Improving Students' Ability to Write Requirements by Teaching Goal Modeling

Shaza Elbishbishy[1], Jeshwitha Jesus Raja[1], Jennifer Brings[2] and Marian Daun[1]

[1]*Technical University of Applied Sciences Würzburg-Schweinfurt, Schweinfurt, Germany*
[2]*Bingen Technical University of Applied Sciences, Bingen, Germany*

Abstract

Software engineering and requirements engineering are important aspects of the education of robotics students as software plays a vital role for the development of modern robotic systems. However, robotics students lack the background provided in computer science and software engineering curricula. As one consequence, students often lack the ability to describe robotic systems at the level of detail required for a proper requirements specification. We propose teaching goal modeling to improve students abstract thinking capabilities and improve the understanding of system details by abstraction. This paper investigates the use of goal modeling as a learning instrument in requirements engineering education for robotics students. The study focuses on how students write natural language requirements before and after constructing a goal model as part of a course assignment. The findings indicate that goal modeling supports the identification and inclusion of additional detail, such as measurable constraints, role differentiation, and operational context conditions such as the workspace organization where a robot will be deployed.

Keywords

Requirements Engineering, Goal Modeling, Robotics, Requirements Engineering Education

1. Introduction

An increasing number of systems depend on software as their primary source of innovation. As a result, software is now developed not only by software engineers, but often also by technical engineers. In our case, we focus on educating robotics students, who need a solid understanding of software development but frequently lack essential foundations typically provided in a computer science curriculum [1]. This gap is particularly evident in the area of requirements engineering [2]. Contemporary robotic systems are both highly complex and safety-critical, which further elevates the importance of requirements engineering practices. Yet, many students do not adequately grasp the importance of requirements engineering, nor are they sufficiently able to describe technical robotic systems with the level of detail required for a comprehensive requirements specification.

In this paper, we explore how goal modeling can be used to enhance students' ability to write requirements. The explicit decomposition used in goal models and their way to connect different aspects of the system, shall help students gain an understanding of factors that are important when documenting requirements. Therefore, we first introduced the major topics of requirements engineering as suggested by IREB [3], and then taught goal modeling with i* [4, 5] and GRL [6]. Students (a) specified requirements for a case system, (b) modeled the case system using goal models, and (c) refined their natural language requirements based on their understanding gained through goal modeling.

The remainder of this paper is structured as follows. Section 2 presents related work. Section 3 describes the case study context and procedure. Section 4 presents the case study results, while Section 5 discusses the findings. Finally, Section 6 concludes the paper.

Joint Proceedings of REFSQ-2026 Workshops, Doctoral Symposium, Posters Tools Track, and Education and Training Track. Co-located with REFSQ 2026. Poznan, Poland, March 23-26, 2026

shaza.elbishbishy@study.thws.de (S. Elbishbishy); jeshwitha.jesusraja@study.thws.de (J. Jesus Raja); j.brings@th-bingen.de (J. Brings); marian.daun@thws.de (M. Daun)
0009-0002-0975-272X (S. Elbishbishy); 0009-0008-7886-7081 (J. Jesus Raja); 0000-0002-2918-5008 (J. Brings); 0000-0002-9156-9731 (M. Daun)

2. Related Work

In requirements engineering education, classroom studies have been used to explore how modeling techniques influence requirements understanding and documentation practices [7]. In this paper, we investigate the effect teaching goal modeling has on students' ability to write technical requirements. This work is informed by prior research on natural language requirements specification, goal modeling, and traceability between natural language requirements and goals [8, 9].

Requirements engineering has been studied extensively as a research and practice area [10, 11]. Natural language remains a widely used medium for requirements documentation. Existing work has examined ambiguity, linguistic issues, and writing patterns in natural language requirements [? ? ? 12]. These studies highlight challenges associated with informal requirements descriptions.

Goal-oriented requirements engineering (GORE) approaches represent system objectives explicitly and refine them into requirements [13, 14, 15]. Prominent goal modeling frameworks include KAOS, i*, and Tropos [13, 4, 16]. Traceability between early artifacts and requirements has been investigated as a requirements engineering problem [17]. Standards such as IEEE 830 and ISO/IEC/IEEE 29148 describe recommended structures and information items for requirements specifications [18, 19].

In requirements engineering education, studies often emphasize exposing students to industrial challenges and structured documentation practices [7]. In contrast, this study focuses on robotics students with a strong technical background in robotic systems but limited formal training in requirements specification.

Goal modeling has also been explored in educational contexts. Dalpiaz [20] reports on teaching goal modeling in undergraduate education and highlights its value for structured reasoning. Amyot [21] describes experience teaching GRL and emphasizes modeling as a learning instrument. More recently, Li et al. [22] analyzed challenges students face when learning iStar, identifying abstraction level and model complexity as key difficulties.

While prior studies primarily focus on computer science students, our study examines robotics students. Robotics students typically possess strong technical system knowledge but limited exposure to formal requirements engineering methods. This difference in educational background provides an additional perspective on the pedagogical role of goal modeling.

In addition, to educational approaches on goal modeling, recent research on goal modeling for robotics has also confirmed the applicability of goal modeling practices to the development of robotic systems, highlighting the value of educating robotics students in this area [23].

3. Study Design

This section describes the goals of the study, the course setting in which it was conducted, the case system used as a study subject, and the procedure followed during the study.

3.1. Goal

The goal of this study is to examine the role of goal modeling as a learning instrument in requirements engineering education for robotics students. The study focuses on how constructing a goal model influences the way students understand and document system requirements when working with natural language specifications.

Specifically, the study investigates how students express system objectives, constraints, and responsibilities in natural language requirements before and after constructing a goal model as part of a course assignment. The intention is to observe changes in requirements documentation that occur as students apply goal modeling concepts introduced during the course. In particular, we investigate two research questions:

- **RQ1:** How does goal modeling influence the way system objectives are expressed in natural language requirements?

- **RQ2:** How does goal modeling influence the level of detail and explicitness of natural language requirements?

3.2. Course Setting

The study was conducted as part of a university-level course on requirements engineering within a seven semester robotics bachelor program.[1] The course was offered in the sixth semester and introduces the field of requirements engineering to robotics students without computer science background. Requirements specification as well as goal modeling were taught as part of the curriculum and applied by the students in the context of the case study.

A total of 18 students participated in the course. Students were given approximately 90 minutes for the initial natural language specification, two 90-minute sessions for constructing the goal model, and one 90-minute session for revising the requirements specification.

Informal feedback collected during the course indicated that students initially struggled with distinguishing goals from requirements. However, after completing the modeling exercise, students reported that the graphical representation helped them identify missing aspects of the system, particularly related to safety constraints and actor responsibilities.

3.3. Case System

The case system used in the study is a collaborative toy car production system. The system is designed to support the assembly of toy cars through collaboration between a human operator and a collaborative robot operating in a shared workspace.

The system involves multiple interacting elements, including:

- manipulation tasks such as picking, placing, holding, and fastening parts,
- coordination between human and robot activities,
- workspace organization with human, robot, and collaborative zones,
- safety-related constraints such as distance monitoring, speed regulation, collision avoidance, and emergency stopping,
- organizational requirements such as operator training and periodic maintenance.

The case system was selected because it involves multiple objectives, actors, and constraints, making it suitable for goal modeling and requirements analysis.

3.4. Procedure

The study was conducted as a structured classroom intervention. The professor first provided the entire class with the case description of the collaborative toy car production system. Students were instructed to produce a natural language requirements specification based solely on the textual description.

During this phase, students worked without the use of modeling techniques. The resulting specification was characterized by long textual descriptions, overlapping statements, and ambiguities regarding actor responsibilities and operational constraints.

After completion of the first specification, goal modeling using i* and GRL was introduced. Students constructed a goal model representing the system objectives, actor responsibilities, safety constraints, and workspace structure. The model decomposed the overall objective into sub-goals associated with individual actors and operational conditions.

In the final phase, students used the constructed goal model as a reference structure to rewrite the requirements specification. They systematically examined each actor in the model and translated the associated goals and tasks into explicit natural language requirements. Each goal, task, and constraint represented in the model was reformulated as one or more precise requirement statements.

This process resulted in a revised requirements specification with a clearer separation of objectives, constraints, and actor responsibilities.

[1] Details about the degree program can be found at https://iro.thws.de

4. Case Study Results

This section presents the results of the case study based on a detailed examination of the produced artifacts. The results focus on differences between the two versions of the natural language requirements and on the relationship between the goal model and the revised requirements specification.

4.1. Natural Language Requirements

Table 1 presents representative natural language statements from the requirements specification written before goal modeling and from the revised specification written after goal modeling. In the first column, each row represents a system aspect derived from the use case. The second column shows representative formulations from the initial natural language specification. These statements typically describe system behavior at a high level without explicitly distinguishing actors, constraints, or conditions. The third column shows corresponding statements from the revised specification, which were derived after constructing the goal model.

Table 1
Representative Natural Language Statements Before and After Goal Modeling

Aspect	NL Before Goal Model	NL After Goal Model
System Objective	The system shall assemble toy cars.	The system shall support the assembly of toy cars in a human–robot collaborative workspace.
Production Target	The system shall produce toy cars efficiently.	The system shall enable the production of 1,000 toy cars per month under defined operating conditions.
Task Allocation	The robot and operator shall perform assembly tasks.	The collaborative robot (cobot) shall pick and hold parts while the human operator performs fastening tasks.
Safety Behavior	The system shall operate safely.	The robot shall slow down when the operator is within 30 cm and activate emergency stop below 5 cm.

As shown in Table 1, the revised requirements include additional information related to task allocation, quantitative thresholds, and operational conditions that are not explicitly stated in the initial specification.

4.2. Goal Model

Figure 1 shows the goal model constructed for the case study. The top-level goal represents the overall objective of assembling toy cars in a collaborative environment. This goal is decomposed into sub-goals addressing task execution, collaboration, safety constraints, workspace organization, and organizational requirements. Actor-specific goals are associated with the human operator and the cobot, reflecting their respective responsibilities within the system. The goal model also captures constraints related to system operation and safety. These constraints include distance-based speed regulation, emergency stopping conditions, and the separation of the workspace into human, robot, and collaborative zones. Each constraint is represented as a distinct element in the model and is associated with the actors and tasks to which it applies.

By representing these constraints explicitly, the goal model specifies under which conditions certain system behaviors are permitted or restricted. For example, distance-related constraints define when the cobot must reduce speed or stop operation, while workspace-related constraints define allowable movement areas for the human operator and the cobot. These constraints are linked to task-related goals, ensuring that operational and safety considerations are represented alongside functional objectives.

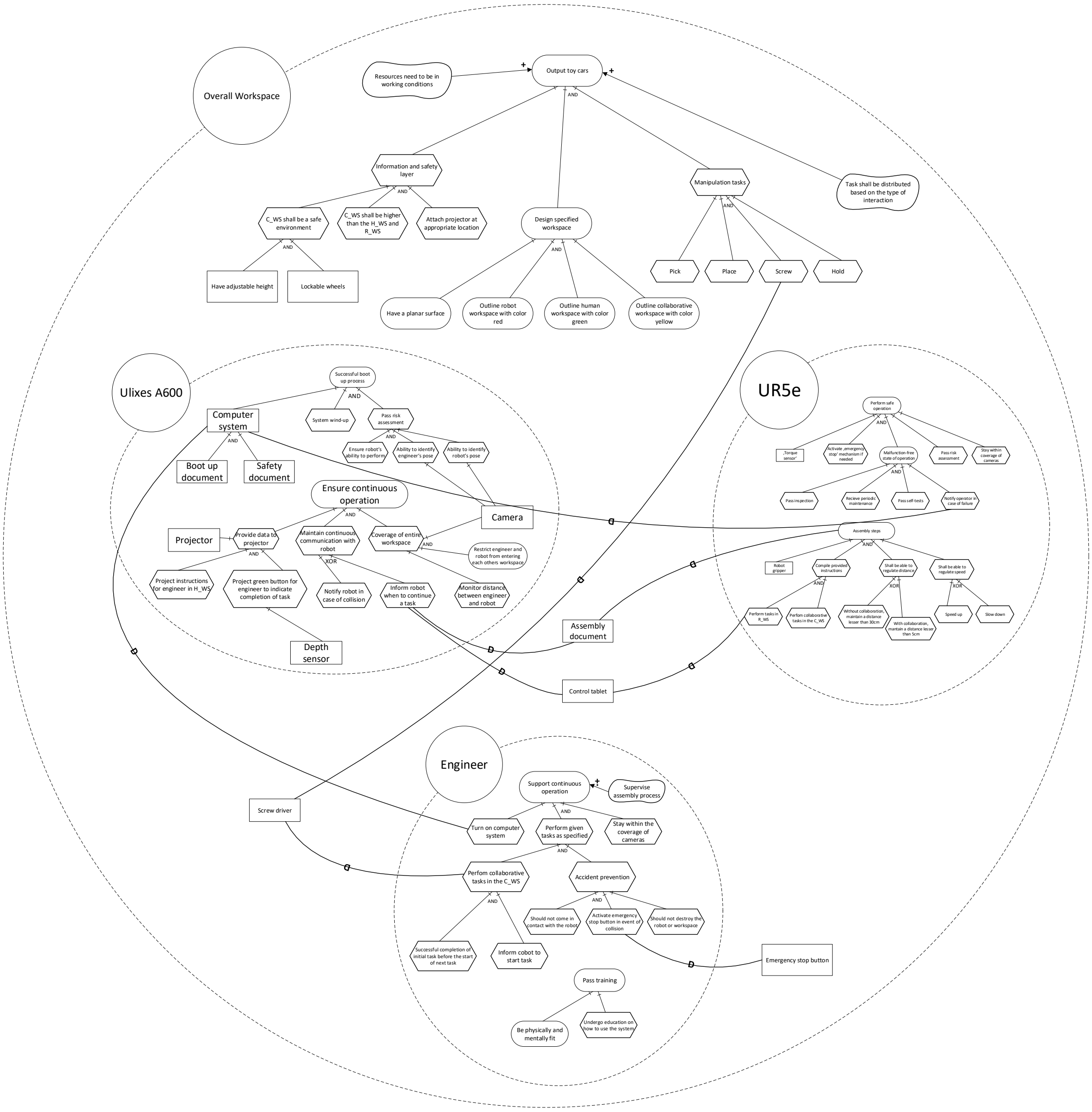

Figure 1: Goal model of the collaborative toy car production system

As a result, the goal model provides a structured representation of both objectives and constraints, enabling the identification of system properties that are not expressed as a single high-level statement but emerge from the combination of goals, actors, and conditions.

4.3. Traceability Between Goal Model and Requirements

Traceability between the goal model and the revised requirements specification was established through a systematic visual inspection process. Students used the goal model as a structuring reference and examined each actor, goal, and task node individually.

For each actor represented in the model (e.g., human operator, cobot), students reviewed the associated goals and sub-goals and reformulated them as explicit natural language requirement statements. Safety constraints and workspace-related goals were similarly translated into requirement statements describing measurable conditions or operational rules.

The traceability process did not rely on automated tooling or formal traceability matrices. Instead, the goal model functioned as a visual decomposition structure from which requirements were derived.

The mapping was therefore direct and manual: model elements were inspected and transformed into textual statements.

As a result, most refined requirements can be associated with one or more specific goal model elements. In some cases, a single requirement consolidates multiple related goal nodes. In other cases, numerical thresholds and contextual details were introduced during the refinement process while referring to the corresponding safety or operational goals in the model.

5. Discussion

The analysis shows observable differences between the initial and revised natural language requirements specifications. After goal modeling, the requirements include a broader range of system aspects, such as explicit production targets, safety-related constraints, workspace organization, and actor-specific responsibilities. These aspects are represented explicitly in the goal model and subsequently reflected in the revised requirements.

In addition, the traceability analysis demonstrates that goals and constraints captured in the model correspond to specific requirements in the revised specification. This correspondence indicates a systematic relationship between system objectives represented in the model and textual requirements.

5.1. Findings Related to RQ1

RQ1 examines how goal modeling influences the understanding and expression of system requirements. Before goal modeling, the natural language requirements describe system behavior primarily at a high level. Objectives related to collaboration, safety, and task execution are stated in general terms, and relationships between these objectives are not explicitly represented. After constructing the goal model, system objectives are decomposed into distinct goals associated with collaboration, safety, workspace structure, and actor responsibilities. These goals are reflected in the revised natural language requirements, which explicitly reference the roles of the human operator and the cobot, as well as their interactions. The traceability between goal model elements and requirements indicates that system objectives are more explicitly articulated in the revised specification.

These findings show that goal modeling supports requirements understanding by making system objectives and their relationships explicit and by providing a structured basis for expressing them in natural language.

5.2. Findings Related to RQ2

RQ2 examines how goal modeling influences the level of detail in natural language requirements. The comparison between the initial and revised requirements specifications shows that the revised version contains additional information not present in the initial specification. This includes quantitative production targets, explicit safety thresholds, defined workspace boundaries, and detailed task allocation between actors. These elements correspond to goals and constraints represented in the goal model. The revised requirements include measurable conditions and defined system responses that can be traced back to specific goals. This indicates a higher degree of specification granularity in the revised requirements.

These findings indicate that goal modeling supports the identification and inclusion of explicit constraints, conditions, and responsibilities in natural language requirements.

5.3. Threats to Validity

The findings are based on a single case study conducted by one class using one use case. As a result, the observations may be influenced by group-specific decisions and familiarity with the problem domain. The evaluation relies on qualitative artifact analysis rather than quantitative metrics. Therefore, the findings should be interpreted within the scope of the presented case study. Given the educational setting and the absence of a controlled comparison group, the findings should be interpreted as observational rather than causal.

6. Conclusion

Teaching requirements engineering to robotics students is hard. Although they already know technical details about robotic systems through their studies, it is difficult for them to define precise requirements including these technical details. In this paper, we investigted how teaching goal modeling can support students in refining their requirements and developing an understanding for the attention to detail needed for proper requirements specification. Therefore, this paper presented a classroom case study examining the role of goal modeling in structuring natural language requirements. The study focused on a collaborative toy car production system and analyzed artifacts produced by a class of robotics students, including an initial natural language requirements specification, a goal model, and a revised requirements specification derived from the goal model.

The results show observable differences between the natural language requirements written before and after goal modeling. After constructing the goal model, the revised requirements explicitly represent system objectives, actor responsibilities, constraints, and operational conditions that were only implicitly or partially described in the initial specification. The traceability analysis demonstrates a systematic relationship between goal model elements and requirements in the revised specification. By addressing the research questions, the study illustrates how goal modeling influences the expression of system objectives in natural language requirements and how it supports the inclusion of additional detail, such as measurable constraints, role differentiation, and workspace organization. The findings indicate that goal models can serve as an intermediate artifact that structures reasoning about system objectives and supports the derivation of textual requirements.

As the study is limited to a single use case and an educational setting, future work may involve applying the same procedure to multiple groups, additional systems, or industrial contexts, as well as incorporating quantitative analyses to complement artifact-based observations.

Declaration on Generative AI

During the preparation of this work, the authors employed the following AI tools: *TexGPT* to support paraphrasing and grammar checks. All content was then reviewed and edited by the authors, who take full responsibility for the final version of this work.

References

[1] M. Daun, Software engineering education for technical engineering degrees: A comparison with the needs of robotics software engineering education, IEEE Software 40 (2023) 40–46.
[2] M. Manjunath, J. Jesus Raja, R. Herrler, M. Daun, Robotics software engineering education: An experience report, in: 2025 IEEE/ACM 37th International Conference on Software Engineering Education and Training (CSEE&T), IEEE, 2025, pp. 336–346.
[3] M. Glinz, Standard Glossary for the Certified Professional for Requirements Engineering (CPRE) Studies and Exam v2.0.0, Technical Report, Int. Requirements Engineering Board e.V., 2020.
[4] E. S. K. Yu, Modelling strategic relationships for process reengineering, IEEE Expert (1995).
[5] F. Dalpiaz, X. Franch, J. Horkoff, istar 2.0 language guide, arXiv preprint arXiv:1605.07767 (2016).
[6] ITU International Telecommunication Union, Recommendation ITU-T Z.151: User Requirements Notation (URN), Technical Report, 2018.
[7] M. Daun, A. M. Grubb, V. Stenkova, B. Tenbergen, A systematic literature review of requirements engineering education, Requirements Engineering 28 (2023) 145–175.
[8] J. Jesus Raja, S. Elbishbishy, J. Brings, M. Daun, Teaching requirements engineering to robotics students through reverse engineering: An experience report., in: REFSQ Workshops, 2025.

[9] M. Manjunath, J. Jesus Raja, M. Daun, How teaching conceptual modeling to robotics students changes their perception of software engineering., ER (Companion) 2025 (2023).
[10] B. Nuseibeh, S. Easterbrook, Requirements engineering: A roadmap, in: Proceedings of the Conference on the Future of Software Engineering (ICSE FOSE), 2000, pp. 35–46.
[11] P. Zave, M. Jackson, Four dark corners of requirements engineering, ACM Transactions on Software Engineering and Methodology 6 (1997) 1–30.
[12] H. Femmer, D. M. Fernández, S. Wagner, S. Eder, Rapid quality assurance with requirements smells, Journal of Systems and Software 123 (2017) 190–213.
[13] A. Dardenne, A. van Lamsweerde, S. Fickas, Goal-directed requirements acquisition, Science of Computer Programming 20 (1993) 3–50.
[14] A. van Lamsweerde, Goal-oriented requirements engineering: A guided tour, in: Proceedings of the 5th IEEE International Symposium on Requirements Engineering (RE'01), 2001.
[15] A. Lapouchnian, Goal-Oriented Requirements Engineering: An Overview of the Current Research, Technical Report, University of Toronto, 2005.
[16] P. Bresciani, P. Giorgini, F. Giunchiglia, J. Mylopoulos, A. Perini, Tropos: An agent-oriented software development methodology, Autonomous Agents and Multi-Agent Systems (2004).
[17] O. Gotel, A. Finkelstein, An analysis of the requirements traceability problem, in: Proceedings of the 1st IEEE International Conference on Requirements Engineering, 1994.
[18] Ieee std 830-1998: Ieee recommended practice for software requirements specifications, 1998.
[19] Iso/iec/ieee 29148:2011 systems and software engineering — life cycle processes — requirements engineering, 2011.
[20] F. Dalpiaz, Teaching goal modeling in undergraduate education, in: iStarT@CAiSE, 2015.
[21] D. Amyot, Goal modeling education with grl: Experience report, in: iStar 2015, 2015.
[22] T. Li, Q. Zhou, et al., Understanding the challenges and requirements for facilitating istar learning, Information and Software Technology (2025).
[23] M. Daun, M. Manjunath, J. Jesus Raja, Safety analysis of human robot collaborations with grl goal models, in: International Conference on Conceptual Modeling, Springer, 2023, pp. 317–333.