Human Factors in Software and Systems Engineering, Vol. 181, 2025, 25–33
https://doi.org/10.54941/ahfe1006391

IT Tool Stack Optimization in Collaborative Projects: An Evaluation and Recommendation Framework

Can Çağıncan[1], Juliane Balder[2], Rainer Stark[2], and Roland Jochem[1]

[1]Department of Quality Science, Technical University of Berlin, Berlin 10587, Germany
[2]Department of Industrial IT, Technical University of Berlin, Berlin 10587, Germany

ABSTRACT

In modern industrial engineering, configuring IT tool systems is fundamental to ensuring productivity and quality in collaborative projects. Small and medium-sized enterprises (SMEs) face distinctive challenges in selecting and adopting these systems—not only due to limited IT and AI expertise but also because of insufficient consideration of human perceptual factors such as technology acceptance and subjective practical experience. These challenges adversely affect overall quality, productivity, and technology adoption. This study proposes a user-centric framework for evaluating and recommending IT tools for system design. The framework employs standardized human-centric evaluation methods based on the Critical to Quality (CTQ) methodology. By integrating requirements engineering with machine learning (ML) models, including collaborative and content-based filtering techniques—the approach systematically analyzes data to identify similarities among users and project archetypes, thereby recommending the most effective tool configurations. Moreover, ML models are utilized to refine recommendations by matching individuals across projects and incorporating cognitive factors related to perceived tool usage efficiency. This systematic approach to IT tool stack configuration aligns with organizational objectives and project-specific requirements, ultimately enhancing collaborative capabilities, productivity, and technology adoption and acceptance rates in SMEs.

Keywords: It tool stack, Collaborative projects, Recommendation systems, SMEs, Machine learning models, System design, Collaborative product development, Human-centric evaluation, AI integration, Technology adoption, Tool stack optimization

INTRODUCTION

In today’s digital world, software tools are akin to modern sonic screwdrivers—essential instruments for executing tasks. The use and effectiveness of these IT tools vary among individuals and organizations, directly influencing both productivity and product quality (Mauerhoefer, 2017). When multiple independent tools are combined to perform broader and more complex tasks, an IT Tool Stack is formed.

The concept of an IT Tool Stack originated from modular coding practices in the early 1970s, aimed at handling complex tasks more efficiently and flexibly. Today, this concept has evolved into various standalone software tools from different companies, each with extensive features, which can be used either in parallel or integrated together.

The increasing complexity and scale of products, combined with the need for specialized expertise in niche fields, make inter-company collaboration more essential than ever. Successful collaborative projects heavily depend on the proper configuration of IT tools during the initial planning phase that is often underestimated by businesses (Anderl, 2012). Additionally, the short lifecycles of many collaborative projects result in the frequent involvement of multiple partners over brief periods, necessitating frequent readjustments of the IT Tool Stack.

Moreover, the integration of artificial intelligence (AI) into business models is revolutionizing how companies operate and compete (Lee, 2018). This transformation enables companies to be more agile and responsive to market changes, ensuring they remain competitive and relevant in an increasingly digital and data-driven economy. As AI continues to evolve rapidly, the development of AI technologies has introduced specialized roles and cutting-edge methods, along with innovative tools that continually emerge in industry. Like any significant innovation, the daily emergence of more powerful products and models continually expands its impact on business models. This evolution drives further efficiencies and opens new possibilities for growth and differentiation. Consequently, this progress demands greater expertise in AI technologies to understand their differences, capabilities, and limitations to configure IT Tool Stack.

Small and medium-sized enterprises (SMEs) face significant challenges in IT adaptation (Prasanna, 2019), despite frequently collaborating with other companies and playing crucial roles in joint development and supply chains with larger firms. Collaborative capabilities are vital for SMEs' survival. However, IT adaptation can be demanding due to the lack of IT and AI expertise needed to optimize and implement efficient tool stacks. The existing data, IT infrastructure and the need for tailored solutions, especially for AI applications, require highly customized solutions depending on specific cases within the same industry, further complicate this challenge. All results in limitations in productivity, collaborative capacities, and a slower adoption rate of AI. The adoption of new AI tools and business models will eventually set new industry standards, compelling all market participants to comply. Large enterprises are already creating and optimizing their own AI models (Zhou, 2019).

These dynamics emphasize the importance of a systematic approach to selecting an IT tool stack that not only aligns with business goals but also fits the individuals within the organization and satisfies the unique needs of each collaboration. However, configuring an IT Tool Stack is a complex and often overlooked task, frequently addressed with quick, ad hoc solutions during the rush of a project. This hasty approach can severely impact both the quality and efficiency of the project's outcomes.

The task of configuring an IT Tool Stack is a complex, multi-criteria problem because today's sophisticated business methods require the integration of many specific functions from different IT tools. This complexity makes it difficult to define requirements, compare tools and functions, and make informed recommendations. Additionally, a significant challenge lies in defining quality metrics to assess an IT Tool Stack, as these metrics are often not well defined and key performance indicators (KPIs) can vary greatly between use cases.

METHODOLOGY

Our goal is to create a recommendation system and standardized evaluation method for IT Tool Stacks that also validates the recommendations. We aim to leverage collective experiences in the industry anonymously and utilize the most innovative solutions available in the market to optimize IT tools. By utilizing this approach, businesses may discover innovative solutions through real-world examples derived from the experiences of similar individuals, allowing for their effective implementation.

In practice, businesses provide non-confidential information in a standardized manner via a survey. This survey includes basic details about their organization, previous collaborative projects, and the IT Tool Stacks used in those projects. The recommendation is given through analysis, after evaluating the current tool stack. Every single use case is characterized individually depending on variables in a collaborative project and similarities between are used for recommendation. Afterwards a strategic exchange of software within the IT stack to better align with the requirements of a specific cooperation project can be done. And performance of new configuration can be monitored with developed dashboards with declared KPIs.

SCOPE OF INVESTIGATION

There are different collaborative projects that involve numerous activities across various tools available in the market. That requires not only identifying the properties of these tools but also collecting sufficient samples homogenously from use cases, which demands extensive effort and data. Here, our focus is on the activities typically performed in collaborative projects, which makes this task easier. However, defining strict 'collaborative activities' is challenging, as discussed in discussion section. Besides collaboration, we should further filter our research. Therefore, we focus on Collaborative Product Development Projects, where SMEs are often involved.

In typical collaborative product development projects, ten different scenarios are constructed from real use cases with experts. These clusters have distinctly different characteristics in the activities that the business performs. The use cases are further clustered under these initial ten groups to build 'project archetypes' to map similarities between cases.

ENGINEERING COLLABORATION BUILDING BLOCKS (ECBB)

Software consists of various functions or activities performed during business operations. By listing IT tools' functionalities, we can determine an enterprise's digital operations, identify necessary IT activities, and recommend a more effective tool stack while introducing new tool functions and methods for improvement.

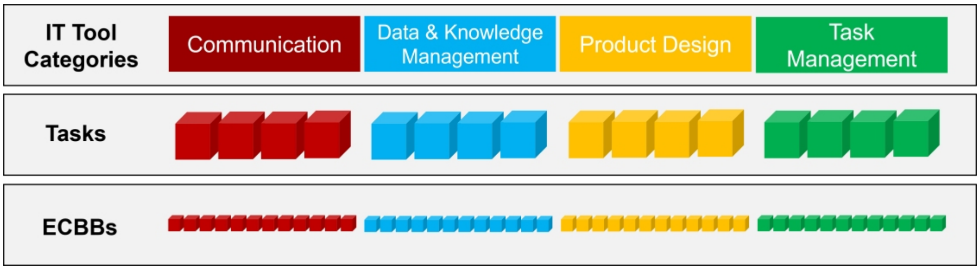

Figure 1: ECBBs decompose software into smaller, standardized parts, facilitating tool stack comparisons. While tools may encompass various functions, categorizing ECBBs under IT tool categories and tasks helps to effectively describe tools in models.

The ECBBs were developed to break down software into smaller standardized components, facilitating tool stack comparison and requirements listing. The challenge is defining the granularity of these components: they must be detailed enough for customization but not so granular that they recreate entire software without complexity. Through a comprehensive investigation, we compile a list of IT tools for product development. We then extract and standardize their functionalities to ECBBs, allowing us to compare the tools effectively within the context of product development.

IT tools are categorized into four main categories (Tucker, 2019), see Figure 1, each containing specific tasks and associated activities that describe the collaborative functions of software. Grouping activities under "tasks" within specific IT tool categories simplifies user input. Defining and standardizing a pool of ECBBs is crucial. Extensive research on literature, market trends, and patent databases identifies commonly used tools in Collaborative Product Development Projects and their functions.

EVALUATION METHOD

The framework of evaluation is built on the fundamental concept in Six Sigma, the Critical to Quality (CTQ) methodology (Krzysztof, 2015). CTQ is employed to identify quality characteristics of ECBBs.

These quality metrics are used to assess the usage and output efficiency of a function performed by a tool. The ratings of ECBBs as the time required to complete tasks and the output quality, gathered from users via the Voice of the Customer method, serve as default metrics for all ECBBs.

However, the goals of businesses, and therefore their quality and KPI can differ. Therefore, a project's KPIs must be identified and need to be translated into proper quality characteristics within the CTQ framework, in addition to default metrics. For instance, one project might prioritize the number of reviews or comments per unit time, while another might focus on the amount of data or files created per task, etc.

The software industry lacks standardized metrics for assessing software usage, automated methods (like APIs) for collecting tool usage statistics, and interfaces for displaying these statistics. Therefore, during the translation of KPIs to quality metrics, available data should be examined, and only measurable metrics should be declared. Manual information retrieval and input about tool usage can be used. Initially, quality metrics are measured through surveys as a form of implicit data collection.

FEATURE SELECTION

An extensive literature review is performed to identify all important variables in a cooperation project (Peng, 2014). These features should be able to use to compare different examples, find similarities, and identify similar use case groups called as project archetypes. However, defining the variables in a collaborative project is demanding due to numerous factors, including human.

The process primarily involves addressing critical questions such as establishing measurable nominal and ordinal features to identify collaborative use cases and recognizing the most correlated features to describe similarities between for data analysis to define project archetypes effectively.

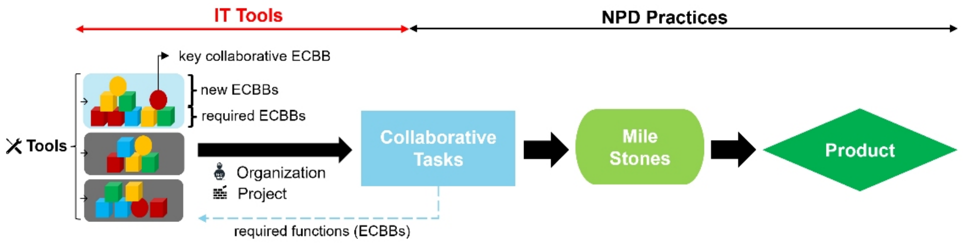

Figure 2: A comprehensive literature review was conducted to investigate factors that characterize successful collaborative projects. The identified variables are categorized under two main themes: Organization and project and used to describe similarities.

As shown in Figure 2, a collaborative project process is described with influencing parameters that affect overall quality. To achieve a product, businesses should complete planned milestones by performing collaborative tasks. Employed New Product Development (NPD) practices change the workflow of a project and therefore drastically affect the usage and needs of IT tools (Peng, 2014). Performing collaborative tasks is possible by accomplishing activities with IT Tools in parallel, and outcome quality depends not only on the tool stack in use but also on who is using the tools, and in which use case or situation.

Therefore, the tool stack, project, and organization are the main independent variables in our models for a collaborative project's success. Each headline includes many sub-parameters. The resulting sub-parameters are verified with experts. However, the parameters and the concept behind them are topics for another paper. For now, it is enough to know that these parameters are ordinal and nominal variables and can be used in data analysis to define similarities between different use cases.

It is important to include both the project and the organization simultaneously to cover more specific situations. This comprehensive approach ensures that the recommendations are not only relevant but also practical and applicable to the unique contexts of the types of projects and organizations involved.

RECOMMENDATION SYSTEM

The recommendation framework is built on requirement engineering and recommender system techniques, collaborative and content-based filtering. Collaborative filtering uses similarities between users and their interactions with tools, while content-based filtering uses similarities between tools to provide recommendations. For collaborative filtering, algorithms like Matrix Factorization (e.g., SVD, ALS) and Neighbourhood-based methods (k-NN) are used. For content-based filtering, decision trees, random forest, logistic regression, SVM or similarity measures such as Cosine Similarity and Euclidean distance, etc. are used.

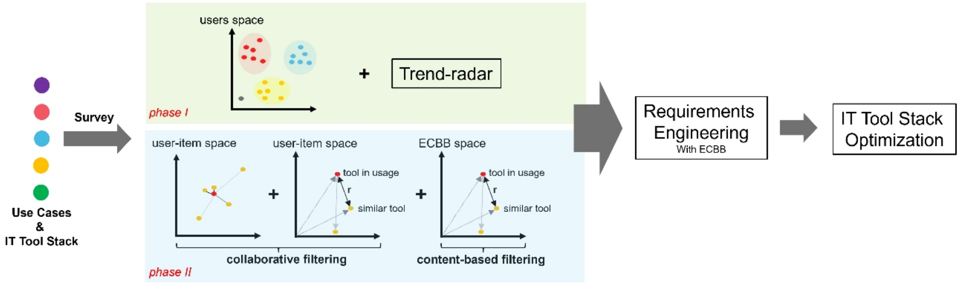

Figure 3: Implicit data acquisition is utilized to recommend ECBBs by identifying similarities between use cases within the dataset. The system is structured in two sequential phases, each aimed at progressively improving the quality of recommendations to better suit business needs. Following this process, requirement engineering guarantees that all necessary and innovative functions are incorporated into the new tool stack.

In general, the methods are based on a similar principle: vectorizing the information of use cases and IT tools. Afterwards, the vectors are mapped in an n-dimensional ECBB or User-Tool space and analysed with various approaches, such as measuring distances between the vectors to identify project archetypes. Project archetypes that are groups of similar cases that are used for recommendations within the groups. Recommendations can be made at different quality levels, ranging from tradational filtering recommendations for a group, to more tailored recommendations by ML models to a specific case. ML enables, identifying individual clusters, specific to a use case instead of project archetype. As with any ML project, the first step is to conduct exploratory and explanatory data analysis with domain knowledge before building the ML models.

Therefore, the project is planned in two phases, aiming to improve recommendation quality with more specific recommendations. (i) The first phase involves manual data analysis to identify project archetypes and obtain list of highly rated ECBBs of tools within groups. For every project archetype, a list of highly related ECBBs is combined with a list of the most popular and innovative tools on the market, obtained through trend radar research from databanks and official tool documentation. Expert consulting is crucial to validate the resulting list to ensure these ECBBs and tools work best for a specific project archetype. (ii) The second phase involves utilizing the results of the first phase to employ ML recommendation systems, to find similar use cases from real-world examples specific to that project and business, consider human factors.

Afterwards, requirement engineering should be performed. All required ECBBs, that the IT tool stack must cover, should be identified by analysing the current stack. All workers' and stakeholders' additional changes should be identified as ECBBs, and critical to quality metrics should be defined. Further requirement analysis, such as prioritizing requirements like must-have vs. nice-to-have or performing feasibility analysis to assess practicality with experts, should also be conducted. Here, an algorithm should be developed for requirement engineering since the requirements and ratings are in ECBB form. The best combination of tools needs to be calculated that covers all required ECBBs with high ratings. Additionally, new ECBBs that are used in similar cases and functioned well should also be recommended to the business.

After this point, businesses can decide amongst the recommendations according to their further needs, capabilities, or partners. This decision should be made after evaluating the current quality metrics of the IT tool stack, considering the pros and cons of transformation. The newly implemented tool is monitored to validate and track progress, through a developed user-friendly dashboard.

DISCUSSION

Data quality is paramount for effective recommendations. To expand the dataset, a survey is created using an interactive, user-friendly web page. The survey gathers non-confidential information to encourage higher participation rates. This web page also supports ongoing online data acquisition, which is critical for ML methods that continuously refine model performance.

The process of transforming information into data is critical. It involves creating standardized ECBBs with the right level of granularity, able to describe all tools, and selecting highly correlated features of cooperative projects. These foundational elements are essential for identifying similarities between cases. Over 87 activities are defined under 13 tasks across the four IT Tool categories. Efforts to refine granularity continue, aiming to simplify user input and better describe available activities. The ECBB framework is complex and detailed, warranting its own dedicated paper. The information provided here is sufficient to understand the model's construction.

Some activities might be conducted independently by partner companies rather than collaboratively. However, since the activities remain consistent, ECBBs should incorporate collaborative and non-collaborative aspects as additional features and promote them during recommendation. Promoting and adopting SaaS technologies is also a critical factor, as it enables businesses to rapidly implement the IT tool without concerns about IT infrastructure (Benlian, 2011), maintenance, scalability, or reverting to previous settings.

Recommendation systems have their own strengths and limitations, depending on the case and dataset available. Content-based filtering delivers personalized recommendations by matching item attributes to user profiles, which is particularly effective for new users or items. In our scenario, initially used ECBBs can be included as item-user actions. However, this approach may overlook subtle preferences and lacks diversity. Collaborative filtering harnesses user collective preferences, providing diverse suggestions and discovering new items. It faces challenges with cold start issues, where there is limited data on new users or items, and data sparsity, where minimal interaction data affects recommendation quality and efficiency. Hybrid models that combine both techniques are often employed in successful applications to address these weaknesses (Cano, 2017).

CONCLUSION

In summary, the research underscores the pivotal role of an optimized IT Tool Stack in enhancing collaborative capabilities, productivity and ensure high-quality outputs, particularly in the context of SMEs engaged in collaborative product development projects. The dynamic and complex nature of modern business environments necessitates the integration of sophisticated IT tools, tailored to specific project requirements and organizational goals. The integration of AI technologies further complicates this task, necessitating specialized expertise and tailored solutions. Therefore, here we proposed a framework that leverages standardized evaluation methods, a combination of recommendation systems to utilize collective industry experiences to optimize IT tools effectively.

The development of a comprehensive evaluation framework, leveraging CTQ methodology and ECBBs, provides a structured approach to assess and recommend IT tools. This framework addresses the multi-criteria nature of tool selection, ensuring alignment with project-specific KPIs and quality metrics.

The implementation of a recommendation system, utilizing ECBB, requirements engineering, collaborative and content-based filtering techniques, enhances the ability to provide personalized and effective IT tool recommendations. Combining the two filtering techniques for a hybrid recommendation system, addresses potential cold start, data sparsity and AI technology challenges, enhancing recommendation relevance.By employing a two-phased approach—manual data analysis followed by ML recommendations—we aim to create increasingly specific recommendations. Our goal is to develop consecutive recommendation systems that evolve into an extensive model employed on an online platform. This platform will utilize ML models that improve over time, allowing businesses to input their basic data and monitor their progress via a dashboard at any time.

Implementing our framework can significantly enhance collaborative capacities, productivity, and AI adoption rates among SMEs. By systematically configuring IT Tool Stacks, businesses can better meet the demands of modern collaborative projects and maintain a competitive edge in the digital economy.

ACKNOWLEDGMENT

The topic of this paper stems from the "Metis" project. The authors would like to sincerely thank the IGF (Grant No. 22534N) for their financial support of the METIS project, which enabled this research. We also wish to express our appreciation to all those who contributed to the success of this project, including our collaborators and participants. Their invaluable insights and feedback have played a crucial role in shaping the outcomes of this research.

REFERENCES

Anderl, R. (2012), "Smart Engineering: Interdisziplinäre Produktentstehung" (Book).

Benlian, A. (2011), "Opportunities and risks of software-as-a-service: Findings from a survey of IT executives", Decis. Support Syst., Vol. 52, pp. 232–246.

Çano, E. (2017), "Hybrid Recommender Systems: A Systematic Literature Review", Intelligent Data Analysis, Vol. 21, pp. 1487–1524.

Krzysztof, K. N. O. P. (2015), "Using the CTQ tree to identify requirements of quality inspection customers" (Book).

Lee, J. (2018), "Industrial Artificial Intelligence for industry 4.0-based manufacturing systems", Manufacturing Letters.

Mauerhoefer, T. (2017), "The Impact of Information Technology on New Product Development Performance", Journal of Product Innovation Management, Vol. 34, No. 6, pp. 719–738.

Peng, D. X. (2014), "Collaborative Product Development: The Effect of Project Complexity on the Use of Information Technology Tools and New Product Development Practices", Production and Operations Management, Vol. 23, No. 8, pp. 1421–1438.

Prasanna, R. (2019), "Sustainability of SMEs in the Competition: A Systemic Review on Technological Challenges and SME Performance", Journal of Open Innovation: Technology, Market, and Complexity, Vol. 5, No. 4, p. 100.

Tucker, M. (2019), "The Influence of Collaborative Information Technology Tool Usage on NPD".

Zhou, G. (2019), "The effect of artificial intelligence on China's labor market", China Economic Journal, Vol. 13, pp. 24–41.