective and Data Models

Figure 5-2 illustrates the high-level functional architecture of the competency assessment system from an administrative perspective. It highlights the system's key data models that administrators must set up to prepare it for effective SE competency evaluation.

The SE competency assessment app is designed to offer a structured and adaptable framework that allows administrators to manage and configure essential components. These include the Role-process matrix and the Process-competency matrix, which are derived from core foundational elements such as ISO processes, competencies, role clusters, and organizational configurations. Together, these components form the fundamental data models for the SE competency assessment framework, enabling a standardized competency evaluation process in diverse roles and organizational contexts.

The system administrator configures and manages the components required for competency assessment, as illustrated in Figure 5-2. This includes defining ISO processes and competencies, which are the foundations for competency evaluation. These elements are then used to establish the Role-Process matrix and Process-Competency matrix, which are used to determine the competency requirements for different roles 3.4. Since the Role–Competency Matrix is conceptually derived from the Role–Process Matrix and the Process–Competency Matrix, any updates to these foundational matrices require recomputation of the Role Competency Matrix.

To support multiple organizations with various competency requirements, a corresponding Role–Process Matrix is defined for that organization whenever a new organization is introduced into the system. The administrator, in collaboration with the organizational representative, is responsible for configuring this matrix by defining the roles and ISO processes performed by the roles within the organization. This ensures that the competency assessment framework accurately reflects the organization's structure and job functions.

The system is designed to support evolving organizational needs. By structuring assessments around well-defined matrices from KÖNEMANN et al's framework[KWA+22], the system provides a standardized yet flexible approach to competency evaluation.

The following section discusses the data models in Figure 5-2. The administrator should configure these data models to prepare the system for SE competency evaluation. Additionally, we explain how these models are interconnected and designed to depend on each other to ensure a structured and consistent competency assessment framework.

5.2.1 Overview of Data Models

Figure 5-2 illustrates the key data models of the SE competency assessment system and their interdependencies. These models serve as the foundation for the SE competency evaluation and define the relationships between roles, processes, and competencies[KWA+22]. The administrator is crucial in managing and updating these elements to align the system with organizational structures and competency requirements. The administrator is responsible for configuring and refining these data models to ensure the assessment framework remains relevant and effective. In the following subsections, we describe each data model and its role in the competency assessment process.

ISO Processes and Competencies

Our framework for assessing SE competencies is built on a predefined set of ISO processes [Int23b] and SE competencies 3.2.2, which serve as a foundation for evaluating SE competencies within organizations.

- **ISO Processes** Data Model defines the standardized engineering processes organizations follow to ensure compliance with industry regulations and best practices [Int23b]. The administrator configures this model by adding the required ISO processes to the system. These processes form a fundamental part of the competency assessment framework, establishing the relationship between roles and required competencies.

  Since ISO processes are derived from established standards, they are considered static and are not expected to change frequently. Any modification to this model would require adjustments to the Role–Process Matrix and the Role–Competency Matrix to maintain consistency in competency evaluations. However, in the current design, modifications (such as removing a process from the system) to ISO processes are not supported, as they are assumed to remain stable. The administrator's role is limited to viewing all ISO processes within the system, ensuring transparency in the competency framework.

  Future system enhancements could introduce flexibility for updating ISO processes if necessary. If implemented, any modifications must trigger automatic recalculations of dependent matrices to maintain assessment accuracy. While this feature is not part of the current design, the system architecture allows for potential future extensions to support evolving industry requirements.

- **Competencies** Data Model defines the skills and knowledge areas required for effectively performing systems engineering processes. The competencies in the system are based on the INCOSE Systems Engineering Competency Framework which have been reworked by KÖNEMANN et al[KWA+22] (Subsection: 3.2.2). Competencies in this system are categorized into distinct competency areas: Core, Socio-Personal, Managerial, and Personal. These categories ensure a structured and comprehensive evaluation of competencies across different aspects of systems engineering functions.

  Since these competencies are derived from established standards, they are considered static and require minimal modifications. As a result, the current design does not allow for modifications (such as removing a competency from the system), and the administrator is only provided with a viewing option for transparency. This approach ensures consistency across competency assessments while maintaining alignment with industry standards.

However, the system architecture is designed with the potential for future flexibility. If customization is introduced in future enhancements, any modifications to the Competency Data Model would need to automatically propagate to all dependent matrices, including the Process–Competency Matrix and the Role–Competency Matrix and the recomputation of Role-Competency Matrix, ensuring that competency assessments remain accurate and up to date.

Role Clusters

The Role Cluster Data Model defines 14 predefined role clusters identified from the SE4OWL research project [KWA+22]. These clusters group related job roles within systems engineering, providing a structured approach to competency mapping by linking roles to relevant ISO processes and competencies. More on this is explained in the foundation section of this thesis(Subsection 3.2.1).

In the current design, role clusters are static, meaning the administrator can only view them without making any modifications. This is based on the presumption that no near future changes are expected in the Role Clusters. However, the system architecture is designed with the potential for future flexibility. Future implementation could include features to add or remove Role Clusters to the system. Any future modifications to Role Clusters would directly impact the Role–Process Matrix, which defines the relationships between roles and ISO processes. To maintain consistency and accuracy in competency mapping, such modifications would require careful management to ensure they accounts for the matrix dependencies.

Role–Process Matrix Data Model

The Role–Process Matrix is designed to support the configuration of role involvement levels in ISO 15288 processes, as defined in Section 3.2.3. By default, the system applies predefined values from KÖNEMANN et al. [KWA+22] as the standard reference for competency assessments. The system design allows administrators to configure this matrix by assigning involvement levels for each role–process combination, selecting from:

- **Not Relevant - 0:** The role does not participate in the process.
- **Supporting - 1:** The role provides assistance in performing the process.
- **Responsible - 2:** The role is primarily accountable for the process execution.
- **Designing - 3:** The role actively involves designing and improving the process.

Considering organizational requirements, the system is designed to support both individual and organization-specific competency assessments. For individual assessments, the default Role–Process Matrix is used. For organization-specific assessments, an initial matrix is generated based on default values but can be modified by administrators (together with organizational representatives) to reflect actual organizational role-process involvements.

The system is also designed to automatically recalculate the Role–Competency Matrix whenever changes are made in role-process values. This ensures that the competency assessments remain accurate and aligned with updated role process definitions. Adjustments to role-process values are applied only within the relevant assessment context, individual or organizational, to maintain consistency and adaptability in competency evaluation.

Process–Competency Matrix Data Model

The Process–Competency Matrix Data Model maps SE competencies and ISO 15288 processes, as explained in Section 3.3. It establishes the competency levels required for executing each ISO process, ensuring alignment with industry standards and organizational needs.

The SE competency assessment system is designed to allow administrators to configure this matrix by assigning competency levels for each competency–process pair, selecting from:

- **Not Useful - 0:** The competency is not required for the process.
- **Useful - 1:** The competency is beneficial but not essential for process execution.
- **Necessary -2:** The competency is critical for process execution.

By default, the matrix is pre-filled with values derived from KÖNEMANN et al. [KWA+22], ensuring consistency with established SE competency frameworks. However, administrators have the flexibility to update these values as needed.

Since the Process-Comptency Matrix directly influences the Role–Competency Matrix, any modifications made by the administrator trigger an automatic recalculation, ensuring assessments remain accurate. As the competency-to-process relationships remain consistent across organizations, this matrix is static by default and is not considered to change between organizations. This means that the Process–Competency Matrix is not organization-specific and does not vary between organizations like the Role-Process matrix.

Role–Competency Matrix Data Model

The Role–Competency Matrix Data Model is designed to define the competency requirements for each role, as explained in Section 3.4. It is derived by multiplying the Role–Process Matrix and the Process–Competency Matrix, assigning the highest competency requirement across all relevant processes:

$$Role\_Competency\_Value = \max\left(Role\_Process\_Value \times Process\_Competency\_Value\right)$$

The computed competency levels are categorized as follows:

- **0 – Not Relevant:** The competency is not required for performing the role.
- **1 – Knowing:** The role requires basic awareness and familiarity with the competency
- **2 - Understanding:** The role requires a conceptual understanding of the competency.
- **3/4 – Applying:** The competency is needed for practical application in work-related tasks.
- **6 – Mastering:** The role demands expertise and leadership in the competency.

The system is designed to automatically update the Role–Competency Matrix whenever changes are made to the Role–Process Matrix or Process–Competency Matrix, ensuring competency assessments remain accurate.

For organization-specific assessments, a custom Role-Process matrix is defined with default values from [KWA+22] whenever a new organization is added, and a corresponding Role-Competency matrix is generated for the newly added organization. Whenever the organizational-specific Role-Process matrix is modified to address the diverse role-process mapping in organizations, the corresponding Role-Comptency matrix is also recomputed. This ensures that competency mapping reflects organization-specific role definitions while aligning with industry standards.

For individual users, the system applies a default Role–Competency Matrix, based on predefined competency requirements from KÖNEMANN et al. [KWA+22], ensuring standardized competency assessments.

The system dynamically recomputes the Role–Competency Matrix in the following cases:

- When the Role–Process Matrix or Process–Competency Matrix is modified.
- When a new organization is added. A new Role-Competency Matrix is generated for this organization based on the values in the organizational-specific Role-Process matrix.

This design ensures scalability, consistency, and adaptability, allowing competency assessments to reflect real-time organizational needs while remaining structured and aligned with existing standards.

Organizational Configuration

The Organizational Configuration Data Model supports organization-specific competency models, ensuring that competency assessments reflect each organization's structure and role expectations. When a new organization is added, the system automatically generates a dedicated Role–Process Matrix, initially populated with default values from KÖNEMANN et al. [KWA+22]. The systems also automatically generate A Role-Competency matrix based on the default values in the Role-Process matrix.

To allow flexibility, administrators can modify the Role–Process Matrix in collaboration with organizational representatives, ensuring role-process associations align with actual work practices. Since the Role–Competency Matrix is derived from the Role–Process Matrix, any modifications trigger an automatic recalculation in the organizational context to maintain accuracy.

The system also distinguishes between individual users and organization-affiliated users:

- Individual users are assessed using the default Role–Competency Matrix.
- Organization-affiliated users are evaluated based on the organization-specific Role-Competency matrix.

This ensures tailored and relevant competency assessments, aligning with industry standards and organization-specific requirements.

5.3 SE Competency Assessment Survey Workflow Design

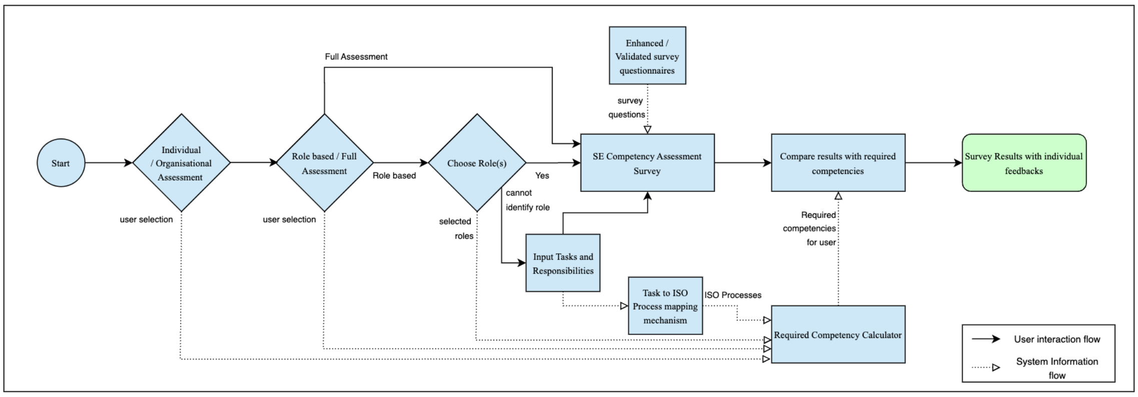


Figure 5-3: Workflow of the SE Competency Assessment Survey

Figure 5-3 illustrates the structured workflow design of the SE Competency Assessment System. It outlines the different assessment pathways available to users while taking the SE competency assessment survey. The system supports three types of competency assessments: role-based assessment, task-based assessment, and full competency assessment. These distinct survey pathways provide flexibility to users by enabling them to select the most relevant evaluation method based on their professional responsibilities and assessment goals.

The assessment begins with the user selecting whether they participate as an individual or part of an organization. This selection is passed to the Required Competency Calculator component, which calculates the survey user's competency requirements based on user selections. The Required Competency calculator uses this organization information to consider the correct organizational context when determining the target competency requirements for the user. Since different organizations may have distinct competency expectations, this step ensures that assessments remain relevant to the user's professional environment.

Users then choose between role-based assessment and full competency assessment. In the full competency assessment, users proceed directly to the survey questionnaire, where they are evaluated across all available competencies in the system. Instead of comparing responses against predefined role requirements, the system analyzes the user's competency profile and mathematically tries to suggest the most suitable roles based on their demonstrated proficiency. This approach helps users explore roles that best match their skills and identify potential career paths.

In the role-based assessment, users select one or more predefined role clusters that align with their job roles. The Required Competency Calculator then uses this information to determine the competency benchmarks from the Role–Competency Matrix (3.4), identifying the expected proficiency levels required to perform the selected role(s). The user's competencies level recorded from the survey is then assessed against these competency requirements.

If users taking the role-based assessment cannot map their job role to any of the 14 predefined role clusters, they can opt for the task-based competency assessment. In this approach, users provide details of their tasks and responsibilities as natural language responses, categorized into responsible for, supporting, or designing. An intelligent mapping mechanism processes this information to identify the ISO processes performed by the user based on the user input. Based on these identified processes, the system derives the necessary competency requirements for performing these processes utilizing the role-process, process-competency and role-competency mappings defined in [KWA+22]. This ensures that users who do not fit into predefined role clusters receive a competency assessment tailored to their job functions.

Once the survey is completed, users receive personalized competency results. In the role-based and task-based assessments, the results compare the user's recorded competency levels with the required levels for the selected role(s) or identified performing processes. In the full competency assessment, the system identifies the best-matching roles based on the user's competency assessment by analyzing the user's competency profile and mapping it to the Role–Competency Matrix. This allows users to explore roles closely aligning with their current skill set.

The system also provides individualized feedback, offering insights into competency strengths and areas for improvement. Users can download their results, supporting further professional development planning.

From an organizational perspective, the system enables administrators to analyze competency data across the organization. Administrators can access individual scores and conduct aggregated analyses to evaluate the organization's overall competency standing. This capability helps organizations identify skill gaps, develop targeted training programs, and align workforce competencies with strategic objectives.

5.4 Design of Survey Questionnaires

The SE Competency Assessment System survey questionnaires evaluate user SE competency levels using a set of well-defined questionnaires. The questionnaires are designed using a structured, science-based approach to ensure accurate and meaningful competency measurements.

The competency levels assessed by the questionnaire align with KÖNEMANN et al. [KWA+22], applying Bloom's Taxonomy [Blo56] to classify competencies into hierarchical categories:

- **Not Relevant –** The competency does not apply.
- **Knowing –** Basic awareness of the competency.
- **Understanding –** Conceptual knowledge and explanation.
- **Applying –** Practical execution of competency in tasks.
- **Mastering –** Expertise and leadership in the competency.

To ensure structured competency level progression in questionnaires, the questionnaire uses principles from Bloom's Taxonomy [Blo56] and the Dreyfus Model of Skill Acquisition [DD80]. Bloom's Taxonomy and the Dreyfus Model organize knowledge into progressing levels, ensuring clear differentiation between competency levels.

5.4.1 Psychometric and Survey Design Principles

The questionnaire incorporates psychometric principles to enhance accuracy in competency assessment:

- **Item Response Theory (IRT)[Bak01] -** Questions are structured in increasing order of difficulty, meaning selecting a higher competency level implies proficiency in all lower levels.

- **Guttman Scaling [Eng05] –** Responses follow a hierarchical structure, ensuring logical progression in skill assessment. The questionnaire follows a structured format where respondents select from predefined hierarchical competency groups—ranging from basic (Group 1) to advanced (Group 4), with an additional *Not Relevant* option (Group 5). Although it does not use a traditional Likert scale [Lik32], it follows the principles of Guttman scaling [Eng05], ensuring that responses reflect a structured competency progression.

Additionally, the survey design also considers the common survey biases identified by Choi and Pak [CP05], and tries to minimize them.

5.4.2 Cognitive UX Considerations

To optimize usability and reduce response bias, the questionnaire incorporates cognitive UX principles:

- **Hick's Law [Hic52] –** Limits the number of response choices to minimize cognitive overload and decision fatigue.
- **Dual-Process Theory [Kah11] –** Balances intuitive (fast) and analytical (slow) thinking: Quick selections for familiar competencies and Deeper thinking needed for complex competencies levels.

By scientific approaches from psychometric principles, as well as cognitive UX techniques, the survey tries to ensure a structured, reliable, and bias-minimized competency assessment.

5.5 Key Design Considerations

The system is designed with core principles that ensure an effective SE competency assessment system. The key factors guiding the design include:

- **Scalability:** The system design can accommodate multiple organizations with distinct competency requirements. New organizations could be easily added to the system with minimal effort. The system automatically creates distinct matrices with default values to be used in the context of the newly created organization. This automated approach minimizes configuration needs and ensures seamless integration of new organizations into the competency assessment framework.

- **Flexibility:** The framework adapts to evolving organizational structures and competency needs without affecting existing competency evaluations. The design allows admins to reconfigure the role-process and process-competency mappings easily. The system would automatically identify the changes and trigger the recomputation of the Role-Competency matrix. Future-proofing measures have been considered to accommodate industry standards and business objectives changes.
- **Consistency:** Ensuring alignment with established ISO standards and competency frameworks to provide reliable and standardized assessments in different organizational contexts.
- **Usability:** The design prioritizes an intuitive and user-friendly interface for administrators and survey takers, allowing efficient management of the SE competency assessment system and easy-to-navigate survey workflow.
- **Maintainability:** The modular design allows easy configuration of the components in the SE Competency Frameworks. The design also considers the possible future improvements that might be required on the system.
- **Security:** The system does not collect personal information to assess user SE competency levels.
- **Scientific Validity of Survey Questions:** The survey design follows scientifically validated methodologies for competency assessment. The questionnaires are designed based on Bloom's Taxonomy and the Dreyfus Model to assess competency levels. Additionally, psychometric principles such as Item Response Theory (IRT) are considered to ensure accurate and structured competency evaluations. Furthermore, cognitive UX science methods are considered to reduce the cognitive overload and decision fatigue that could arise from survey question design choices. The question design also tries to minimize the common biases that could happen in the survey[CP05].

By structuring the competency assessment system around these design principles, the framework remains robust, adaptable, and capable of addressing evolving future requirements. The system architecture is intended to provide a comprehensive and scalable solution for managing and evaluating competencies in systems engineering.

6 Implementation

Implementing the Systems Engineering Competency Assessment app uses the system design guidelines outlined in the previous chapter. This section explains how the conceptual framework is converted into a fully functional web application. It provides a complete overview of the technologies, frameworks, and methodologies employed to develop the system, ensuring scalability, flexibility, and maintainability.

The system is implemented using a modern and most widely used technology stack for frontend, backend, database, and LLM framework implementations. Components are implemented as separate modules, such as the Administrative module, where the administrator configures the application for SE competency assessment; the Survey Engine, where the user takes the survey; and the Results and Feedback Module, where individual and organizational survey results are displayed and analyzed.

Generative AI capabilities are integrated to solve complex problems that required more human work in the past. This AI-driven approach enables the intelligent mapping of user-defined tasks and responsibilities to ISO processes, ensuring personalized and accurate competency evaluations based on the tasks performed at work. Additionally, the system uses Generative AI models to offer individualized, personalized assessment feedback based on the user's survey results. This allows for a more user-centric experience, helping individuals identify areas for growth and development based on their unique job functions and career aspirations.

To address the security and data privacy concerns, the SE competency assessment app is designed to process survey results without capturing any personal information. This approach is maintained throughout the implementation, ensuring compliance with industry standards and protecting user privacy.

The following sections discuss the technology stack, core system functionalities, security mechanisms, and strategies to overcome implementation challenges.

6.1 Technology Stack and Framework

This section provides an overview of the technology stack used to implement the SE competency assessment system's front-end, back-end, LLM application development frameworks, database, and deployment infrastructure. These tools are carefully selected to ensure ease of use, community support, performance, and maintainability. The choice of technologies and frameworks is driven by the need to provide a seamless user experience, efficient back-end processing, and robust data management.

6.1.1 Front-End Framework

In developing the SE competency assessment system, **Vue 3**[Vue25] was selected as the front-end framework due to its relative ease of learning and developer-friendly features. Research indicates that Vue has a gentler learning curve compared to other widely used frameworks such as Angular and React, making it an ideal choice for those new to front-end development[LZ21]. Additionally, Vue's *component-based architecture* facilitates modular development by allowing functionalities to be broken down into reusable components[ZJL23][SZX19]. This approach enhances code maintainability and scalability, making it easier to manage complex applications[ZJL23]. Vue's progressive framework design further allows developers to incrementally adopt its features, providing flexibility in aligning the front-end implementation with evolving project requirements[SZX19]. Other notable advantages of Vue include its reactive data binding, virtual DOM rendering for improved performance and comprehensive documentation that aids faster development and troubleshooting[Vue23].

For managing application state, **Pinia** [Pin19] was adopted as the state management solution. State management is essential in applications like survey web applications to maintain a consistent flow of data across different components[DJK24]. In this SE Survey App, Pinia stores user details and survey responses before the final submission. This enables users to navigate through the survey, review their responses, and make modifications before submission. Pinia was chosen based on recommendations from experienced developers and is now officially recognized as the primary state management library for Vue 3[Pin19].

6.1.2 Backend-end framework

In developing the competency assessment system, **Flask** was chosen as the backend framework primarily due to my prior experience with Python. Flask is a lightweight and highly adaptable web framework that facilitates rapid development, making it well-suited for small and medium-sized projects [Zan24]. Its minimalist design allows developers to focus on core functionality without the constraints typically associated with more complex frameworks [Zan24]. This flexibility is advantageous for applications requiring iterative enhancements, as Flask supports seamless integration of additional features and components over time. This flexibility also helps follow an agile development process for this thesis, where features are added with time in different iterations[SB02][SANS12].

Moreover, Flask's modular architecture helps to maintain the simplicity of the system [Zan24]. Flask's modular approach helps to ensure that only the necessary components are included, helping maintain a lean and efficient application and enhancing performance. Additionally, Flask benefits from a large and active open-source community and comprehensive documentation, which facilitates troubleshooting and encourages adherence to best development practices[Fla25a][Pal25]. These attributes make Flask ideal for building our SE competency assessment app's maintainable and efficient backend.

6.1.3 LLM Development Framework

For integrating generative AI functionalities within the SE Competency Assessment System, LangChain [Lan] was chosen as the primary LLM development framework. Several factors influenced this decision:

- **Structured Output Capabilities:** At the time of development, LangChain provided structured output functionalities[Lan24], which were essential for generating JSON-formatted responses. Since structured output is crucial for seamless communication between the front-end and back-end of our web app, LangChain was a suitable choice.
- **Strong Community Support:** LangChain is one of the leading frameworks for developing LLM-based applications and retrieval-augmented generation (RAG) systems, with over 100K stars on GitHub at the time of writing[Lan25a]. The framework is observed to have continuous improvements, frequent updates, and extensive documentation[Lan25a].
- **Python-Based Framework:** Since the back-end of our web app is implemented in Flask, which is Python-based, choosing LangChain, which is also Python-based, maintained uniformity in the tech stack. This choice facilitated seamless integration with Flask APIs and enabled efficient pipeline development.
- **Advanced Chaining Mechanisms:** The intelligent task mapping mechanism in the system involves multiple steps that need to be executed in a structured manner. These steps involve processing user inputs, performing lookups, doing RAG

(Retrieval Augmented Generation) and generating structured outputs, all requiring seamless coordination.

Since these processes involve multiple interconnected LLM operations, LangChain's chaining mechanisms [Lan24] provided a structured way to orchestrate these steps efficiently. This capability made LangChain a natural fit for building the generative AI capabilities of our web app.

6.1.4 Database

In developing the SE competency assessment web app, PostgreSQL[Pos96] was chosen as the database management system due to my prior experience. PostgreSQL is a widely used, free, open-source object-relational database known for its reliability and advanced capabilities. It is an open-sourced database management system, making it a cost-effective solution with extensive community support[Pos96]. Additionally, PostgreSQL offers seamless integration with Docker since it is available as an official Docker image [Doc25]. This simplifies the deployment process and enables the containerization of applications which can reduce the web app's deployment effort. Using the docker-image, the application could be containerized and easily deployed on any infrastructure. While another open-source relational database, MySQL[MyS95], was considered an alternative, PostgreSQL was ultimately selected due to prior experience working with it.

6.1.5 Deployment Infrastructure

The SE competency assessment system is deployed within a virtual machine (VM) environment to provide a controlled and isolated infrastructure. A virtual machine (VM) offers dedicated resources that improve system performance, security, and reliability [Der24]. The primary components of the web app are — Flask for the back end, Vue.js for the front end, and PostgreSQL for data management. They are containerized using Docker [Doc22]. Containerization encapsulates the web app and its dependencies into lightweight, portable environments (called containers)[Mer14]. This ensures consistency across different environments and simplifies deployment. Each component runs within its container, providing modularity and ease of maintenance[Mer14]. Docker Compose [Doc24] is used to manage and coordinate these containers efficiently. Docker Compose enables seamless orchestration of multi-container applications within the VM environment[Doc24].

Additionally, to support the generative AI functionalities in the web app, API calls are made to *OpenAI's GPT-4o mini* (chat model) [Ope24a] and *text-embedding-ada-002* (embedding model) [Ope24b]. These AI capabilities enhance the SE competency assessment web app by enabling intelligent task process mapping and personalized feedback generation. This deployment approach follows modern DevOps practices, fostering an agile and robust infrastructure for efficient operations and future scalability.

6.2 Implementation of Core Functionalities

The SE Competency Assessment System consists of two primary modules: the Administrative Management Module, which admins use to configure and prepare the system for competency assessments, and the Survey Module, which facilitates user assessments by dynamically determining required competencies based on initial user selections. These modules ensure seamless system setup and competency evaluation for individuals and organizations.

The Administrative Management Module helps admins configure the fundamental data models that serve as the foundation of the competency assessment framework. These include the Role–Process Matrix, the Process–Competency Matrix, and Organizational Configurations. Administrators populate and manage these matrices to ensure accurate competency mapping and organizational alignment. The Role–Process Matrix and the Process–Competency Matrix are essential for automatically computing the Role–Competency Matrix that gives the required competencies for roles. Additionally, administrators can add or remove organizations to the system. This enables tailored SE competency assessments that is specific to each organization's role-process mappings and competency requirements. The system design also allows for future modifications to competency models, ensuring adaptability to evolving needs.

The Survey Module governs the user assessment process and determines the required competencies based on the selected survey path. Users can take different assessment routes, including role-based, task-based, and complete competency assessments. Implementing this module involves dynamically calculating target competency levels based on user choices—whether selecting predefined roles, inputting job-related tasks, or undergoing a full competency evaluation.

The following sections provide a detailed explanation of the technical implementation of these core functionalities, including database interactions, API logic, and system workflows, ensuring efficient and accurate competency assessments.

6.2.1 Administrative Management Module: Preparation of SE Competency Assessment

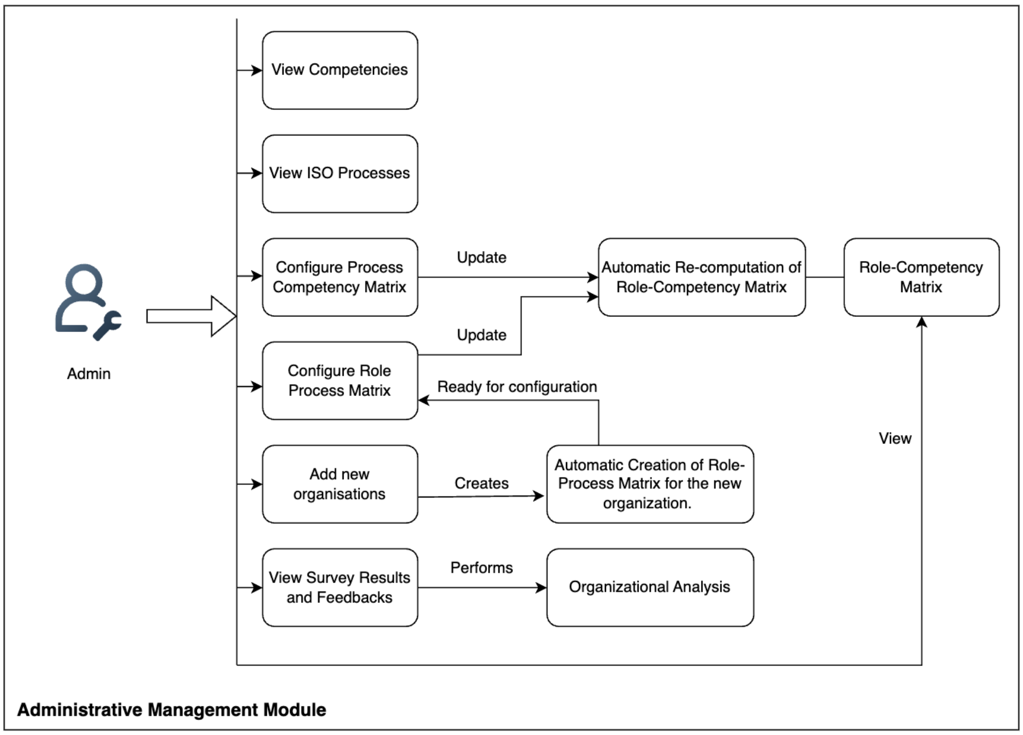


Figure 6-1: Admin tasks for preparing the system for competency assessment

The Administrative Management Module is a core component of the SE Competency Assessment System. This module enables administrators to configure and manage the fundamental data models that drive competency evaluation. It implements the functionality outlined in the design section (5.2) and is accessible exclusively via a password-protected web interface, ensuring secure and restricted access to administrative tools.

Figure 6-2 illustrates the Administrative Management Module and its key functionalities. The module provides capabilities such as viewing competencies and ISO processes, configuring the Process-Competency and Role-process matrices, Viewing the Role-Competency matrix for individuals and organizations, adding new organizations, and performing organizational analysis based on survey results.

The system is designed to automatically trigger updates where necessary—for example, modifying the Process–Competency Matrix or the Role–Process Matrix leads to the automatic recalculation of the Role–Competency Matrix in the context of the organization, ensuring that competency evaluations remain consistent with the latest configurations.

In the following sections, we explain each functionality the administrator performs to prepare the SE Competency Assessment System for competency evaluations.

6.2.1.1 Competency Management

The Competency Management functionality lets administrators view all the competencies in the system that facilitate the assessment. These competencies form the foundation of the SE Competency Assessment System and many other data models using competencies for their derivations.

The system provides administrators with a read-only interface to view all competencies in the system. The UI lists all competencies and categorizes them into different competency areas. This allows administrators to review the competency framework without making direct modifications.

When an administrator accesses the competency management interface, the front end retrieves competency data via API calls to the backend, which fetches the required information from the database. The system has all 16 competencies from [KWA$^+$22] 3.2.2 prefilled. The current implementation does not encourage modifications since competencies are derived from established industry frameworks. However, the system design remains open to future enhancements, and features could be introduced to add or remove competencies from the framework.

By maintaining a fixed competency structure, the system ensures consistency in assessments while allowing administrators to efficiently reference standardized competency definitions.

6.2.1.2 ISO Process Management

The ISO Process Management functionality provides administrators with read-only access to all the ISO Processes used in the SE Competency Assessment System and their descriptions. These processes represent the standardized engineering practices in the industry [Int23b]. This will ensure that our SE competency assessment follows industrial standards. The current implementation allows administrators to view all ISO processes available within the system and their descriptions. The interface retrieves this data from the database through API calls, ensuring that administrators can review processes without making modifications.

The system can be extended in future iterations to include adding and removing processes to the system. However, this change must ensure that any addition or removal of processes should automatically trigger the following:

- The creation of corresponding entries in the Role–Process Matrix and Process–Competency Matrix.
- The automatic recalculation of the Role–Competency Matrix to reflect the updated processes.

The static design ensures that the the competency assessments remain consistent with predefined industry standards while leaving room for future adaptability.

6.2.1.3 Process–Competency Matrix Configuration

The Process–Competency Matrix defines the relationship between ISO 15288 processes and the competencies required for their execution, as explained in Section 3.3. It serves as one of the two foundational matrices for deriving the Role–Competency Matrix.

Pre-population and Modifiability

The matrix is pre-populated with values from KÖNEMANN et al. [KWA+22], ensuring consistency with industry frameworks. While it remains largely static, the system allows administrators to update competency relevance levels if needed. Any modification automatically triggers an update to dependent Role-Competency matrix calculations to maintain accuracy.

Administrator Interface and Backend Logic

Administrators manage the Process–Competency Matrix through a Vue. js-based interface, enabling:

- Viewing predefined competency mappings.
- Assigning relevance levels (0, 1, or 2) to each ISO process using checkboxes:
  - **0: Not Relevant** – The competency is not required.
  - **1: Useful** – The competency is beneficial but not essential.
  - **2: Necessary** – The competency is critical for process execution.
- Saving updates trigger an automatic recalculation of the Role–Competency Matrix.

The backend (Flask) processes each modification, updating the database and ensuring that all role competency values reflect the latest competency relevance. This implementation provides flexibility for future refinements while maintaining the integrity of the competency assessment framework.

6.2.1.4 Organizational Management

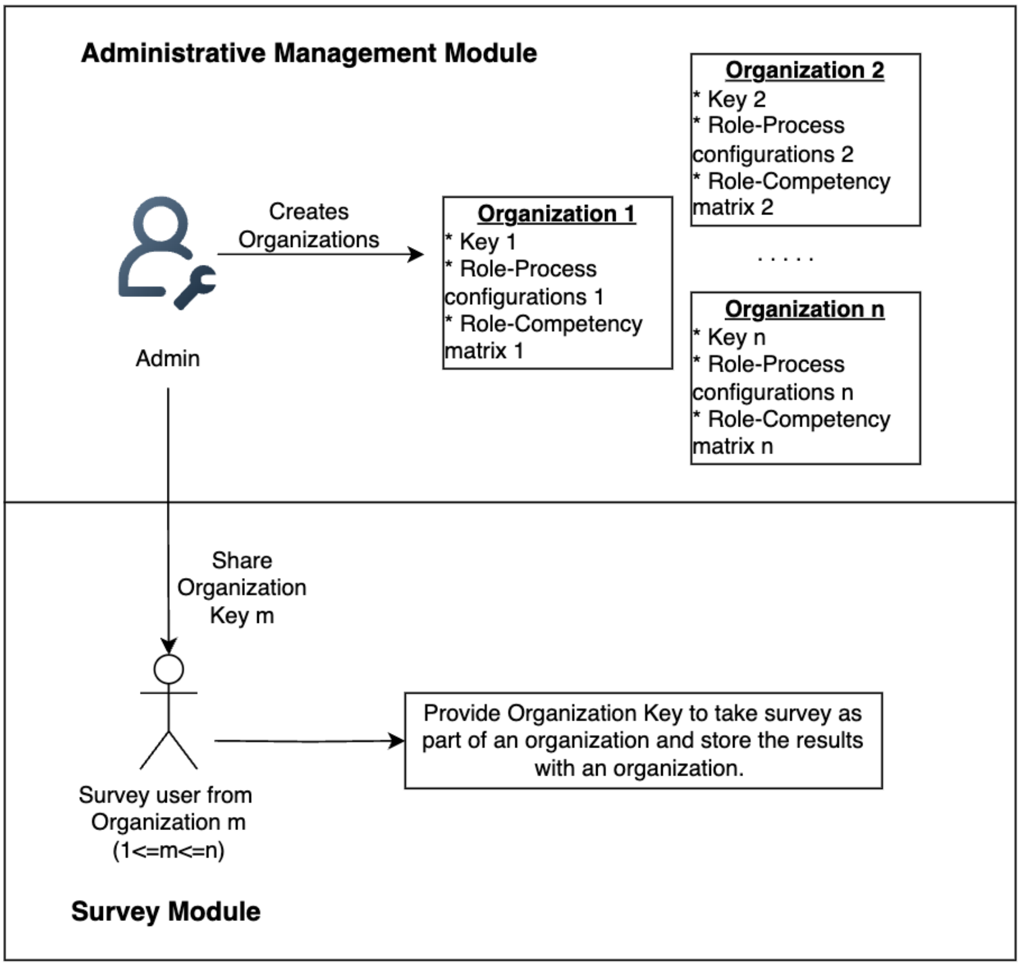


Figure 6-2: Organizational Management: Admin Creating and Configuring Organizations

The Organizational Management module enables administrators to create and manage organizations within the SE Competency Assessment System. The administrator interface provides a dedicated UI where admins can enter the organization name and assign a unique organization key to create an organization. This key is later shared with users who wish to take the survey as part of that organization, ensuring that their responses are correctly associated with their respective organizations and evaluated within the selected organizational context.

Once an organization is created, the system automatically generates a Role–Process Matrix and a Role–Competency Matrix for that organization, initialized with default values derived from Könemann et al.'s [KWA+22] competency framework. These matrices establish competency expectations before any organization-specific modifications.

For further customization, administrators can navigate to the Role–Process Matrix module within the Administrative Management Module to configure role involvement levels for the newly created organization in collaboration with organizational representatives. Any changes to the Role–Process Matrix will automatically trigger a recalculation of the Role–Competency Matrix specific to that organization, ensuring that competency assessments remain aligned with the organization's roles and associated processes.

Figure 6-2 illustrates the organizational management workflow, showing how administrators create organizations, assign unique organization keys, and configure role-process mappings to tailor competency assessments for different organizational contexts. It also depicts how survey users utilize these keys shared by the admin to take the survey as part of their respective organizations.

6.2.1.5 Role–Process Matrix Configuration

The Role–Process Matrix configuration is implemented through a dedicated Vue UI component that enables administrators to manage process involvement levels for each roles in the system. This configuration process follows a structured workflow:

1. **Default Values:**
   By default, the Role–Process Matrix for individual users is prefilled with default values derived from Könemann et al.'s research [KWA+22]. Similarly, newly created organization's Role-Process matrices are prefilled with these default values.

2. **Organization Selection:**
   The administrator first selects the target organization from a drop-down menu, for which the role-process mapping must be configured. This dropdown includes the the default "Individual" option that gives the default individual mappings and names of any previously created organizations that show us the mapping of these specific organizations. After selecting the context in which role mappings should be modified (individuals or organization), all changes made to the role-process mapping will apply exclusively to the selected individual or organization, ensuring that configurations remain specific to the chosen entity.

3. **Role Selection:**
   Once an organization is selected, the administrator chooses a role from one of the

14 predefined role clusters. This selection determines which role's process involvement levels will be configured.

4. **Adjusting Involvement Levels:**
   Once the role is selected, the UI lists all ISO processes in the systems and the role's level of involvement for the corresponding processes. TheseThe following numbers represent this involvement levels:
   • 0 – Not Relevant
   • 1 - Supporting
   • 2 - Responsible
   • 3 - Designing

   The role-process involvements will use the default values from Könemann et al.'s research [KWA+22]. Administrators can modify them in collaboration with organizational representatives to better reflect the real-process mappings of their respective organizations.

5. **Saving Changes:**
   Upon saving by clicking the *SAVE* button, the updated role involvement levels are sent to the backend, where the Flask API updates the corresponding role_process_matrix table in the PostgreSQL database. Additionally, an automatic trigger initiates the recomputation of the Role–Competency Matrix for the selected individual or organization. This ensures that any changes in role-process relationships are consistently reflected in the competency assessment framework.

This implementation ensures that the Role–Process Matrix remains flexible, scalable, and dynamically configurable while aligning with predefined competency expectations.

6.2.1.6 Role–Competency Matrix Computation

The Role–Competency Matrix is dynamically computed based on the Role–Process Matrix (which is organization or individual-specific) and the Process–Competency matrix (which is static across the system and is same for individuals and organizations). Role–Competency Matrix determines the required competencies for each role and is automatically updated whenever modifications occur in its dependent matrices.

Automatic Recalculation Mechanism

1. **Trigger Conditions:**
   • If the Process–Competency Matrix is updated, the recalculation applies system-wide since this matrix remains static across all organizations and individuals.

• If the Role–Process Matrix is modified, the recalculation affects only the corresponding Role-Competency Matrix based on the context where the update occurred (individuals or organizations).

2. **Computation Process:**
   • The system performs a simple multiplication between the Role–Process Matrix and the Process–Competency Matrix and takes the maximum value for each role-competency pairs to derive the Role-Competency Matrix, which gives us competency requirements for individual roles.
   • The computed values indicate the competency levels as per Könemann et al.'s framework [KWA+22] that uses Bloom's Taxonomy[Blo56]:
   • 0: Not relevant
   • 1: Knowing
   • 2: Understanding
   • 3/4: Applying
   • 6: Mastering

   The computation follows the formula:

$$\textbf{Role\_Competency\_Value} = \max\left(\textbf{Role\_Process\_Value} \times \textbf{Process\_Competency\_Value}\right)$$

3. **Backend Implementation:**
   • When an administrator modifies the Role–Process Matrix or Process–Competency Matrix via the Vue UI, the frontend triggers an API call to the Flask backend.
   • The Flask backend receives the update request and invokes a stored procedure in the PostgreSQL database.
   • The stored procedure updates the corresponding Role-Competency matrix. It is executed with the organization ID as a parameter, ensuring that only the relevant organization's Role–Competency Matrix is updated.
   • The database updates the table representing the Role-Competency matrix data model, reflecting the newly calculated competency values.

This automated update mechanism ensures that competency assessments remain consistent and up to date across individual and organizational contexts. This approach eliminates the need for manual recalculations and possible errors that could arise, thereby reducing administrative effort. It maintains efficiency, ensuring competency evaluations dynamically adapt to administrative modifications while preserving data consistency and assessment accuracy across the system.

6.2.2 Survey Module: SE Competency Assessment

The previous section discussed the administrative management module, which prepares the system for the SE Competency Assessment survey by configuring the necessary data models. This section focuses on the survey module, where users conduct the assessment individually or as part of an organization.

Users can select one of three survey approaches: role-based, task-based, or full assessment. This section explains the system workflow and describes how target competencies are determined based on user selections. The workflow follows the structure outlined in Section 5.3.

Each assessment path is examined separately to clarify the user journey and the calculations performed by the required competency calculator to determine. Finally, the section details how the system presents results and feedback to the user.

6.2.2.1 Organizational Tailoring Implementation

Before starting the survey, the user must specify whether they are taking the assessment as an individual or as part of an organization. If the user selects an individual assessment, the required competency levels are determined using the default role-competency mappings from [KWA+22]. If the user is part of an organization, the system applies organization-specific role–process mappings and competency requirements, ensuring a tailored assessment.

The system supports organization-specific competency assessments by allowing users to take the survey individually or as part of an organization. If an organization-based assessment is selected, users enter a unique organization key provided by the administrator to ensure competency requirements align with the organization's role–process mappings.

The system stores the Role–Competency Matrix entries in a single database table, where each row is mapped to an organization key. This key allows the system to filter and retrieve role-competency mappings specific to an organization or individual users. When a user takes the survey as an individual who is not part of any organization, a default organization key, which is internally assigned to individual default mappings from [KWA+22], is used to retrieve the role-competency mappings for these individuals from the Role-Competency Matrix. In subsequent steps, these values are then used to determine the required competency levels based on the user's further selections in the survey workflow. As depicted in Figure 6-3, when the user takes the survey as part of an organization, they must provide their respective organization key shared by the administrator. The system

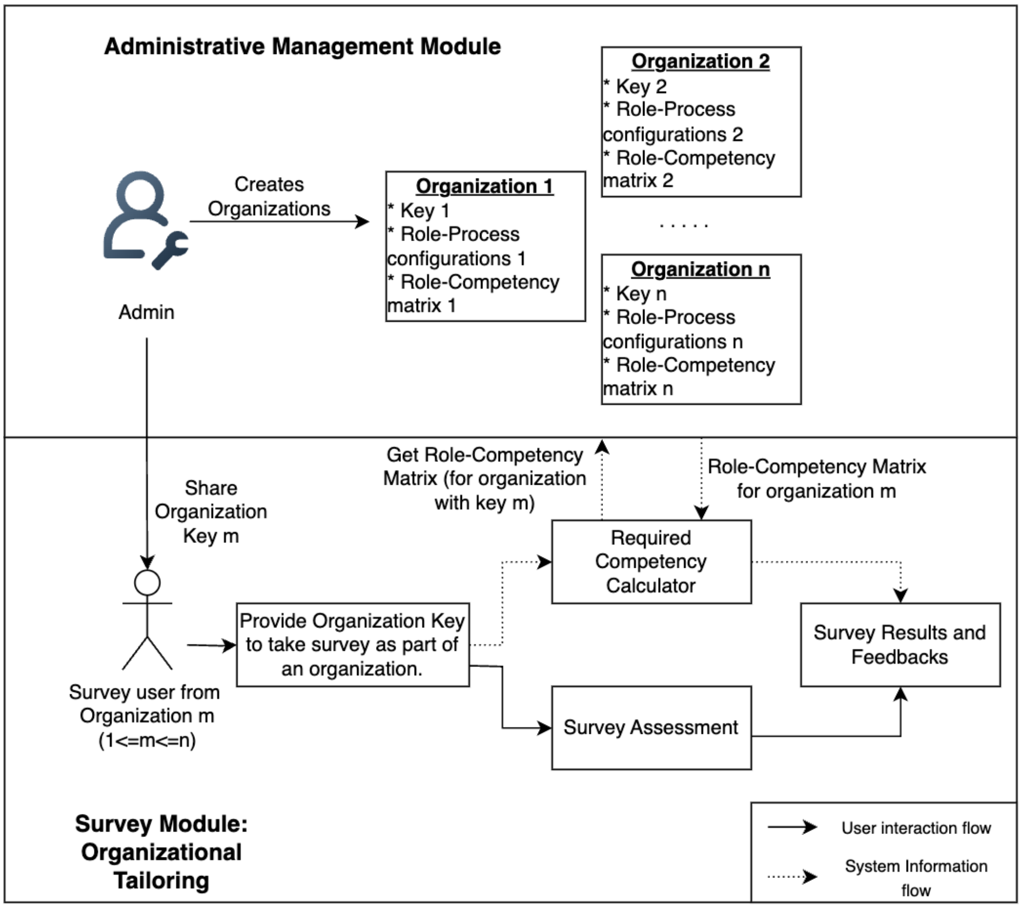


Figure 6-3: Organizational Tailoring: User Perspective

then uses this key to retrieve the corresponding role-competency mappings specific to that organization. These mappings are then used to determine the required competency levels for that user in the organization's context.

This implementation ensures that competency assessments remain scalable for multiple organizations and adaptable based on each organization's competency needs.

6.2.2.2 Assessment for multiple role(s)

Figure 6-4 illustrates the implementation of multiple role assessments in the SE Competency Assessment System. The Workflow ensures that required competencies are dynamically computed based on the user's choices (role(s) performed) and organizational context.

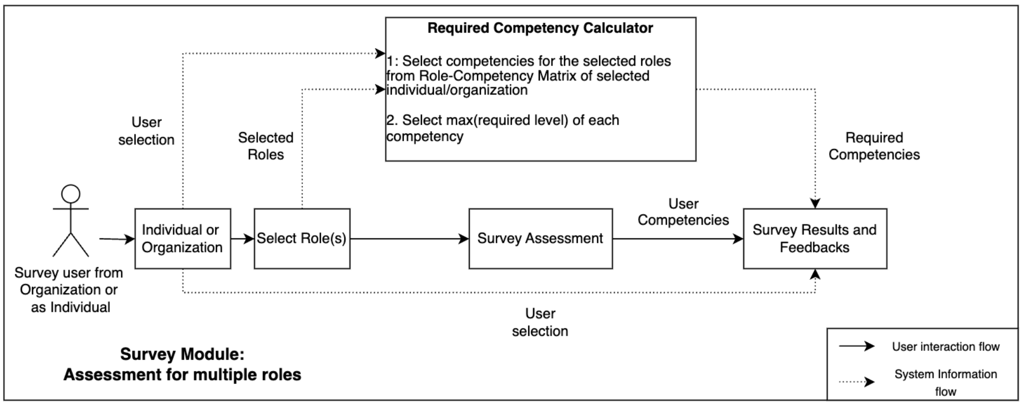


Figure 6-4: Workflow for multiple-role SE competency assessment

Once the user has selected to take the survey in an individual or organizational context, the user selects one or more roles from the 14 predefined role clusters (Subsection 3.2.1). The Required Competency Calculator then uses the selected roles to dynamically determine the competency levels required to perform these roles. The calculation follows the steps outlined in Algorithm 1.

Algorithm 1 Required Competency Calculation

Input: Selected roles, Organization Key

Step 1: Retrieve all required competencies for the selected role(s) from the Role–Competency Matrix. Retrieve role-competency mappings corresponding to the selected organization or individual user.

Step 2: For each competency, select the maximum required level across all selected role(s).

Output: Required competency levels for performing the selected role(s)

The Required Competency Calculator applies the logic explained in Algorithm 1 whenever a user chooses the role-based survey. The user choosing this survey path should then select one or more roles they perform at work. If the user selects only one role, the required competency levels are directly retrieved from the Role–Competency Matrix. However, when multiple roles are chosen, the system ensures that the highest required level across the selected roles is considered for each competency.

After selecting the roles, the user proceeds to the survey assessment, where they evaluate themselves on all competencies. Meanwhile, in the backend, the Required Competency Calculator computes the required competency levels based on the selected roles and organizational context.

Once the user completes the survey, the computed required competencies are sent to the Results and Feedback Module. This module compares the user's recorded competency levels from the survey with the required competency levels determined by the Required Competency Calculator. The results are then used to generate personalized feedback, allowing the user to identify competency gaps and areas for improvement.

This implementation ensures that competency assessments are accurately tailored based on user selections while maintaining flexibility in handling individual and organizational competency requirements. The structured approach guarantees consistency in evaluations, allowing users to receive competency insights aligned with their chosen roles and organizational context.

6.2.2.3 Assessment for unknown role(s)

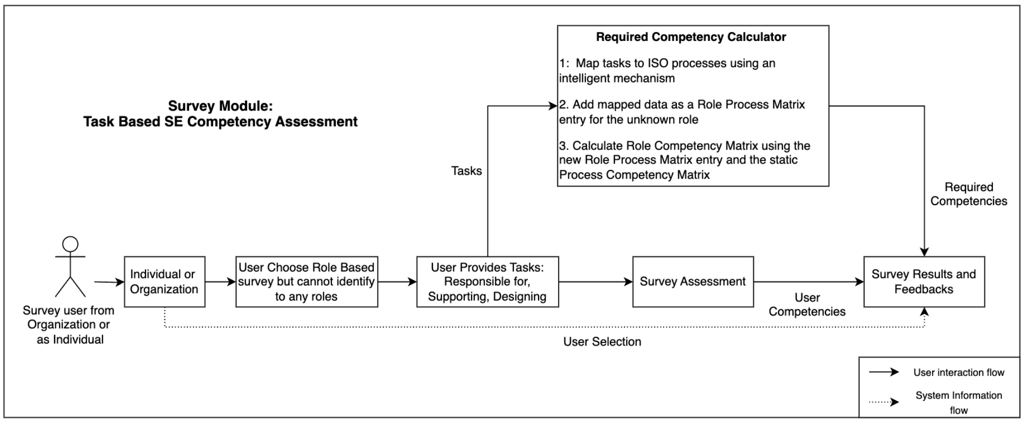


Figure 6-5: Workflow for Task-Based SE Competency Assessment

Figure 6-5 illustrates the implementation of the Task-Based SE Competency Assessment, which is used when a survey user opts for a role-based survey but cannot identify with any of the 14 predefined role clusters. This scenario commonly arises when organizations have different internal naming conventions for the same roles or when entirely new role structures exist within a company [DAR+21, INC23b]. The system provides a mechanism for these users to define their job responsibilities in natural language text and derive competency requirements based on this input.

As explained in the previous case, the user first selects whether to take the survey as an individual or as part of an organization. This selection is passed to the Results and Feedback Module to ensure the user's competency requirement is derived from the correct organizational context and that the user's survey results are grouped correctly under the selected organization or stored as an individual assessment. If the user opts to perform a role-based survey assessment and cannot identify their role in the predefined role clusters, they can select the option *"Cannot identify to any role."* This takes the user to a new interface where they must input their job tasks in natural language, categorized into three involvement levels:

•*Supporting*: User is assisting in the process.

•*Responsible*: User is responsible for performing the process.

•*Designing*: User is designing and defining the process

These levels directly map to the levels in Role–Process Matrix (3.2.3), where 1 represents supporting the process, 2 represents being responsible for the process, and 3 represents designing the process.

Once the user submits their tasks, the front-end UI sends the data to the backend, where an intelligent mapping mechanism is employed to process the user's natural language input. This mechanism uses a Retrieval-Augmented Generation (RAG) model [LPP$^+$21], which then identifies the ISO processes performed by the user based on the described tasks. The RAG model uses a custom knowledge base of embedded[MCC$^+$13, Ope24b] ISO Process documentation [Int23b] and retrieval logic[DGD$^+$25] to ensure that user-defined tasks are accurately mapped to the relevant ISO processes.

After identifying the ISO processes performed by the user, the system inserts an entry in the Role–Process Matrix, creating a customized role-process mapping for the unknown role that this user performs. This entry is then multiplied with the static Process–Competency Matrix followed by a selection of maximum competency level from each competency to derive the Role–Competency Matrix for the user. The calculation follows the steps outlined in Algorithm 2.

Once the processes performed are identified, the user then takes the survey, where the user is assessed on all competencies. Meanwhile, in the back end, the Required Competency Calculator further calculates the required competency levels based on the new entry in the Role-Competency Matrix for this unknown role.

The RAG model implementation is developed using the LangChain framework [Lan]. This framework facilitates seamless interactions with pre-trained large-language models to generate structured outputs[Lan24]. These structured outputs are essential in web application development since front-end and backend communication is mostly facilitated via structured JSON files. The implementation details of the LangChain-based RAG application are explained in the next section.

Algorithm 2 Task-Based Required Competency Calculation

Input: User-defined tasks in natural language text categorized into Supporting, Responsible, and Designing levels
Step 1: Map tasks to ISO processes using an intelligent mapping mechanism.
Step 2: Store the mapped data as an entry in the Role–Process Matrix for the unknown role.
Step 3: Compute the Role–Competency Matrix by multiplying the new Role–Process Matrix entry with the static Process–Competency Matrix and selecting each competency's maximum level.
Output: Required competency levels for performing the described tasks

Once the required competencies are determined, the results from the Required Competency Calculator and the user's survey responses are passed to the Results and Feedback Module. This module compares the user's recorded competency levels from the survey with the required competency levels determined by the Required Competency Calculator module. The assessment's final results and personalized feedback are then generated and provided to the user.

By structuring the Task-Based SE Competency Assessment, the system ensures that users who do not fit into predefined role clusters still receive accurate competency assessments tailored to their job functions. The automatic mapping and computation processes eliminate the need for new manual role definitions while maintaining the integrity of the competency evaluation framework.

6.2.2.4 Assessment for the unknown role(s) continued: Retrieval Augmented Generation (RAG) mechanism for mapping task to ISO process

This section explains how the RAG framework 3.5 [LPP+21] is implemented to solve role-process mapping problem in our SE Competency Assessment System. The first step explains the preparation of the knowledge base for the RAG framework. The RAG mechanism retrieves relevant information from this knowledge base in order to identify the ISO processes performed by the user. The knowledge base comprises relevant information from the ISO Processes documentation [Int23b]. The second step explains how user-inputted tasks are preprocessed and validated before being processed in the RAG pipeline. The third step explains how we use the validated user inputs to retrieve relevant chunks of ISO documents from the knowledge base prepared in Step 1 and then use a large language model to answer questions based on the user's validated inputs and retrieved documents.

That is, based on the user-inputted tasks and relevant ISO document chunks retrieved; a large language model identifies the ISO Processes the user is *responsible for*, *supporting* and *designing*. All these three steps are explained in detail in the following subsections.

Step 1: Preparation of the knowledge source RAG Framework

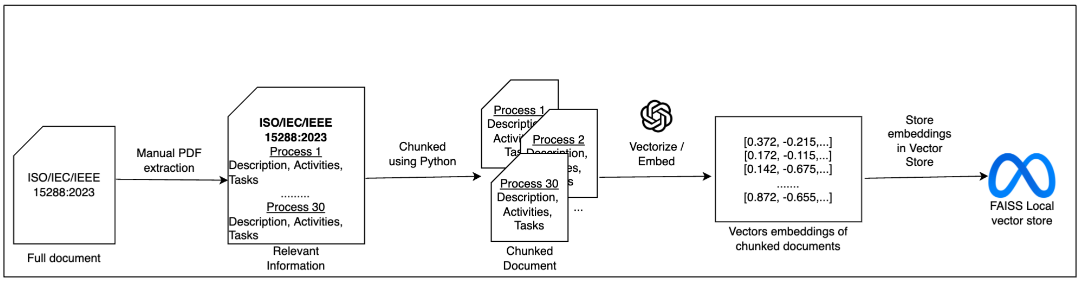


Figure 6-6: Preprocessing and Vector Storage for the RAG-based Task-to-ISO Process Mapping

This subsection describes the implementation of the indexing and building of the knowledge base step in RAG, which is explained in the foundation section of this thesis 3.5. The first step in this process involves preprocessing the ISO/IEC/IEEE 15288:2023 document [Int23b] to extract only the relevant content required for task-to-process mapping. The document contains various sections that are not necessary for this application, so a manual extraction was performed to retain only the 30 ISO processes, along with their descriptions, associated activities, and tasks under those activities. Since this was a one-time preprocessing task, manually extracting the relevant sections was deemed more efficient than implementing an automated PDF parsing solution, which would be unnecessarily complex for a fixed and limited dataset.

Once the relevant content is extracted, the document is chunked using Python. Chunking refers to splitting the document into separate independent parts[LPP+21]. Each chunk consists of a single ISO process, description, activities, and associated tasks. This chunking strategy ensures no overlap or information leakage while keeping all the relevant data for a single process together in a structured format. Since our primary goal is to identify the most relevant process, segmenting the document at the process level is the most effective approach.

After chunking, the next step involves vectorizing the text. This is done using OpenAI's *text-embedding-ada-002 model*[Ope24b]. Vector embeddings convert textual data into high-dimensional numerical representations that capture semantic relationships[MCC+13].

This enables the system to compare user-inputted tasks with preprocessed ISO process descriptions by computing similarity scores between the embeddings[SM83].

The computed vector embeddings of the chunked document are then indexed and stored locally using a vector database provided by the FAISS (Facebook AI Similarity Search) library[DGD+25]. FAISS library is optimized for fast and efficient similarity searches on large-scale datasets. Since implementing a dedicated vector database is beyond the primary scope of this master's thesis, storing embeddings locally with FAISS is sufficient for the current implementation. FAISS allows fast retrieval without requiring an external database, ensuring a lightweight yet efficient approach to handling vectorized data[DGD+25].

Figure 6-6 illustrates the preprocessing pipeline for the RAG-based task-to-ISO process mapping. The ISO/IEC/IEEE 15288:2023 document is manually extracted, chunked, and vectorized before being stored in FAISS, ensuring efficient retrieval during the task-based competency assessment process. This vectorized representation of processes will later be used in semantic similarity comparisons to identify the closest ISO processes corresponding to the user's input tasks.

Step 2: User Input Pre-Processing

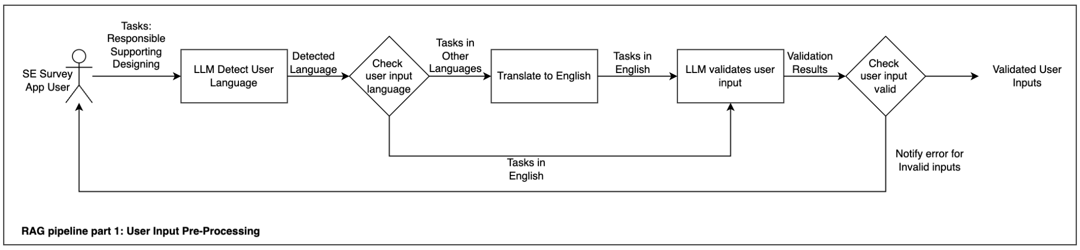


Figure 6-7: Pre-processing pipeline for user task inputs in the RAG system

Figure 6-7 illustrates the pre-processing phase of the Retrieval-Augmented Generation (RAG) pipeline. This step ensures that user-provided tasks and responsibilities are correctly formatted, validated, and prepared before being mapped to ISO processes. Pre-processing is essential for standardizing user input, improving retrieval accuracy, and aligning responses with the competency framework.

When users select role-based assessment but cannot identify with predefined roles, they are prompted to enter their work tasks and responsibilities as natural language texts. This input is then sent to the flask backend, where it is processed using a LangChain-based

LLM workflow. The pre-processing step consists of three core operations: language detection, translation (if needed), and input validation. Each step is executed by a dedicated LLM prompt, ensuring a structured and controlled processing workflow.

User input first passes through the language detection step, where an LLM determines if the input is in English. Since retrieval models perform better when queries and stored embeddings share a common language[Lin06], non-English inputs are automatically translated into English. This translation helps to improve the semantic matching in the FAISS vector store and ensures better retrieval accuracy. The language detection prompt asks the LLM to check whether the input is in English or another language. If the detected language is not English, a separate translation step is triggered, where an LLM is prompted to translate the tasks into English.

The same LLM model is used for language detection and translation but with different parameters. For translation, the temperature is set to 0. This restricts the LLM's creativity and, thus, ensures an accurate one-to-one translation[HBD+20]. This guarantees a direct and accurate translation without unnecessary variations.

Once translated, the input undergoes validation, where another LLM checks if the user-inputted tasks are meaningful and relevant to systems engineering. Since LLMs generate responses for any input, filtering out irrelevant or nonsensical entries is necessary[AG24]. The validation prompt instructs the LLM to determine whether the provided tasks are structured correctly and belong to the systems engineering domain. If the inputs are invalid, the system notifies the user, prompting them to refine their response. If the input is valid, the pre-processing phase is completed, and the validated task descriptions are ready for the next steps in the RAG pipeline.

The LangChain framework enables structured interaction with Large Language Models (LLMs) by allowing the system to enforce predefined response formats. This ensures that the LLM outputs structured categorical values rather than free-text responses [Lan24], making it easier for the backend to process and execute conditional logic dynamically. The system can efficiently determine subsequent actions by instructing the LLM to return specific expected outputs for certain inferences. For example, suppose the system needs to verify whether a user-provided task description is in English. In that case, the LLM can be instructed to return only two possible values: *English* or *Not English*. The backend can then implement logic to handle each case accordingly. The workflow is implemented as follows:

- If the detected language is not English (LLM outputs ”Not English”), the system invokes the translation function. If the detected Language is English (LLM output ”English”), then no translation is needed, and proceed for task validation.
- If the tasks are invalid (LLM outputs ”invalid”), the user is notified to re-enter the tasks.
- If the tasks are valid, the pipeline moves to retrieval and process identification.

This approach enhances predictability, automation, and system reliability in LLM-based application development. It helps to minimize the ambiguity problem in LLM-generated responses. This modular approach ensures robust handling of edge cases, including invalid inputs and non-English text, thereby minimizing errors and enhancing user experience.

After pre-processing, the validated user inputs are passed to the retrieval and reasoning pipeline, where ISO process identification is performed using vector search and reasoning LLMs.

The following algorithm outlines the pre-processing workflow:

Algorithm 3 User Input Pre-Processing Pipeline

Input: User-provided tasks (Responsible, Supporting, Designing)
Output: Validated tasks in English
Initialize LangChain LLMs for language detection, translation, and validation
Step 1: Language Detection
Call **Language Detection LLM** to check if input is in English
if input is not in English **then**
    Call **Translation LLM** (temperature = 0) to translate input to English
else
    Proceed to Input-Validation step in the Pre-Processing Pipeline
end if
Step 2: Input Validation
Call **Validation LLM** to verify task relevance
if input is invalid **then**
        Return error message to user
else
        Proceed to next steps in the RAG pipeline
end if

Step 3: Enhanced ISO Process Identification via Pre-RAG Reasoning

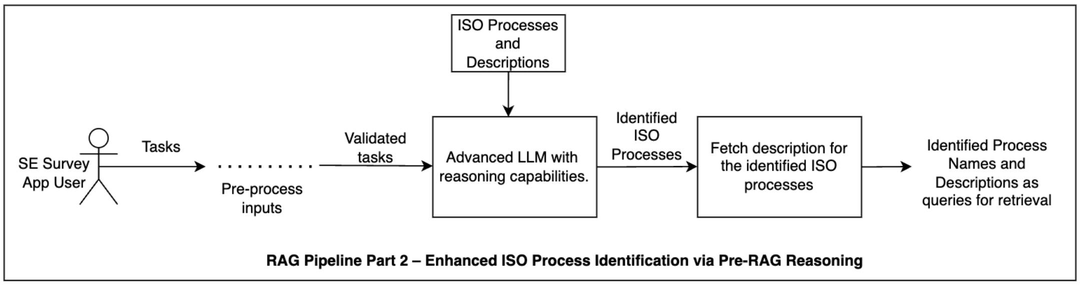


Figure 6-8: Enhanced ISO Process Identification via Pre-RAG Reasoning

Figure 6-8 illustrates the ISO process identification step, which acts as a pre-RAG reasoning mechanism to improve retrieval effectiveness. This step was introduced after workshop discussions with domain experts revealed that even experienced professionals find it challenging to accurately map job descriptions to ISO processes without extensive reasoning and context analysis (cite workshop). Given these insights, an LLM was used before the retrieval step to identify the processes to retrieve before the actual retrieval step. Another reason to include this step was to introduce a bias into the Process Identification pipeline by giving a little extra consideration to certain ISO Processes, which was because it is known that the app users would be mainly performing these processes from domain knowledge.

Instead of directly retrieving documents from the vector store using raw user input, we provide a creative large language model (such as GPT-4o with high temperature) with all the 30 ISO Process names and their short definitions and then ask the model first to try to deduce which tasks the user is performing. While preparing the prompt, we also induce biases into the system by including the ISO processes that need to be weighted more while the model responds. The model then outputs the identified ISO processes as structured data, which is used as an intermediate step before retrieval. This approach significantly increases the accuracy of subsequent retrieval processes and, thereby, the final task-process mappings compared to a basic RAG implementation.

Once the ISO processes are identified, their corresponding descriptions are extracted from an internal database. These descriptions and their respective ISO process names are then used in the next step of the pipeline, where the actual retrieval operation is performed using this data by employing a vector search mechanism [DGD+25].

This pre-RAG reasoning step was introduced because initial experiments with direct retrieval-based RAG showed poor performance in identifying relevant ISO processes. The basic RAG setup struggled to classify tasks effectively, resulting in low success rates for matching user tasks with ISO processes. This approach helped to:

- Reduce token usage by narrowing down the search space before retrieval.
- Improve retrieval accuracy by ensuring queries are more contextually aligned with user input.

This enhancement is observed to significantly improve the effectiveness of competency assessments significantly, ensuring that user tasks are mapped to the most relevant ISO processes before proceeding to final retrieval and reasoning in the next step.

Step 4: ISO Process Retrieval and User Task Classification Using Retrieval Augmented Generation (RAG)

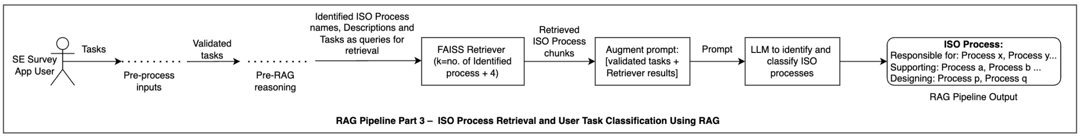


Figure 6-9: Final Step of RAG Pipeline: ISO Process Retrieval and User Task Classification

Figure 6-9 illustrates the final stage of the Retrieval-Augmented Generation (RAG) pipeline, where the identified ISO processes from the pre-reasoning step are used to retrieve relevant ISO Process chunks from the vector database that contains information about the description, activities and tasks within each process. The system employs FAISS (Facebook AI Similarity Search), an efficient vector search library, to retrieve semantically similar documents from the pre-stored ISO process embeddings.

To ensure no relevant processes are missed in the retrieval process, the retriever fetches $k+4$ documents from the FAISS vector store, where k is the number of ISO processes identified in the pre-reasoning step. The additional four retrieved processes serve as a buffer to capture potentially related processes that the pre-reasoning LLM might have missed. The value of k is adjustable to optimize retrieval efficiency.

Once the relevant ISO processes are retrieved, they are combined with the validated user tasks to form an augmented prompt. The LLM is then queried with this prompt, asking it to identify the ISO processes performed by the user based on the retrieved ISO processes chunks and categorize them into three categories of user involvement:

- **Responsible for** – Processes the user directly executes or leads.
- **Supporting** – Processes where the user assists.
- **Designing** – Processes where the user defines or improves workflows.

The LLM outputs a structured output, returning three separate lists of ISO Processes the user is performing categorized to these three involvement levels. This ensures a clear mapping of the user's provided tasks to relevant ISO processes.

This final output is used as input to the competency assessment framework, where the identified responsible, supporting, and designing processes are referenced against the Process–Competency Matrix to determine the required competencies for the user's role. The competency computation follows the Algorithm 2 which dynamically derives the necessary skill levels based on the identified processes.

6.2.2.5 Full Comprehensive SE Competency Assessment

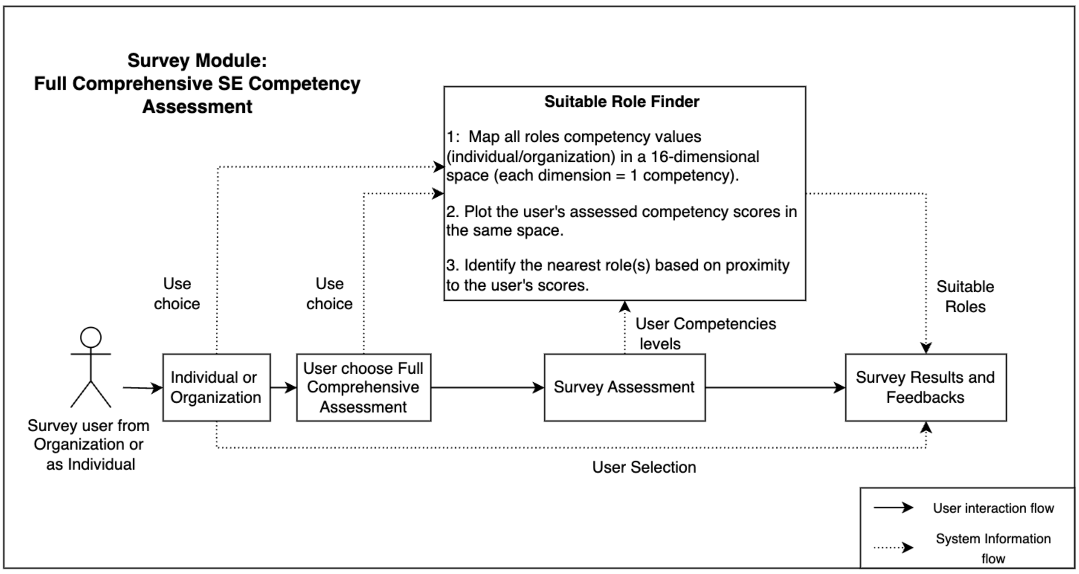


Figure 6-10: Workflow for Full Comprehensive SE Competency Assessment

Figure 6-10 illustrates the Full Comprehensive SE Competency Assessment workflow. Unlike role-based assessments, where competency gap analysis is performed, this assessment evaluates the user across all competencies in the system. The system then determines the role(s) that best fits the user based on the assessment results.

The process begins with the user selecting whether they are taking the survey as an *individual* or as part of an *organization*. This selection is passed to the *Suitable Role Finder* module, which then uses the appropriate Role–Competency Matrix for this user based on the selected organization to determine the best-fit role(s). That is:

- If the user is assessed as an *individual*, the system utilizes the default Role–Competency Matrix from Könemann et al.[KWA+22].
- If the user is assessed as part of an *organization*, the system retrieves the organization-specific Role–Competency Matrix configured in the administrative module (Refer Section 6.2.1.4).

The user then proceeds to complete the self-assessment survey. The users' recorded SE competency assessment scores are then analyzed in a 16-dimensional space, where each dimension represents one competency in the system (there are 16 competencies in the system at the moment). The system then identifies the most suitable role(s) by computing the proximity between the user's competency vector and predefined role competency vectors (from the selected Role-Competency Matrix).

Algorithm 4 Suitable Role Identification Algorithm

Input: User competency scores, Role–Competency Matrix
Step 1: Map all roles' competency values (individual/organization based on user selection) into a 16-dimensional space.
Step 2: Plot the user's assessed competency scores in the same space.
Step 3: Compute proximity to each predefined role using a distance metric.
Step 4: Identify the nearest role(s) based on the computed distances.
Step 5: Perform Competency Gap Analysis of user-score vs the role-competency mapping for the nearest role.
Output: Most suitable role(s) for the user and Competency Gap Analysis.

The system supports two distance metrics for nearest role calculation:

- **Euclidean Distance:** Measures the direct spatial distance, favouring balanced competency distribution [DD17].

- **Manhattan Distance:** Measures absolute differences, ensuring competency gaps are accounted for separately [DD17].

The choice of distance metric influences how the system determines the most suitable role for a user based on their competency profile. Euclidean distance prioritizes candidates with evenly distributed competencies, making it useful when an organization seeks well-rounded individuals with balanced skill sets. Manhattan distance, on the other hand, treats each competency separately, making it more effective when evaluating specialists who may excel in some areas while having gaps in others. For example, if a role requires proficiency across multiple competencies without major weaknesses, Euclidean distance is preferable since it penalizes significant variations. Suppose the role prioritizes expertise in just a few key competencies; Manhattan distance is better suited in that case, as it allows for strong specialization without being overly affected by missing skills in unrelated areas. The system allows switching between these metrics, enabling organizations to tailor the role-matching process based on their specific requirements.

Once the most suitable roles are identified, they are sent to the Results and Feedback Module, where personalized recommendations are provided, along with competency gap analysis relative to the best/fit role(s) identified. The workflow of this survey path is explained in Algorithm 4. This approach allows the assessment system to go beyond competency gap analysis and offer data-driven career guidance, ensuring users receive recommendations aligned with their strengths.

6.2.2.6 Survey Assessment

Implementation of Assessment Questionnaires

The SE Competency Assessment System survey assessment is designed to evaluate users' competencies based on a structured, science-based approach discussed in the design section of survey questionnaires (Section 5.4). The questionnaires follow the competency levels defined by KÖNEMANN et al.[KWA+22], which are aligned with Bloom's Taxonomy [Blo56] to ensure a hierarchical classification of competencies. Each of the 16 competencies is assessed by a single question designed to measure the user's proficiency level in that competency.

The Role–Competency Matrix, as previously explained (Section 3, [KWA+22]), categorizes competency levels as follows:

- **0 – Not Needed**: Competency is not required for the role.

- **1 – Knowing**: Basic awareness of concepts required.
- **2 – Understanding**: Ability to grasp and explain fundamental concepts.
- **3/4 – Applying**: Ability to actively use knowledge in tasks.
- **6 – Mastering**: Expert-level knowledge and ability to guide others.

For each competency in the system, KÖNEMANN et al.[KWA+22] have defined the definition for the five possible competency levels (Not Relevant, Knowing, Understanding, Applying and Mastering). These definitions are directly mapped to the Role-Competency Matrix. For example, in Figure 6-11, we see how *Systems Thinking* Competency definitions by KÖNEMANN et al.[KWA+22] are used to create the survey questionnaire to assess the competency levels for this competency. The questions are directly derived by reframing the definition into a second-person perspective statement, making it more direct and personal. This enables self-assessment using the questionnaires. Since the questions are directly derived from established definitions, the approach's validity is ensured.

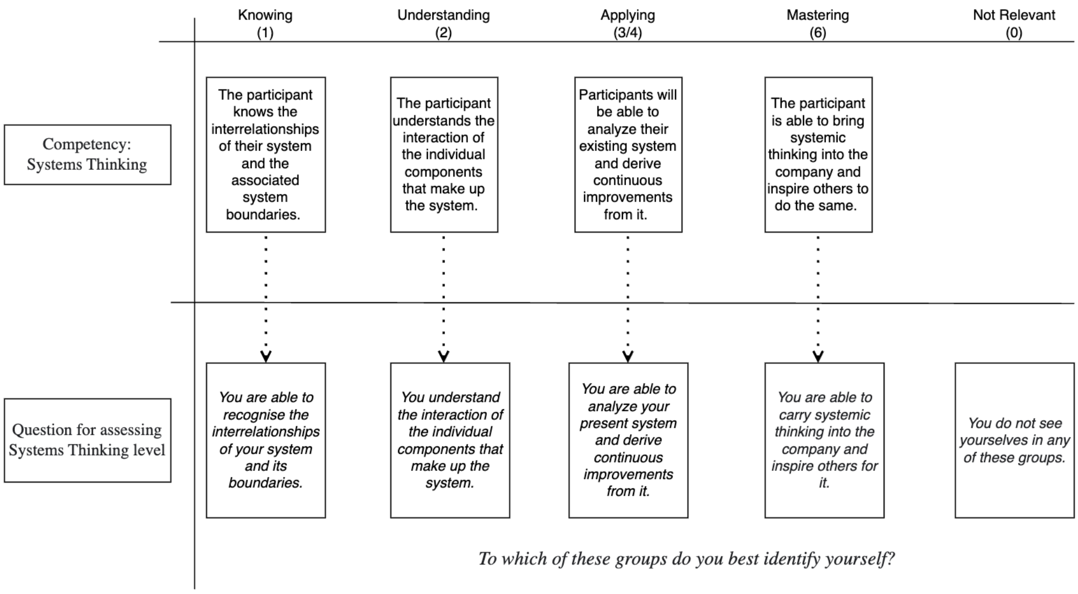


Figure 6-11: Formation survey questionnaires from KÖNEMANN et al.[KWA+22] competency level definitions

These definitions guided the formulation of the survey questions. Rather than using Likert scales[Lik32], the assessment follows a structured response format similar to the Guttman scale[Eng05], ensuring competency levels are progressively arranged. Users can select from predefined hierarchical competency groups to the Group that the user thinks they fits best into:

- **Group 1**: Basic recognition of concepts (Level 1).
- **Group 2**: Conceptual understanding (Level 2).
- **Group 3**: Analytical capability to improve processes (Level 3/4).
- **Group 4**: Mastery and organizational influence (Level 6).
- **Group 5**: Does not identify with any levels (Level 0).

If a user selects Group 5, all other selections are invalidated, and their competency level for that competency is recorded as 0. If multiple groups are selected, the highest selected value of the Group is recorded as the user's competency level in that competency. This ensures a streamlined and unbiased assessment while maintaining a structured competency evaluation.

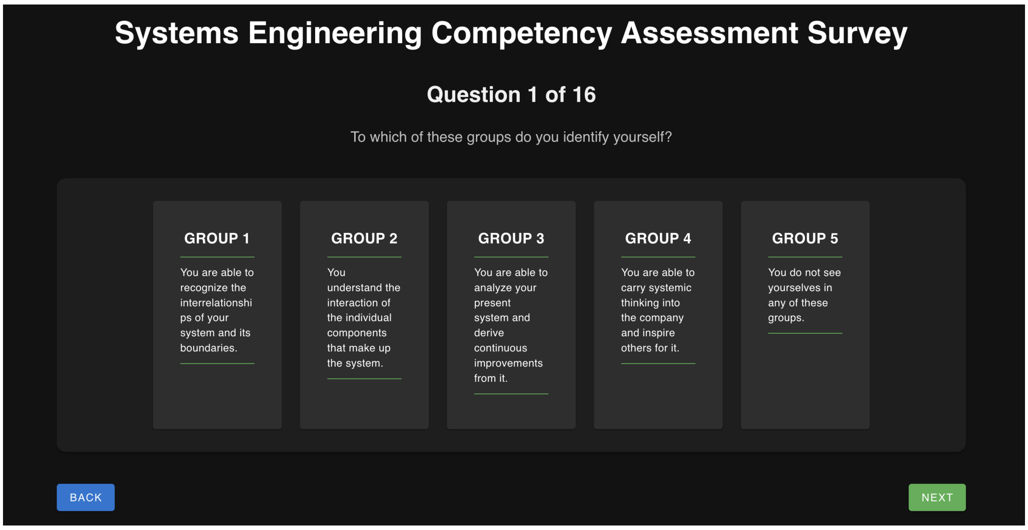


Figure 6-12: Survey Assessment User Interface: Competency Level Selection

Figure 6-12 illustrates the survey assessment interface, where users select their competency levels for one of the 16 SE competencies. The structured selection process ensures that users can intuitively navigate through the UI and perform the assessment.

Implementation of Survey Assessment Module

The survey assessment module is the interface where users engage with the competency questionnaire. Once users choose their assessment path—whether through role(s) based assessment, task-based assessment, or full comprehensive SE competency assessment—they will proceed to the survey interface.

Users must answer 16 questions, one for each competency, selecting their self-assessed competency level from the predefined hierarchical groups. The selections are stored in the front end, allowing users to navigate back and forth to review and adjust responses before submission.

Upon submission, the final responses are sent to the backend for processing. The recorded responses follow the mapping defined by KÖNEMANN et al.[KWA+22]:

- **Group 1** → Level 1
- **Group 2** → Level 2
- **Group 3** → Level 3/4
- **Group 4** → Level 6
- **Group 5** → Level 0

Since the highest selected group determines the competency level, the system stores the highest group level for each competency as the user's assessed levels. The final assessment results are then forwarded to the Results and Feedback Module, which performs gap analysis on the user's required level of competence to the user's recorded level of competence. This includes the visualization of score gaps, personalized feedback, and insights for improvements. The processed data is also passed to other modules for further role identification or competency-based evaluations, depending on the chosen survey path.

Applying this methodology to all 16 competencies ensures scientifically valid competency evaluation, maintaining a balance between simplicity and precision in skill measurement.

6.2.2.7 Survey Results and Feedback Implementation

Once the user submits the survey, their results are stored in the results table in the database. If the user took the survey as part of an organization, the organization key is also recorded, allowing collective retrieval and analysis of all organizational users' results. If the user took the survey as an individual and not part of an organization, then such user results are recorded together with the results of other independent survey takers. This enables competency analysis at both individual and organizational levels.

The system aggregates the user's recorded competencies with the required competency values into a structured dataset containing:

- **Competency Area** – Categorization of the competency.
- **Competency Name** – The specific competency being assessed.
- **User Recorded Level** – The competency level selected by the user.
- **User Recorded Competency Indicator** – The description of the selected level.
- **Required Level** – The expected competency level for the selected role or task.
- **Required Competency Indicator** – The description of the required level.

The front end receives these datasets and renders them into a Radar Chart (Figure 6-13) showing competency gaps between the user-recorded competency level and the required competency level for the user, which the system calculates based on the user's pre-selections before taking the assessments. This visualization is implemented using the Vue-charts package in the Vue front-end development library[Vue14][Vue19]. The radar chart compares user-recorded competencies (displayed in one colour) against required competencies (displayed in another colour). The interactive chart allows users to filter competency areas dynamically to focus on specific skills.

Beyond visualization, the system generates personalized feedback using an LLM-based feedback generator. This is implemented with the help of a LLM. We pass the user's recorded competency level and required competency level details to the LLM along with a definition for each of these levels and ask the LLM to generate detailed insights on strengths, competency gaps, and improvement strategies. The prompt that is used for this use-case is shown below.

```
Prompt:
You are a helpful assistant specializing in providing detailed
   feedback for Systems Engineering Competency Assessments. You
   will receive information about the user's competency levels in
   various areas of systems engineering and the required competency
    levels for each area. Your role is to analyze this information
   and provide constructive, personalized feedback.
```

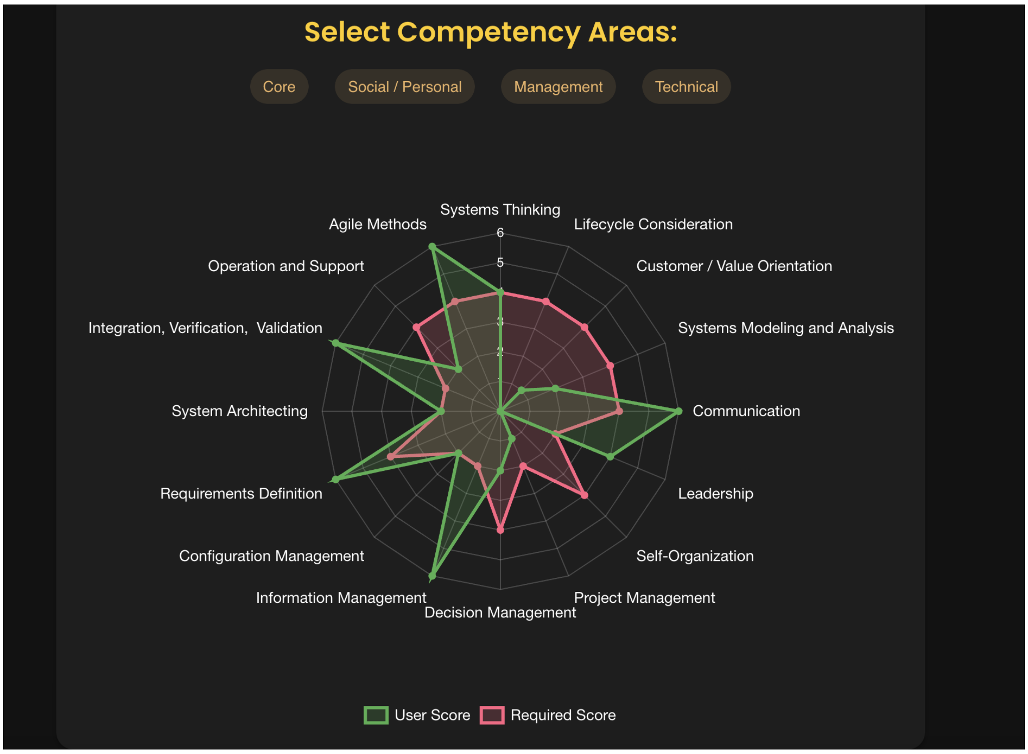


Figure 6-13: Radar Chart Visualization of Survey Results: User Competency vs. Required Competency

```
User Details:
    {
    "competency_area": Category of the competency (e.g., Core,
       Technical, Management)
    "competency_name": Specific competency being assessed (e.g.,
       Systems Thinking, Communication)
    "user_level": Recorded user level from survey assessment
    "user_indicator": Indicator of the user's recorded level based
       on Konemann et al.'s definitions.
    "required_level": Level assigned for this competency by the
       Required Competency Generator logic
    "required_indicator": Indicator of the user's required level
       based on definitions of Konemann et al. levels.
    }

The feedback should:
1. Summarize the user's strengths, highlighting areas where they
   meet or exceed expectations.
```

```
2. Identify gaps where the user's current level does not meet the
   required level.
3. Offer specific, actionable suggestions on improving the user's
   competency in each area.
4. Use a supportive tone encouraging growth, emphasizing practical
   methods to close knowledge gaps,
   such as additional training, hands-on practice, mentorship, or
      specific resources.
5. Do not explicitly specify to the user their competency levels
   like: "knowing", "understanding", etc.
   Instead, use neutral terms like "recorded level" and "required
      level" to maintain objectivity.
```

This feedback generation process runs for all 16 competencies, ensuring a comprehensive feedback report. The generated feedback is displayed alongside the radar chart, providing users with a clear understanding of their competency strengths and areas needing improvement. The system also allows users to export their results as a PDF using Vue.js jsPDF[jsP25], enabling them to retain a structured record for future reference.

Integrating quantitative analysis (radar charts) and qualitative feedback (LLM-generated insights) ensures a scientifically robust and user-friendly competency evaluation process, allowing users to make informed decisions about their professional development.

7 Evaluation and Validation

This chapter presents the evaluation and validation of the Systems Engineering Competency Assessment Web App to ensure its functionality, usability, and accuracy. The evaluation process is structured into multiple parts. Each of these parts addresses a critical aspect of the system's effectiveness. First, we assess the fulfillment of the requirements. This verifies whether all the functionalities defined in this thesis were successfully implemented. It also includes validating that the web application meets its intended purpose and covers all specified requirements. Next, we conduct a usability and user experience evaluation to assess the system's ease of use, design, and visual appeal. In addition, we measure user feedback on performance and functional reliability, ensuring that the system runs smoothly in real-world usage scenarios. Following this, we evaluate the accuracy of the Retrieval-Augmented Generation (RAG)[LPP+21] pipeline(Section 3.5), which identifies ISO 15288 processes[Int23b] performed by the user from user-provided natural language inputs of what they do at work. The goal is to measure how effectively the AI model maps user-described tasks to relevant ISO processes. Finally, we validate the survey questionnaires used in the competency assessment to ensure they are well-structured and aligned with established frameworks. This includes verifying that the competency definitions align with INCOSE standards[sec23] and performing a statistical / survey evaluation of the questionnaire design. This structured evaluation approach ensures that the developed system is functionally complete, user-friendly, reliable, and scientifically grounded.

7.1 Functional Validation: Requirement Fulfillment

Chapter 2.3.1 identifies and lists the core functional objectives the Systems Engineering Competency Assessment Web App should fulfil. This section verifies whether these objectives were successfully implemented.

The web app was designed to provide a structured SE competency assessment platform based on established evaluation techniques. Table 7-1 presents the Requirement Coverage Matrix, mapping each functional requirement to its implementation status and corresponding section reference where the implementation is explained in detail. This structured validation using a Requirement Coverage Matrix confirms that all identified requirements were successfully implemented and that the system met its intended objectives.

Table 7-1: Requirement Coverage Matrix

Requirement	Description	Implemented	Section Ref
Competency assessment for known roles	Users assess competencies for predefined SE roles	Yes	Section 6.2.2.2
Competency assessment for unknown roles	Users assess competencies based on tasks performed	Yes	Section 6.2.2.3
Role recommendation	Suggests best SE role(s) based on competency assessment	Yes	Section 6.2.2.5
Multi-role assessment	Users performing multiple roles are assessed accordingly	Yes	Section 6.2.2.2
Team/organizational competency analysis	Aggregates individual results for competency insights	Yes	Section **??**

7.2 Evaluation of the RAG Pipeline for ISO Process Identification

Large Language Models (LLMs) generate outputs for any input, but this often results in inaccuracies, including hallucinations [XJK24], which is when an LLM makes up invalid responses, claiming them as valid. Therefore, when LLMs are used in production-grade applications, they must be evaluated. This ensures and validates LLMs' reliability in solving domain-specific problems. Es et al. [EJEA+23] introduced *RAGAS*, which is a framework for evaluating Retrieval-Augmented Generation (RAG)[LPP+21] application results using some defined metrics. The *RAGAS Python library* implements the evaluation techniques in [EJEA+23]. This provides an easy-to-use Python-based systematic assessment tool[RAG24]. The next sections explain the steps taken to assess our RAG application using *RAGAS*.

7.2.1 Dataset Preparation

Since no structured dataset could be found that mapped Systems Engineer's tasks and responsibilities to ISO 15288 processes, one was manually created. Job descriptions from

LinkedIn [Lin25] were used to identify the everyday tasks that Systems Engineering roles performed at work. Using *Apify* [Beb25], a web-scraping API, job descriptions were extracted by searching for relevant Systems Engineering job titles such as *Systems Engineer, System Developer, Verification & Validation Engineer, Product Owner, Requirements Engineer, and Quality Engineer*.

After aggregating over 1000 job descriptions, deduplication was performed in Python to remove identical listings. Since manual classification was infeasible, an LLM was used to classify whether each job extracted by the API in-fact belonged to Systems Engineering domain. Each descriptions were also assigned to one of the 14 predefined role clusters3.2.1 by the LLM. This resulted in a filtered dataset of 304 job descriptions.

Analysis revealed that 205 of 304 SE jobs identified were classified under the *Systems Developer* role cluster by the LLM. This resulted in a class imbalance in the dataset since most of the descriptions extracted from LinkedIn belonged to one particular role cluster. To avoid this, a random sampling was done to retain only 51 Systems Developer jobs. This ensured a more balanced distribution across role clusters. Figure 7-1 shows the distribution of Systems Engineering Job Records per Role after balancing.

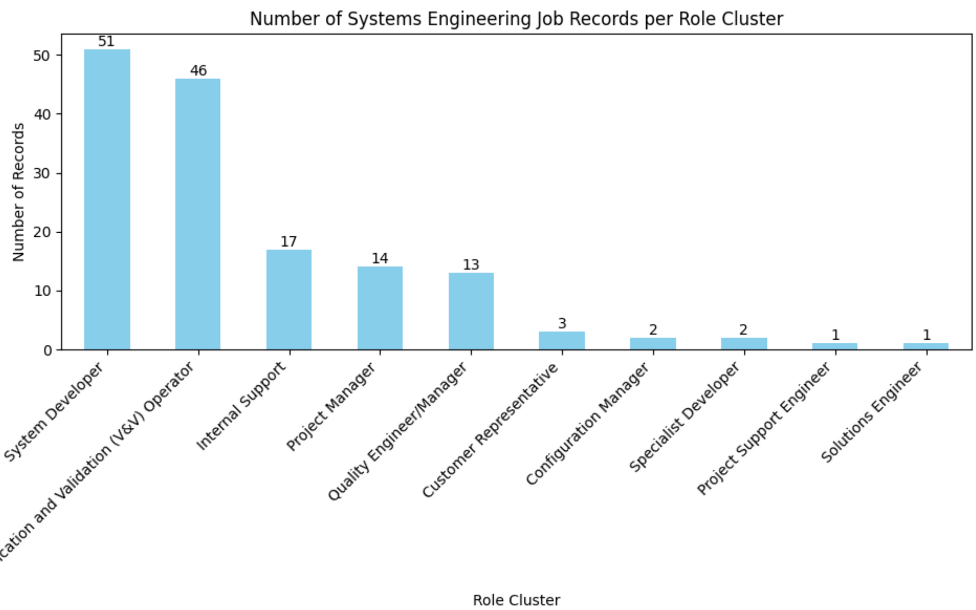


Figure 7-1: Distribution of Systems Engineering Job Records per Role Cluster after Balancing

7.2.2 Extracting User-Performed Tasks from Job Descriptions

For each job description in the final dataset, a Python script that used Langchain [Lan] library was created. This script used an LLM to extract the below details from each job description in the balanced dataset:

- Tasks the candidate is **responsible for**
- Tasks the candidate is **supporting**
- Tasks the candidate is **designing**

The gist of the prompt used for this information extraction is given below:

```
This is a job description from LinkedIn: "{description}"

Read the job description and extract the following information:
- Tasks the candidate is responsible for.
- Tasks the candidate is supporting.
- Tasks the candidate is designing.
```

The output formed the structured input for ISO process identification.

7.2.3 Creating the Gold Standard for Evaluation

A workshop with Systems Engineering experts was conducted to create a ground-truth reference for evaluating the LLM response. Participants were asked to map extracted job tasks manually (from Section 7.2.2) to ISO 15288 processes for 20 randomly sampled job descriptions; the remaining were discarded due to the time availability of experts.

Each participant received:

- **ISO 15288 process definitions**
- **Process-related tasks and activities**
- **Guidelines for task-to-process mapping**

The expert-annotated mappings are used as the expected outputs (gold standard) for evaluating LLM-generated responses.

7.2.4 Evaluation Using RAGAS

The *RAGAS* evaluation metric: *Factual Consistency*[RAG25], which measured *how accurately the LLM identified claims* (in our case, the ISO processes performed by the user) was used for our evaluation use case. The method applies *Natural Language Inference (NLI)* [FWF00] to compare LLM predictions with expert annotations.

Each LLM response was compared to the expert-labeled dataset based on:

- **True Positives (TP)**: Number of correctly predicted ISO processes that the user actually performs.
- **False Positives (FP)**: Number of incorrectly predicted ISO processes.
- **False Negatives (FN)**: Number of relevant ISO processes that the user performs which the LLM did not identify.

The above information is automatically inferred from the Gold Standard Reference and the LLM outputs as claims by the *RAGAS* library, which uses an LLM in the backend for identifying claims and performing inference.

The evaluation results provided by Factual Correctness uses *Precision*,*Recall* and *F1 score* metrics[VR79] [Tha20] to provide the results. These are well-established, commonly used metrics in Machine Learning and retrieval systems.

- *Precision*: Out of all the identified ISO processes, how many ones are correct.
- *Recall*: Measure how many actual ISO Processes the user performs are correctly identified.
- *F1 score*: The harmonic mean of the above two metrics provides a balanced evaluation when there is an uneven distribution between the number of relevant ISO processes and incorrect predictions.

7.2.5 Evaluation Results and Analysis

The *RAGAS Python library* [RAG24] was used to compute and provide these metrics values by comparing LLM predictions against expert-verified mappings. The observed results are provided in the Table 7-2

Findings:

Table 7-2: Performance Metrics for ISO Process Identification

Metric	Score	Interpretation
Precision	0.408	40.8% of predicted ISO processes were correct
Recall	0.266	Only 26.6% of relevant ISO processes were identified
F1-score	0.319	Moderate performance, indicating room for improvement

- **Higher Precision than Recall**: The model avoids incorrect ISO process assignments but misses many valid ISO processes.
- **Low Recall**: The model fails to capture all relevant ISO processes.
- **Low F1 Score**: The balance between precision and recall is low, showing room for improvement.

7.2.6 Justification and Future Improvements

The relatively low scores indicate key challenges:

- **Dataset Limitations**: Experts in the dataset annotation workshops commented that the LinkedIn job descriptions are broad, often failing to match real-world task descriptions.
- **Human Labeling Complexity**: Even domain experts found it difficult to accurately map tasks to processes for the datasets curated from LinkedIn.
- **Task vs. Process Mapping Discrepancies**: Job descriptions list generalized responsibilities, which may not align precisely with ISO process tasks.

Despite numerical limitations, practical observations indicate that the LLM pipeline correctly identified processes in most cases when certain biases were introduced into the prompt based on domain knowledge. This suggests that:

- **Fine-tuning with domain-specific datasets** can improve recall.
- **Enhancing prompt engineering** to incorporate industry-specific ISO process biases may improve accuracy. This is implemented in the current solution (Refer Section6.2.2.3).

- **Using structured competency-based datasets** rather than broad job descriptions can help create more reliable mappings.

While improvements are needed, the task-based approach for ISO process identification shows promise. With further refinements, it can serve as a viable method for mapping job descriptions to structured process frameworks.

7.3 Validation of Survey Questionnaires

To ensure the scientific validity of the Systems Engineering Competency Assessment survey methodology, we check whether the designed questionnaires align with established frameworks and accurately measure the intended competencies it claims to measure(3.2.2).

First, we assess whether the competency statements used in the survey align with recognized standards. This is done by mapping each statement used for assessing the competency level to the equivalent INCOSE competency indicators[INC23b]. If a precise mapping exists, we can infer that the assessment maintains consistency with established frameworks.

Next, we validate whether the translations of competency definitions—derived from KÖNEMANN et al.'s [KWA+22] work—correctly capture the intended competency concepts. This step involves statistical verification to ensure that the translated statements preserve the meaning and intent of the original definitions by KÖNEMANN et al. [KWA+22].

Finally, we conduct a human evaluation to identify pitfalls or problems in survey design. For this, we check if the designed questionnaires have any relevant biases or problems highlighted in [CP05]. This ensures that the questionnaire provides a user-friendly, reliable, and effective assessment of Systems Engineering competencies.

7.3.1 Mapping to INCOSE Definitions

To ensure that the competency level definitions used in our survey align with the *INCOSE Systems Engineering Competency Framework (SECF)*, we systematically mapped the competency level assessment statements used in our assessment questionnaires (Refer Section 6.2.2.6) to SECF indicators. The goal is to validate whether the competency definitions in our system map correspond to established INCOSE competency indicators.

7.3.1.1 Methodology

The mapping was performed as follows:

Algorithm 5 Mapping Competency Definitions to INCOSE Indicators

Input: User-defined competency level indicators
Output: Mapped INCOSE indicators
for each competency c in the system **do**
  for each level l in competency c **do**
    Retrieve the user-defined indicator $I_{user}(c,l)$
    Retrieve all INCOSE indicators $I_{incose}(c)$ for competency c
    Use an LLM to semantically map $I_{user}(c,l)$ to the most relevant $I_{incose}(c)$
    Store the best one or two matches
  end for
end for
Return: Mapped indicators and corresponding INCOSE levels

The mapping was executed using a Large Language Model (LLM). The LLM was instructed to compare the semantic similarity between competency indicators in our system to that in INCOSE's definitions. The following prompt was used:

```
This is a competency indicator defined by the user:
Competency ID: {competency_id}
Competency Name: {competency_name}
User-Defined Indicator: {indicator_en}

Below are the competency indicators from the INCOSE Systems
   Engineering framework:
{List of all indicators for competency: competency_name}

Map the user-defined indicator to the most semantically similar
   INCOSE indicator(s). Return at most two matches. If no
   indicators are sufficiently similar, respond with "none".
```

This process was repeated for all 16 competencies across all four competency levels(3.2.2) in our framework.

7.3.1.2 Results and Analysis

The results indicate that *every competency level statement used in our system mapped to at least one INCOSE indicator*. This confirms that our definitions align with SECF[sec23] at some level. However, to validate whether the mappings also maintain the correct competency level alignment, how our mapped levels correspond to INCOSE's predefined levels was analyzed.

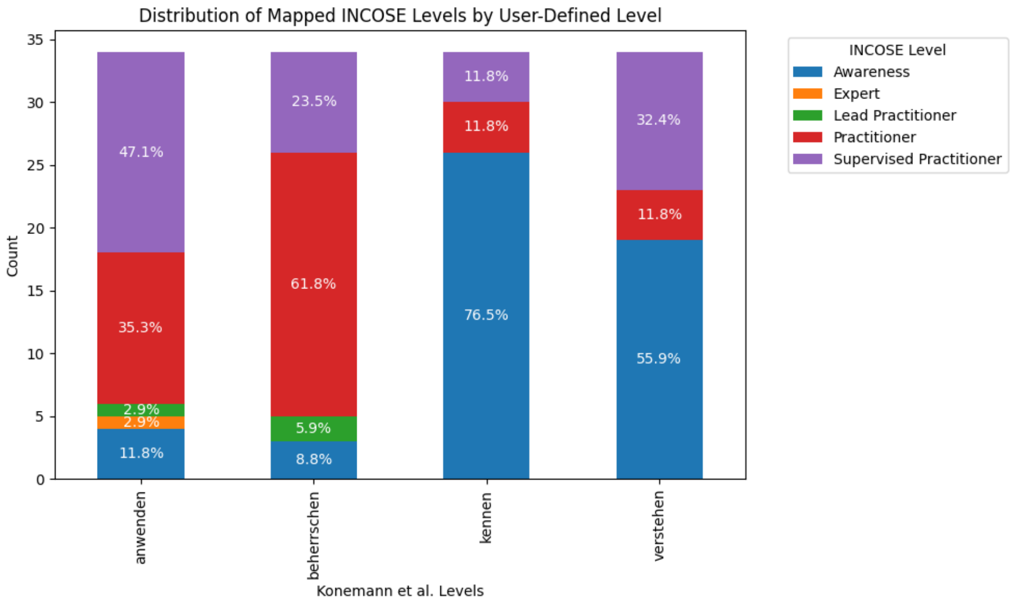


Figure 7-2: Distribution of Mapped INCOSE Levels by User-Defined Levels

Figure 7-2 illustrates how different user-defined competency levels were mapped to INCOSE levels. The results suggest that while mappings exist, there is some variation in level assignments. To assess whether these mappings occurred due to genuine semantic similarity or random chance, a statistical validation using a *T-test* was conducted.

7.3.1.3 Statistical Validation

This statistical validation aims to prove that the LLM mappings did not happen by chance and that there is a significant similarity between our level definitions and INCOSE indicators. To quantify the semantic similarity between our competency indicators and INCOSE's definitions, we computed *cosine similarity scores* [SM83] after embedding the texts using *OpenAI's text-embedding-ada-002 model*[Ope24b]. Higher cosine similarity values indicate more significant semantic similarity between the levels used in the system and the mapped INCOSE's competency indicators.

To validate statistical significance, we:

- Calculated cosine similarity for each correctly mapped competency indicator.
- Randomly sampled an unrelated INCOSE indicator from the same competency category and computed its cosine similarity.
- A statistical test: *T-test*, is to compare correct vs. random mappings.

The *T-test* compares the mean value of similarities between the mapped indicators and the randomly sampled indicator group[Gos08][Fie17]. If the mean values are significantly different, we can infer that the LLM similarity mappings are valid and did not happen by chance. The T-test was conducted using the *SciPy*[VGO$^+$20] stats library in Python.

Results: The below results were recorded for the experiment:

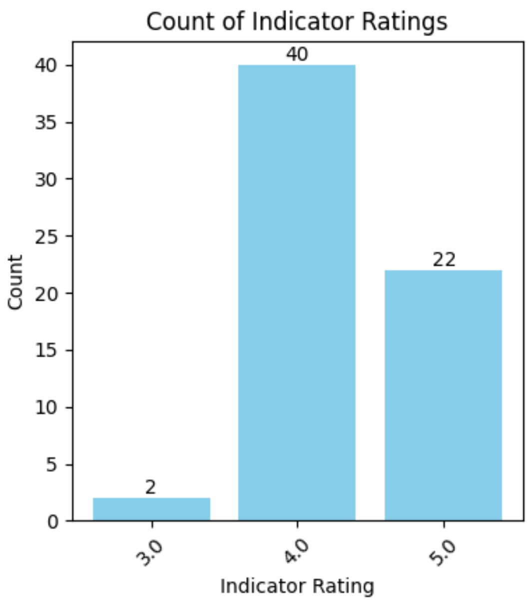
- **T-statistic:** 4.52
- **P-value:** 0.002
- **Average similarity (correct mappings):** 0.78
- **Average similarity (random mappings):** 0.53

The statistical evaluation confirms that the competency statements used in the survey align meaningfully with INCOSE competency definitions and are not random associations which happened by chance. The observed *T-statistic* of 4.52 and a *P-value* of 0.002 is below the standard significance threshold of 0.05[Fis92a]. This indicates that there is only a 0.2% probability that the correct mappings occurred by random chance. Additionally, the average similarity for correct mappings (0.78) was significantly higher than that of randomly assigned mappings (0.53), indicating that the LLM-mapped competency indicators were semantically closer to INCOSE-defined indicators than randomly selected ones. This confirms that aligning the survey's competency levels with INCOSE definitions is scientifically valid and statistically significant by making them grounded in established frameworks.

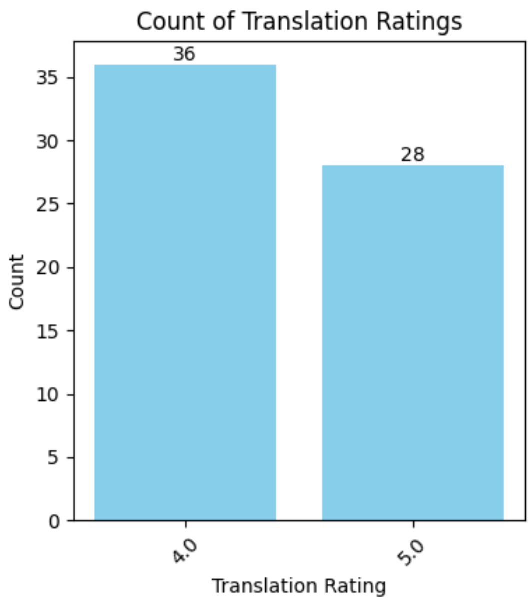
7.3.2 Accuracy of Statement Translation and Interpretation

The questionnaires used for assessing the SE competency assessment were formed by translating (from German) and adapting the competency level definitions by KÖNEMANN et al.[KWA+22](Refer Section 6.2.2.6, Figure 6-11). To evaluate the quality of translated and adapted statements used in the competency assessment questionnaire, we compared them against the original definitions. An LLM-based evaluation approach ensured that the translations preserved their intended meanings. The LLM was provided with the statements used for assessment and the original definitions. Then, it was prompted as below to provide a rating.

- How well the translated indicator captures the meaning of the original definition. (Rate from 1 to 5)
- The quality of translation from German to English. (Rate from 1 to 5)



(a) Distribution of LLM ratings for Indicators in capturing competency meaning



(b) Distribution of LLM-assigned ratings for the translation quality

The results of the LLM ratings for capturing the originality of definitions and translation quality are shown in Figure 7-3a and Figure 7-3b. This indicates that most translations and quality received high scores (4 or 5), suggesting strong alignment with the original definitions.

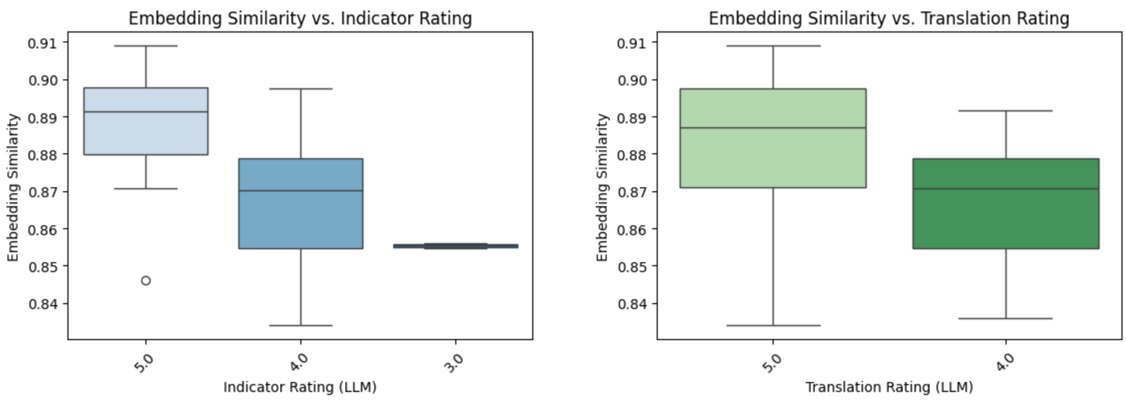
To statistically verify that these ratings were meaningful and not random, a correlation analysis [Pea95] was performed.

7.3.2.1 Correlation Analysis and ANOVA test

Correlation checks whether two variables are connected or if there is a relationship among them[Fie17]. It indicates how strongly one variable reacts to changes in the other[CCW+02]. In our case, the correlation between embedding similarity (a numerical measure of semantic closeness)[Ope24b] and LLM-assigned ratings were studied. In correlation analysis, a correlation value close to 1 means a stronger positive relationship between the variables under consideration, while a value closer to 0 suggests a weaker or no relationship [Fie17].

The correlation results are as follows:

- **0.576** for indicator ratings.
- **0.437** for translation ratings.



(a) Embedding Similarity vs. Indicator Rating

(b) Embedding Similarity vs. Translation Rating

Figure 7-4: Relationship between Embedding Similarity and LLM Ratings.

Figure 7-4 visualizes these relationships. The left figure (7-4a) shows that higher-rated indicators (how well the indicator captures the meaning of the original definition) tend to have higher embedding similarity with their definitions. This is seen by a correlation of 0.576. The figure on the right (7-4b) illustrates that translation ratings also follow a structured similarity pattern. Even though the correlation is relatively weaker at 0.437, these results suggest that the LLM is more likely to assign higher ratings as semantic similarity increases. This reinforces the validity of the assessment process.

Although the correlation values do not indicate a strong relationship (i.e., above 0.7), a moderate correlation is observed. This suggests that LLM-assigned ratings are meaningfully influenced by semantic similarity but are not entirely dependent. Several factors could explain this:

- *LLM Decision Factors Beyond Numerical Similarity:* The LLM does not rely solely on the cosine similarity of embeddings but also considers semantic context, sentence structure, and intent when assigning ratings[VSP+23]. Even if two statements have different wording but the same meaning, the LLM might still assign a high rating.
- *Translation Complexity and Language Differences:* The lower correlation for translation ratings (0.437) may be due to embedding similarity calculations occurring across two different languages (Original German definitions vs. English indicators). Embedding models may not perfectly capture cross-lingual semantic similarity, leading to variations[KPG17].
- *Influence of Prompt Engineering and Rating Criteria:* The prompt used to evaluate the LLM included structured comparison guidelines for rating. The LLM might also weigh these aspects and not only similarity in embeddings.

The observed correlation suggests that LLM ratings are systematically influenced by semantic similarity, but other elements also influence them. While embedding similarity contributes significantly to rating assignments, it is not the sole determinant.

To further check the validity of LLM ratings, an *ANOVA*(Analysis of Variance) test[Fis92b] was performed. This determines whether the mean similarity scores differ significantly for various rating levels. The results showed extremely low p-values:

- 0.00000 for indicator ratings.
- 0.00031 for translation ratings.

In the *ANOVA* test in our context, a low value of p (less than 0.05) means there is a significant difference between mean[Fie17] embedding similarity scores across LLM ratings. This analysis provides further confidence that LLM-generated ratings follow a structured pattern rather than being randomly assigned, strengthening the credibility of LLM ratings.

7.3.3 Usability and User Experience Assessment

The evaluations of the SE competency assessment system conducted so far have mainly relied on LLM-based assessments, which were supported by statistical validation or requirement coverage matrices to ensure that the product aligned with predefined requirements. However, when developing interactive applications that real users will use, accessing the usability and user experience (UX) to enhance user satisfaction and system effectiveness is crucial. Ideally, feedback from this process should be used iteratively to improve the product.

A survey was conducted to evaluate the usability and user experience of the Systems Engineering Competency Assessment App. The survey focused on two key areas:

- Identifying issues in the survey questionnaire used for competency assessment
- Assessing the usability and user experience of the web-based solution

The following subsections explain how these were implemented.

7.3.3.1 Identifying Issues in the Competency Assessment Questionnaire

A well-structured survey should minimize biases that could affect response quality. CHOI et al.[CP05] identified common biases that occur in survey designs. After analysing this study in the context of our application, three biases were selected as particularly relevant to the design of the Systems Engineering Competency Assessment App's questionnaire:

- **Ambiguity in Questions:** Occurs when a question is phrased in a way that allows for multiple interpretations. This can confuse the survey taker and lead to responses that do not accurately reflect the intended meaning of the question[CP05].
- **Complexity of Questions:** Occurs when questions are overly long, structured in a complicated manner, or use technical language that makes comprehension difficult for users[CP05].
- **Double-Barreled Questions:** These questions unintentionally ask for more than one piece of information in a single response. When respondents select an answer, they might agree with one aspect but be forced to accept another unrelated component[CP05].

A survey was created to identify if these biases existed in the existing survey questionnaires used in the SE competency assessment. The survey defined these biases clearly with examples and presented all 16 competency assessment questions and their respective indicators. For each question, respondents were asked:

- **Ambiguity:** Does the question have any ambiguity in meaning? (Yes/No)
- **Complexity:** Is the question overly complex in its structure or wording? (Yes/No)
- **Double-Barreled:** Does any option ask for more than one distinct piece of information in a single answer? (Yes/No)

This assessment was performed with the help of field experts, as a good understanding of Systems Engineering was necessary to accurately assess the survey's clarity and structure. Another reason was that the target audience for competency assessment is also Systems Engineers. Responses have been collected, and preliminary findings indicate the following:

7.3.4 Usability and User Experience Assessment

As part of the same survey, users were invited to test the SE competency assessment web app and provide feedback on its usability and user experience.

To quantify the user-friendliness and overall experience of using the SE competency assessment, the survey participants were asked to provide a score on a scale of 1 to 5 (1 for a low rating and 5 for a high rating) after taking an assessment survey:

- **Usability:** How intuitive and easy the application is to navigate.
- **Design & Visual Appeal:** How aesthetically pleasing and engaging are the layout, colours, and overall interface?
- **Performance & Functionality:** How smoothly the app runs, including responsiveness and absence of bugs.
- Additionally, a free-text feedback section was included to collect qualitative insights and specific user suggestions for improvement.

8 Conclusion

This thesis presented the problem analysis, design, and development of a Systems Engineering Competency Assessment Web Application designed to provide a quick, easy, cost-effective approach to performing Systems Engineering competency analysis based on established standards. The system enables users to assess competencies based on predefined role clusters, task-based assessments, and role recommendations while supporting assessment for independent individuals and individuals in organizational contexts.

The thesis used modern software engineering tools, technologies, frameworks, and established best practices to successfully realize solutions to competency assessment problem in systems engineering. Mathematically grounded techniques were used to calculate the required competencies when a user performs multiple roles and to identify the most suitable role based on their recorded competency assessment results. State-of-the-art GenAi frameworks performed role-process mapping, enabling users to assess their competency based on their work tasks by describing them in natural language. This automated role-process mapping would allow the app to function for organizations where alternative models such as ASPICE models[exp23] are followed. The overall architecture is designed and implemented considering the diversity of organizations in roles, processes and competency requirements. Users could assess their competencies as part of an organization considering the unique organizational competency requirements.

The implementation also prioritized the needs of admins and stakeholders setting up and deploying the app in diverse organizational environments. The admins could easily configure and reconfigure the foundational role-process and process-competency mapping in the context of independent individuals performing the survey or by considering the organization's needs where the product is deployed.

The survey questionnaires for assessing the Systems Engineering Competencies are designed considering widely used psychometric and survey design principles. This questionnaire is validated to ensure their alignment with established competency frameworks. The validation techniques performed with the help of Large Language Models and their strengths in performing natural language tasks and scientifically proving the validity of LLM based assessments open up and highlight new areas and possibilities in using GenAi tools in research and education. User feedback and suggestions after testing the app were systematically collected, recorded and used for future refinements.

A requirements fulfilment analysis of the thesis showed that most of the requirements and fields of action identified were successfully realized. Smart's *Choices of Features*[Sma25] is a commonly used Venn diagram in the software industry to illustrate the relationship between what the user/stakeholders asked for, what the developer built and what the users need. Smart highlights that the real value lies when the software build has features that lie in the intersection of these three areas. Considering this aspect, the features implemented in this thesis all lie in this common overlapping area, which is trying to deliver the most value.

8.1 Key Contributions

The key contributions in the thesis are highlighted below:

- SE competency assessment framework that allows users to evaluate their competencies in alignment with established SE and industrial standards.
- Role-Based and Task-Based competencies assessment addressing the diverse roles and processes in the organization.
- Configurability addressing diverse organizational needs and expected future changes.
- An AI-powered app that solves complex problems leveraging the power of GenAi was introduced. This thesis employed GenAi in role process mapping and providing individualized survey assessment feedback for the users, enhancing the user experience.
- Use mathematical techniques for efficient role identification based on the user's competency profile.
- Validated and grounded survey methodology on established frameworks.
- Explored LLM-based validations reducing human efforts and presented some methods of using LLM in research.

8.2 Evaluation and Findings

The evaluations and validations performed in Chapter 7 confirmed that the system implemented meets its intended objectives:

- Requirement validation demonstrated that all functionalities identified in the Problem Analysis Section (2.1) were successfully implemented.
- Usability testing demonstrated the system's user-friendliness and provided scopes for future improvements.
- GenAi-based role-process mapping revealed moderate accuracy. Areas of future improvement were highlighted.
- Survey questionnaire validation ensured the competency assessment indicators aligned with established standards. These were confirmed using statistical methods.

8.3 Limitations and Future Work

Although the system achieves its objectives, there are potential for improvement:

- **Enhancing the ISO Process Identification Pipeline:** The RAG model that identified ISO processes exhibited moderate recall (0.266), indicating missed processes. Future evaluation must be performed on more relatable dataset to re-validate this. The findings should then drive the improvement of the RAG pipeline. Future works could also look into using fine-tuned LLMs.
- **Support to add news competencies and ISO processes into the system:** The current implementation does not support adding new competencies or ISO processes into the framework. Guidelines are provided in the respective implementation sections on how this features should be implemented and things to consider.
- **Extending Organizational Features:** Current implementation supports tailoring role-process and process-competency mappings for individual organizations based on their needs and conducting the assessment employing the mappings unique to the organizational context. It also supports analyzing the assessment results of individuals and teams. Future improvement could include automated organizational insights and competency trend analysis.

8.4 Final Remarks

This research successfully developed a scientifically grounded and validated web application for Systems Engineering Competency Assessment. Through evaluation and statistical validation, the system was shown to align with established SE standards while delivering a user-friendly and intuitive experience. The integration of LLM-assisted evaluations demonstrated the potential of AI-driven validation and research techniques.

While challenges remain, the study provides clear directions for improvement and guidelines for future enhancements. The work lays a strong foundation for further research and development, not only in SE competency assessment but also in exploring how software engineering techniques can address challenges in other domains, such as Systems Engineering and beyond.

9 Future Works

9.1 Summary of Key Outcomes

9.2 Contributions

9.3 Future Research and Enhancements

lorem ipsum.

10 Bibliography

[AG24] AYYAMPERUMAL, S. G.; GE, L.: Current State of LLM Risks and AI Guardrails. In: *arXiv preprint* (2024). https://arxiv.org/abs/2406.12934

[AMK12] AHMED, N.; MATULEVIČIUS, R. ; KHAN, N. H.: Eliciting security requirements for business processes using patterns. In: *Proceedings of the 9th International Workshop on Security in Information Systems* (2012). http://dx.doi.org/10.5220/0004100200490058. – DOI 10.5220/0004100200490058

[Ang25] ANGULAR TEAM: *Angular Official Documentation*. https://angular.dev/. Version: 2025. – Accessed: 2025-03-06

[BAE24] BAE SYSTEMS: *BAE Systems Homepage*. https://www.baesystems.com/en/home. Version: 2024. – Accessed: [date]

[Bak01] BAKER, F. B.: *The Basics of Item Response Theory*. Second. College Park, MD : ERIC Clearinghouse on Assessment and Evaluation, 2001

[Bea13] BEASLEY, R.: 9.3.2 The need to tailor competency models – with a use case from Rolls-Royce. In: *INCOSE International Symposium* 23 (2013), Jun, Nr. 1, 428–442. http://dx.doi.org/https://doi.org/10.1002/j.2334-5837.2013.tb03029.x. – DOI https://doi.org/10.1002/j.2334–5837.2013.tb03029.x

[Bea17] BEASLEY, R.: Realizing the value of systems engineering. In: *INCOSE International Symposium* 27 (2017), S. 1100–1113. http://dx.doi.org/10.1002/j.2334-5837.2017.00415.x. – DOI 10.1002/j.2334–5837.2017.00415.x

[Beb25] BEBITY: *LinkedIn Jobs Scraper*. https://apify.com/bebity/linkedin-jobs-scraper. Version: 2025. – Accessed: 2025-03-15

[Blo56] BLOOM, B. S.: *Taxonomy of Educational Objectives: The Classification of Educational Goals, Volume 1*. New York : Longmans, Green, 1956

[BMA20] BARON, I. S.; MELANIA, M. ; AGUSTINA, H.: The role of psychological testing as an effort to improve employee competency. In: *GATR Journal of Management and Marketing Review* 5 (2020), S. 1–15. http://dx.doi.org/10.35609/jmmr.2020.5.1(1). – DOI 10.35609/jmmr.2020.5.1(1)

[BMR+20] BROWN, T. B.; MANN, B.; RYDER, N.; SUBBIAH, M.; KAPLAN, J.; DHARIWAL, P.; NEELAKANTAN, A.; SHYAM, P.; SASTRY, G.; ASKELL, A.; AGARWAL, S.; HERBERT-VOSS, A.; KRUEGER, G.; HENIGHAN, T.; CHILD, R.; RAMESH, A.; ZIEGLER, D. M.; WU, J.; WINTER, C.; HESSE, C.; CHEN, M.; SIGLER, E.; LITWIN, M.; GRAY, S.; CHESS, B.; CLARK, J.; BERNER, C.; MCCANDLISH, S.; RADFORD, A.; SUTSKEVER, I. ; AMODEI, D.: *Language Models are Few-Shot Learners*. https://arxiv.org/abs/2005.14165. Version: 2020

[BSF+20] BRAUN, A.; SCHIFFBÄNKER, P.; FALLS, B. ; KÜPPER, K.: Case study: battery system development. (2020), S. 1–23. http://dx.doi.org/10.1007/978-3-319-68847-3_20-2. – DOI 10.1007/978–3–319–68847–$3_2$0 – –2

[CCW+02] COHEN, J.; COHEN, P.; WEST, S. G. ; AIKEN, L. S.: *Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences*. 3rd. Routledge, 2002. http://dx.doi.org/10.4324/9780203774441. http://dx.doi.org/10.4324/9780203774441

[chr25] https://www.trychroma.com/

[CIP19] CIPD: *Competence and Competency Frameworks*. https://www.cipd.org/en/knowledge/factsheets/competency-factsheet/#what. Version: 2019. – Accessed: YYYY-MM-DD

[CP05] CHOI, B.; PAK, A.: A catalog of biases in questionnaires. In: *Preventing Chronic Disease* 2 (2005), Jan, Nr. 1, S. A13. – Epub 2004 Dec 15

[Dan19] DANO, E. B.: A Validated Systems Engineering Competency Methodology and Function-al/Domain Competency Assessment Tool. In: *2019 International Symposium on Systems Engineering (ISSE)*, 2019, S. 1–7

[DAR+21] DUMITRESCU, R.; ALBERS, A.; RIEDEL, O.; STARK, R. ; GAUSEMEIER, J. H.: *Engineering in Germany - The status quo in business and science, a contribution to Advanced Systems Engineering*. Paderborn, 2021

[DD80] DREYFUS, S. E.; DREYFUS, H. L.: A Five-Stage Model of the Mental Activities Involved in Directed Skill Acquisition / University of California, Berkeley.

Version: Feb 1980. https://www.researchgate.net/publication/235125013_A_Five-Stage_Model_of_the_Mental_Activities_Involved_in_Directed_Skill_Acquisition. 1980 (ORC 80-2). – Forschungsbericht. – 22 S.

[DD17] DEZA, M. M.; DEZA, E.: *Encyclopedia of Distances*. Springer Berlin Heidelberg, 2017. http://dx.doi.org/10.1007/978-3-642-30958-8. http://dx.doi.org/10.1007/978-3-642-30958-8. – ISBN 978–3–642–30958–8

[Der24] DERVINIS, D.: Energy and Resource Modeling: A Comparative Analysis of Containers and Virtual Machines. In: *Professional Studies: Theory and Practice* 28 (2024), Nr. 2, S. 29–34. http://dx.doi.org/10.56131/pstp.2024.28.2.323. – DOI 10.56131/p-stp.2024.28.2.323

[DGD+25] DOUZE, M.; GUZHVA, A.; DENG, C.; JOHNSON, J.; SZILVASY, G.; MAZARÉ, P.-E.; LOMELI, M.; HOSSEINI, L. ; JÉGOU, H.: *The Faiss library*. https://arxiv.org/abs/2401.08281. Version: 2025

[DJK24] DONVIR, A.; JAIN, A. ; KUMAR, S. P.: Application State Management (ASM) in the Modern Web and Mobile Applications: A Comprehensive Review. In: *arXiv* (2024), Jul. http://dx.doi.org/10.48550/arxiv.2407.19318. – DOI 10.48550/arxiv.2407.19318. – Preprint

[Doc22] DOCKER TEAM: *Docker: Accelerated Container Application Development*. https://www.docker.com/. Version: May 2022. – Accessed: 2025-03-06

[Doc24] DOCKER TEAM: *Docker Compose Documentation*. https://docs.docker.com/compose/. Version: 2024. – Accessed: 2025-03-06

[Doc25] DOCKER TEAM: *PostgreSQL - Official Docker Image*. https://hub.docker.com/_/postgres. Version: 2025. – Accessed: 2025-03-06

[EBR+18] ERKOYUNCU, J. A.; BUTALA, P.; ROY, R. u. a.: Digital twins: Understanding the added value of integrated models for through-life engineering services. In: *Procedia Manufacturing* 16 (2018), S. 139–146

[EJEA+23] ES, S.; JAMES, J.; ESPINOSA-ANKE, L. ; SCHOCKAERT, S.: RAGAS: Automated Evaluation of Retrieval Augmented Generation. In: *arXiv preprint* (2023). https://arxiv.org/abs/2309.15217

[Eng05] ENGELHARD, G.: Guttman Scaling. Version: 2005. http://dx.doi.org/https://doi.org/10.1016/B0-12-369398-5/00103-1. In: KEMPF-LEONARD, K.

(Hrsg.): *Encyclopedia of Social Measurement*. New York : Elsevier, 2005. – DOI https://doi.org/10.1016/B0–12–369398–5/00103–1. – ISBN 978–0–12–369398–3, 167-174

[exp23] *Automotive SPICE 4.0 – The Process Guide and Workbook*. Expleo, 2023 https://expleo.com/global/en/insights/whitepapers/automotive-spice-workbook/

[Fie17] FIELD, A.: *Discovering Statistics Using IBM SPSS Statistics*. 5th. SAGE Publications, 2017 http://sro.sussex.ac.uk/id/eprint/77862/

[Fis92a] In: FISHER, R. A.: *Statistical Methods for Research Workers*. New York, NY : Springer New York, 1992. – ISBN 978–1–4612–4380–9, 66–70

[Fis92b] In: FISHER, R. A.: *Statistical Methods for Research Workers*. New York, NY : Springer New York, 1992. – ISBN 978–1–4612–4380–9, 66–70

[Fla] FLASK DEVELOPERS: *Flask: Web Development, One Drop at a Time*. https://flask.palletsprojects.com/. – Accessed: 2024-09-15

[Fla25a] FLASK DOCUMENTATION: *Welcome to Flask — Flask Documentation (3.1.x)*. https://flask.palletsprojects.com/en/stable/. Version: 2025. – Accessed: 2025-03-06

[Fla25b] FLASK DOCUMENTATION TEAM: *Welcome to Flask — Flask Documentation (3.1.x)*. https://flask.palletsprojects.com/en/stable/. Version: 2025. – Accessed: 2025-03-06

[FWF00] FYODOROV, Y.; WINTER, Y. ; FRANCEZ, N.: A Natural Logic Inference System. In: *Proceedings of the 2nd Workshop on Inference in Computational Semantics (ICoS-2)*, 2000

[GHS+17] GELOSH, D. S.; HEISEY, M.; SNODERLY, J. R. ; NIDIFFER, K.: Version 0.75 of the Proposed INCOSE Competency Framework. In: *INCOSE International Symposium* 27 (2017), Jul, Nr. 1, 45–56. http://dx.doi.org/10.1002/j.2334-5837.2017.00344.x. – DOI 10.1002/j.2334–5837.2017.00344.x

[Gos08] GOSSET, W. S.: The Probable Error of a Mean. In: *Biometrika* 6 (1908), Nr. 1, S. 1–25. http://dx.doi.org/10.1093/biomet/6.1.1. – DOI 10.1093/biomet/6.1.1

[HBD+20] HOLTZMAN, A.; BUYS, J.; DU, L.; FORBES, M. ; CHOI, Y.: The Curious Case of Neural Text Degeneration. In: *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020

[Hic52] HICK, W. E.: On the Rate of Gain of Information. In: *Quarterly Journal of Experimental Psychology* 4 (1952), Nr. 1, S. 11–26. http://dx.doi.org/10.1080/17470215208416600. – DOI 10.1080/17470215208416600

[HP11] HOLT, J.; PERRY, S.: *A Pragmatic Guide to Competency: Tools, Frameworks and Assessment*. BCS Learning & Development Limited, 2011 (British Comp Society Series). https://books.google.de/books?id=W8sbngEACAAJ. – ISBN 9781906124700

[HTB+19] HUTCHISON, N.; TAO, H.; BURKE, P.; LUNA, S.; ZAVALA, A.; KOTHARI, S.; SONEJI, S. ; RAMÍREZ-MÁRQUEZ, J.: Evolution of the helix project: from investigating the effectiveness of individual systems engineers to systems engineering organizations. In: *Incose International Symposium* 29 (2019), S. 652–668. http://dx.doi.org/10.1002/j.2334-5837.2019.00626.x. – DOI 10.1002/j.2334–5837.2019.00626.x

[HWL17] HUTCHISON, N.; WADE, J. ; LUNA, S.: The Roles of Systems Engineers Revisited. In: *INCOSE International Symposium* 27 (2017), Nr. 1, 200-213. http://dx.doi.org/https://doi.org/10.1002/j.2334-5837.2017.00354.x. – DOI https://doi.org/10.1002/j.2334–5837.2017.00354.x

[INC22] INCOSE: *Professional Development Portal (PDP)*. https://www.incose.org/learn/incose-pdp. Version: 2022. – Accessed: [date]

[INC23a] INCOSE: *ESEP Interviews*. https://www.incose.org/certification/becoming-certified/applying-for-esep/esep-interviews. Version: 2023. – Accessed: [date]

[INC23b] INCOSE ; WHITCOMB, C. (Hrsg.); ZIPES, L. (Hrsg.) ; PRESLAND, I. (Hrsg.): *Systems Engineering Competency Assessment Guide: A combined INCOSE Systems Engineering Competency Framework (SECF) and associated Systems Engineering Competency Assessment Guide (SECAG) document*. John Wiley & Sons, Inc., 2023. http://dx.doi.org/10.1002/9781119862581. http://dx.doi.org/10.1002/9781119862581. – First published: 20 January 2023

[INC23c] INCOSE: Systems Engineering Handbook V5. In: *INCOSE Publications*, 2023

[INC24] INCOSE: Session Track V3: SysML - Full Access: INCOSE Systems Engineering Competency Assessment Guide Systems Modeling Language (SysML) Model Description. In: *INCOSE International Symposium* 34 (2024), July, Nr. 1. – Special Issue: 34th Annual INCOSE International Symposium, Hybrid Event, Dublin, Ireland, July 2–6, 2024

[INC25] INCOSE: *INCOSE Certification Program*. https://www.incose.org/certification. Version: 2025. – Accessed: [date]

[Ins17] INSTITUTE, P. M.: *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*. 6th. Newtown Square, PA : Project Management Institute, 2017

[Int23a] INTERNATIONAL COUNCIL ON SYSTEMS ENGINEERING: *INCOSE Website*. https://www.incose.org/. Version: 2023. – Accessed: YYYY-MM-DD

[Int23b] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION: *ISO/IEC/IEEE 15288:2023 - Systems and Software Engineering: System Life Cycle Processes*. – ISO/IEC/IEEE 15288:2023(E)

[Jav23] JAVAID, M. H.: *Systems Engineering Competency Checker*. Paderborn, Universität Paderborn, Master's Thesis, Feb 2023. – Submitted by cand. Muhammad Haseeb Javaid, Matr.-Nr. 6856740. Company: Fraunhofer IPT

[JKM13] JUNE, S.; KHENG, Y. K. ; MAHMOOD, R.: Determining the importance of competency and person-job fit for the job performance of service smes employees in malaysia. In: *Asian Social Science* 9 (2013). http://dx.doi.org/10.5539/ass.v9n10p114. – DOI 10.5539/ass.v9n10p114

[jsP25] JSPDF MAINTAINERS: *jsPDF - Client-Side PDF Generation for JavaScript*. https://www.npmjs.com/package/jspdf. Version: Feb 2025. – Accessed: 2025-03-15

[Kah11] KAHNEMAN, D.: *Thinking, Fast and Slow*. New York : Farrar, Straus and Giroux, 2011

[KFZ+10] KASSER, J.; FRANK, M.; ZHAO, Y. Y. u. a.: Assessing the competencies of systems engineers. In: *Proceedings of the 7th bi-annual European SE conference (EUSEC), Stockholm, Sweden, may 23* Bd. 26, 2010, S. 2010

[KPG17] KLEIN, P.; PONZETTO, S. P. ; GLAVAŠ, G.: Improving Neural Knowledge Base Completion with Cross-Lingual Projections. In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. Valencia, Spain : Association for Computational Linguistics, 2017, 516–522

[KWA+22] KÖNEMANN, U.; WILKE, D.; ANACKER, H. ; DUMITRESCU, R.: Identification of stakeholder-specific Systems Engineering competencies for industry. In: *2022 IEEE International Systems Conference (SysCon)*, 2022, S. 1–7

[Lan] LANGCHAIN DEVELOPERS: *LangChain: Building applications with LLMs through composability*. https://langchain.com/. – Accessed: 2024-09-15 [lan23] https://python.langchain.com/docs/tutorials/rag/

[Lan24] LANGCHAIN TEAM: *LangChain Documentation*. https://python.langchain.com/docs/how_to/. Version: 2024. – Accessed: 2025-03-06

[Lan25a] LANGCHAIN-AI: *GitHub - LangChain-AI/LangChain: Build Context-Aware Reasoning Applications*. https://github.com/langchain-ai/langchain. Version: Mar 2025. – Accessed: 2025-03-06

[Lan25b] LANGCHAIN TEAM: *LangChain Official Documentation*. https://www.langchain.com/. Version: 2025. – Accessed: 2025-03-06

[Les16] LESLIE, C.: Engineering Competency Model. In: *2016 ASEE Annual Conference & Exposition*. New Orleans, Louisiana : ASEE Conferences, June 2016

[LGP+23] LIM, W. M.; GUNASEKARA, A.; PALLANT, J. L.; PALLANT, J. I. ; PECHENKINA, E.: Generative AI and the future of education: Ragnarök or reformation? A paradoxical perspective from management educators. In: *The International Journal of Management Education* 21 (2023), Nr. 2, 100790. http://dx.doi.org/https://doi.org/10.1016/j.ijme.2023.100790. – DOI https://doi.org/10.1016/j.ijme.2023.100790. – ISSN 1472–8117

[Lik32] LIKERT, R.: A Technique for the Measurement of Attitudes. In: *Archives of Psychology* 140 (1932), S. 1–55

[Lin06] LINDÉN, K.: Multilingual Modeling of Cross-Lingual Spelling Variants. In: *Information Retrieval* 9 (2006), Nr. 3, S. 295–310. http://dx.doi.org/10.1007/s10791-006-1541-5. – DOI 10.1007/s10791–006–1541–5

[Lin25] LINKEDIN: *LinkedIn - Professional Networking Platform*. https://www.linkedin.com/. Version: 2025. – Accessed: 2025-03-15

[LPP+21] LEWIS, P.; PEREZ, E.; PIKTUS, A.; PETRONI, F.; KARPUKHIN, V.; GOYAL, N.; KÜTTLER, H.; LEWIS, M.; YIH, W. tau; ROCKTÄSCHEL, T.; RIEDEL, S. ; KIELA, D.: *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*. https://arxiv.org/abs/2005.11401. Version: 2021

[Lv23] LV, Z.: Generative artificial intelligence in the metaverse era. In: *Cognitive Robotics* 3 (2023), 208-217. http://dx.doi.org/https://doi.org/10.1016/j.cogr.2023.06.001. – DOI https://doi.org/10.1016/j.cogr.2023.06.001. – ISSN 2667–2413

[LZ21] LI, N.; ZHANG, B.: The Research on Single Page Application Front-End Development Based on Vue. In: *Journal of Physics: Conference Series* 1883 (2021), Nr. 1, S. 012030. http://dx.doi.org/10.1088/1742-6596/1883/1/012030. – DOI 10.1088/1742–6596/1883/1/012030

[MCC+13] MIKOLOV, T.; CHEN, K.; CORRADO, G. ; DEAN, J.: *Efficient Estimation of Word Representations in Vector Space*. https://arxiv.org/abs/1301.3781. Version: 2013

[Mer14] MERKEL, D.: Docker: Lightweight Linux Containers for Consistent Development and Deployment. In: *Linux Journal* 2014 (2014), Nr. 239. http://dx.doi.org/10.5555/2600239.2600241. – DOI 10.5555/2600239.2600241

[Met13] META (FACEBOOK) DEVELOPERS: *React: A JavaScript library for building user interfaces*. https://reactjs.org/. Version: 2013. – Accessed: 2024-09-15

[MyS95] MYSQL COMMUNITY: *MySQL: The World's Most Popular Open Source Database*. https://www.mysql.com/. Version: 1995. – Accessed: 2024-09-15

[MyS25] MYSQL TEAM: *MySQL Official Documentation*. https://www.mysql.com/. Version: 2025. – Accessed: 2025-03-06

[Nod09] NODE.JS DEVELOPERS: *Node.js: JavaScript runtime built on Chrome's V8 JavaScript engine*. https://nodejs.org/. Version: 2009. – Accessed: 2024-09-15

[Ope24a] OPENAI: *GPT-4o Mini Model Documentation*. https://platform.openai.com/docs/models/gpt-4o-mini. Version: 2024. – Accessed: 2025-03-06

[Ope24b] OPENAI: *Text Embedding Ada-002 Model Documentation*. https://platform.openai.com/docs/models/text-embedding-ada-002. Version: 2024. – Accessed: 2025-03-06

[Pal25] PALLETS TEAM: *Flask - GitHub Repository*. https://github.com/pallets/flask. Version: 2025. – Accessed: 2025-03-06

[Pea95] PEARSON, K.: X. Contributions to the Mathematical Theory of Evolution—II. Skew Variation in Homogeneous Material. In: *Philosophical Transactions of the Royal Society of London A* 186 (1895), 343–414. http://dx.doi.org/10.1098/rsta.1895.0010. – DOI 10.1098/rsta.1895.0010

[Pin19] PINIA TEAM: *Pinia - The Vue Store*. https://pinia.vuejs.org/. Version: 2019. – Accessed: 2025-03-06

[Pos96] POSTGRESQL GLOBAL DEVELOPMENT GROUP: *PostgreSQL: The World's Most Advanced Open Source Relational Database*. https://www.postgresql.org/. Version: 1996. – Accessed: 2024-09-15

[Pos25] POSTGRESQL GLOBAL DEVELOPMENT GROUP: *PostgreSQL Official Documentation*. https://www.postgresql.org/. Version: Mar 2025. – Accessed: 2025-03-06

[PWV+24] PLAAT, A.; WONG, A.; VERBERNE, S.; BROEKENS, J.; STEIN, N. van ; BACK, T.: *Reasoning with Large Language Models, a Survey*. https://arxiv.org/abs/2407.11511. Version: 2024

[RAG24] RAGAS TEAM: *RAGAS Documentation*. https://docs.ragas.io/en/stable/. Version: 2024. – Accessed: 2025-03-15

[RAG25] RAGAS TEAM: *Factual Correctness Metric in RAGAS*. https://docs.ragas.io/en/stable/concepts/metrics/available_metrics/factual_correctness/#factual-correctness. Version: 2025. – Accessed: 2025-03-15

[RC03] RING, J.; COMPONATION, P.: *A Guide to the Systems Engineering Body of Knowledge (SEBoK) III: Competency Development of SE Practitioners*. http://g2sebok.incose.org. Version: 2003. – INCOSE document

[Rea15] REACT TEAM: *React Official Documentation*. https://react.dev/. Version: 2015. – Accessed: 2025-03-06

[RWC+19] RADFORD, A.; WU, J.; CHILD, R.; LUAN, D.; AMODEI, D. ; SUTSKEVER, I.: Language Models are Unsupervised Multitask Learners. In: *OpenAI Blog* (2019). https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

[SANS12] SAHRAOUI, S.; AL-NAHAS, N. ; SULEIMAN, R.: An Agile Perspective on Open Source Software Engineering. In: *International Journal of Open Source Software and Processes* 4 (2012), Nr. 3, S. 52–65. http://dx.doi.org/10.4018/ijossp.2012070105. – DOI 10.4018/ijossp.2012070105

[SB02] SCHWABER, K.; BEEDLE, M.: *Agile Software Development with Scrum*. Prentice Hall, 2002

[SEB24] SEBOK AUTHORS ; NICOLE HUTCHISON, E. i. C. (Hrsg.): *Determining Needed Systems Engineering Capabilities in Businesses and Enterprises*. https://sebokwiki.org/w/index.php?title=Determining_Needed_Systems_

Engineering_Capabilities_in_Businesses_and_Enterprises&oldid=71814, May 2024

[sec23] https://www.incose.org/publications/products/competency-framework

[SH22] SZABÓ, B.; HERCEGFI, K.: User-centered approaches in software development processes: Qualitative research into the practice of Hungarian companies. In: *Journal of Software: Evolution and Process* 35 (2022), Aug, Nr. 2. http://dx.doi.org/10.1002/smr.2501. – DOI 10.1002/smr.2501

[She96] SHEARD, S. A.: Twelve Systems Engineering Roles. In: *INCOSE International Symposium* 6 (1996), Jul, Nr. 1, 478–485. http://dx.doi.org/10.1002/j.2334-5837.1996.tb02042.x. – DOI 10.1002/j.2334–5837.1996.tb02042.x

[SM83] SALTON, G.; MCGILL, M. J.: *Introduction to Modern Information Retrieval*. McGraw-Hill, 1983

[Sma25] SMART, J. F.: *How Great Teams Deliver Great Products*. https://johnfergusonsmart.com/how-great-teams-deliver-great-products/asked-for-needed-delivered/. Version: 2025. – Accessed: 2025-02-23

[SRC+09] SACCO, M.; REDAELLI, C.; CÂNDEA, C. ; GEORGESCU, A. V.: DiFac: an integrated scenario for the Digital Factory. In: *2009 IEEE International Technology Management Conference (ICE)* (2009), 1-8. https://api.semanticscholar.org/CorpusID:23600919

[SSD+24] SILLITTO, H.; SQUIRES, A.; DAVIDZ, H.; PYSTER, A. ; BEASLEY, R.: *Assessing Systems Engineering Performance of Business and Enterprises*. https://sebokwiki.org/wiki/Assessing_Systems_Engineering_Performance_of_Business_and_Enterprises. Version: 2024

[Sys25] SYSML.ORG: *SysML Open Source Project: What is SysML? Who Created It?* https://sysml.org/. Version: 2025. – Accessed: [date]

[SZX19] SONG, J.; ZHANG, M. ; XIE, H.: Design and Implementation of a Vue.js-Based College Teaching System. In: *International Journal of Emerging Technologies in Learning (iJET)* 14 (2019), Nr. 13, S. 59. http://dx.doi.org/10.3991/ijet.v14i13.10709. – DOI 10.3991/ijet.v14i13.10709

[Tha20] THARWAT, A.: Classification Assessment Methods. In: *Applied Computing and Informatics* 17 (2020), Nr. 1, 168–192. http://dx.doi.org/10.1016/j.aci.2018.08.003. – DOI 10.1016/j.aci.2018.08.003

[VGO+20] VIRTANEN, P.; GOMMERS, R.; OLIPHANT, T. E.; HABERLAND, M.; REDDY, T.; COURNAPEAU, D.; BUROVSKI, E.; PETERSON, P.; WECKESSER, W.; BRIGHT, J.; WALT, S. J. d.; BRETT, M.; WILSON, J.; MILLMAN, K. J.; MAYOROV, N.; NELSON, A. R. J.; JONES, E.; KERN, R.; LARSON, E.; CAREY, C. J.; POLAT, ; FENG, Y.; MOORE, E. W.; VANDERPLAS, J.; LAXALDE, D.; PERKTOLD, J.; CIMRMAN, R.; HENRIKSEN, I.; QUINTERO, E. A.; HARRIS, C. R.; ARCHIBALD, A. M.; RIBEIRO, A. H.; PEDREGOSA, F.; MULBREGT, P. van ; CONTRIBUTORS, S. ..: SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. In: *Nature Methods* 17 (2020), March, Nr. 3, S. 261–272. http://dx.doi.org/10.1038/s41592-019-0686-2. – DOI 10.1038/s41592–019–0686–2

[VNK24] VU, D. C.; NGUYEN, T. V. ; KITJAROONCHAI, N.: Exploring the relationship between working memory capacity and l2 oral fluency. In: *Theory and Practice in Language Studies* 14 (2024), S. 2002–2012. http://dx.doi.org/10.17507/tpls.1407.06. – DOI 10.17507/tpls.1407.06

[VR79] VAN RIJSBERGEN, C. J.: *Information Retrieval*. 2nd. Butterworth-Heinemann, 1979. http://dx.doi.org/10.1002/asi.4630300621. http://dx.doi.org/10.1002/asi.4630300621. – Accessed: 2025-03-15

[VR08] VDI-RICHTLINIEN, V.: *4499 Blatt 1: Digitale Fabrik Grundlagen, Berlin Düsseldorf*. 2008

[VSP+23] VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, L. ; POLOSUKHIN, I.: *Attention Is All You Need*. https://arxiv.org/abs/1706.03762. Version: 2023

[Vue14] VUE.JS COMMUNITY: *Vue.js: The Progressive JavaScript Framework*. https://vuejs.org/. Version: 2014. – Accessed: 2024-09-15

[Vue19] VUE CHART.JS TEAM: *Vue Chart.js Documentation*. https://vue-chartjs.org/. Version: 2019. – Accessed: YYYY-MM-DD

[Vue23] VUE.JS TEAM: *Vue.js Documentation: Introduction Guide*. https://vuejs.org/guide/introduction.html. Version: 2023. – Accessed: 2025-03-06

[Vue25] VUE.JS TEAM: *Vue.js Official Website*. https://vuejs.org/. Version: 2025. – Accessed: 2025-03-06

[WM24] WULF, J.; MEIERHOFER, J.: *Exploring the Potential of Large Language Models for Automation in Technical Customer Service*

[Wri24] WRIGHT, D. K.: *The ultimate guide to competency assessment in health care*. Springer Publishing Company, 2024

[XJK24] XU, Z.; JAIN, S. ; KANKANHALLI, M.: *Hallucination is Inevitable: An Innate Limitation of Large Language Models*. `https://arxiv.org/abs/2401.11817`. Version: 2024

[Zan24] ZANEVYCH, Y.: Flask vs. Django vs. Spring Boot: Navigating Framework Choices for Machine Learning Object Detection Projects. In: *Education and Science of Today: Intersectoral Issues and Development of Sciences* (2024). `http://dx.doi.org/10.36074/logos-29.03.2024.066`. – DOI 10.36074/logos–29.03.2024.066

[ZJL23] ZHANG, Y.; JI, Q. ; LI, D.: A Dynamic Form Rendering Method for Web Multi-Front Framework. In: *Proceedings of the International Conference on Computer, Artificial Intelligence, and Control Engineering (CAICE 2023)*, 2023

[ZSD+14] ZEC, M.; SCHNEIDER, A. W.; DÜRR, P. ; MATTHES, F.: Improving computer-support for collaborative business model design and exploration. In: *Proceedings of the Fourth International Symposium on Business Modeling and Software Design* (2014). `http://dx.doi.org/10.5220/0005424000290037`. – DOI 10.5220/0005424000290037

Appendix

A Title of appendix A

Lorem ipsum

A.1 Title of appendix A.1

Lorem ipsum