

UNIVERSITY OF TARTU
Faculty of Science and Technology
Institute of Computer Science
Software Engineering Curriculum

Rasul Agharzayev

Conflicts Management in Goal-Oriented Requirements Engineering: Socio-Technical Systems Perspective

Master's Thesis (30 ECTS)

Supervisor(s): Ishaya Peni Gambo, PhD

Tartu 2022

Conflicts Management in Goal-Oriented Requirements Engineering: Socio-Technical Systems Perspective.

Abstract:

Goal Oriented Requirements Engineering (GORE) is a fundamental methodology for developing high-quality software systems. However, addressing conflicts in GORE to make the development process of Socio-technical systems (STSs) easier in meeting the business goals, especially in an agile development methodology, is challenging. This thesis focused on conflict management in GORE from the STS perspective. The goal is to explain goal formulation and how conflicts can be identified and resolved in the user stories related to the specified goal(s). The thesis presents a case study where functional goals are developed in an agile development methodology. We followed software engineering design principles to build a software tool that captures the goals and users' stories and then identifies and resolves conflicts in the user stories of those goals using the analytical decision-making technique. The result provides a general overview of the problem domain, which shows the effects of the user and business sides' motivations during conflict management and, as a result, presents the resolved user stories for each goal. Our results provide both analytical and theoretical explanations. It is concluded that considering human and company or organization motivation in a development project is one of the key activities for conflict management because it makes it easier to decide between conflicting user stories. At the same time, human roles (team) are one of the critical points to gaining conflict-free requirements.

Keywords: Goal oriented requirements engineering, conflict management, socio-technical systems, conflict identification, conflict resolution, analytical hierarchy process, agile methodology, user stories

CERCS: P170 Computer science, numerical analysis, systems, control

Konfliktide juhtimine eesmärgipõhises nõuete kavandamises: Sotsiaaltehniliste süsteemide vaatenurk.

Lühikokkuvõte:

Eesmärgile Orienteeritud Nõuete Tehnika (GONT) on kvaliteetsete tarkvarasüsteemide arendamise põhimetoodika. Kuid konfliktide lahendamine GONT-s, et muuta sotsiaaltehniliste süsteemide (STS) arendusprotsess ärieesmärkide täitmisel lihtsamaks, eriti agiilse arendusmetoodika puhul, on keeruline. See lõputöö keskendus konfliktide juhtimisele GONT-s STS-i vaatenurgast. Eesmärk on selgitada eesmärgi sõnastamist ja seda, kuidas on võimalik tuvastada ja lahendada määratud eesmärgi(te)ga seotud kasutajalugudes konflikte. Lõputöö esitab juhtumiuuringu, kus funktsionaalsed eesmärgid töötatakse välja agiilse arendusmetoodikas. Järgisime tarkvaratehnilise disaini põhimõtteid, et luua tarkvaratööriist, mis jäädvustab eesmärgid ja kasutajate lood ning seejärel tuvastab ja

lahendab nende eesmärkide kasutajalugude konfliktid, kasutades analüütilist otsustus-tehnikat. Tulemus annab üldise ülevaate probleemvaldkonnast, mis näitab kasutaja ja äripoolte motivatsiooni mõju konfliktide lahendamisel ning sellest tulenevalt esitab iga eesmärgi puhul lahendatud kasutajalood. Meie tulemused annavad nii analüütilisi kui ka teoreetilisi selgitusi. Järeldatakse, et inimese ja ettevõtte või organisatsiooni motivatsiooni arvestamine arendusprojektis on konfliktijuhtimise üks võtmetegevusi, kuna see muudab vastuoluliste kasutajalugude vahel otsustamise lihtsamaks. Samas on inimrollid (meeskond) üks kriitilisi punkte konfliktitasunõuete saavutamisel.

Võtmesõnad: Eesmärgile orienteeritud nõuete tehnika, konfliktide juhtimine, sotsiaaltehnilised süsteemid, konflikti tuvastamine, konflikti lahendamine, analüütiline hierarhia protsess, agiilne meetodika, kasutajate lood

CERCS: P170 Arvutiteadus, arvutusmeetodid, süsteemid, juhtimine (automaatjuhtimisteooria)

Contents

1	Introduction	8
1.1	Research Motivation	9
1.2	Research Goal	9
1.3	Research Contribution	9
1.4	Outline	10
2	State of the Art	11
2.1	Requirements Engineering	11
2.2	Goal Oriented Requirements Engineering	11
2.3	Socio-Technical System	12
2.4	Conflict Management	12
2.4.1	Identification of conflicts	13
2.4.2	Resolution of conflicts	13
2.5	Agile Methodology	14
2.5.1	Project manager role in Agile	15
2.6	User Stories	15
2.7	Related Works	15
3	Methodology	17
3.1	Research Approach	17
3.2	The Case Study - Social Media for Developers	18
3.3	Approach to Answering Research Questions	19
3.3.1	Approach for RQ1	19
3.3.2	Approach for RQ2	20
3.3.3	Approach for RQ3	22
3.3.4	Approach for RQ4	25
4	Results	26
4.1	Methodology Implementation	26
4.2	Development of the Software Tool	26
4.3	Development of the Functional Goals and User Stories	26
4.4	Conflict Identification Flow Implementation	32
4.5	Conflict Resolution Flow Implementation	34
5	Discussions	37
5.1	Formation of the Functional Goals	37
5.2	Conflict Management	38
5.3	Validation of the Findings	38

6 Threat to Validity	39
7 Conclusion and Future Work	40
References	46
Appendix	51
Lisad	52
I. Glossary	52
II. Licence	53

List of Figures

1	Main flow of Conflict management	17
2	User story collection flow	18
3	Overview of Conflict identification part	20
4	Database tables: User stories and Functional goals	21
5	Code snapshot which made conflict identification	22
6	Part of UI as a first step of Conflict Identification	32
7	Part of UI to select goal for identification	32
8	Part of UI as a result of Conflict Identification	33
9	Part of UI for entering weights which gained from PWC	35
10	Part of UI as a result of Conflict resolution	36
11	Functional goal update popup	37
12	Selected user story for notification flow	48
13	Selected user story for post sharing flow	50
14	Modal for creating functional goal in the UI	51
15	Modal for creating user story in the UI	51

List of Tables

1	Names and values for motivation levels.	19
2	Importance level names and values	22
3	Sample for AHP matrices	23
4	AHP decision making process	24
5	User stories for Log In flow	27
6	Sample for AHP metrics for Login user stories	34
7	User stories for notification flow	47
8	User stories for sharing a post flow	48
9	List of the acronyms in the research	52

1 Introduction

Conflicts management has received a lot of attention in terms of developing effective resolution plans in management science, international relations, psychology, and design science [1]. Curiously, stakeholders' mutual understanding and satisfaction have always been of primary importance in these fields, and negotiation is the preferred method of resolving disputes or conflicts [2]. In the software engineering practice, the requirements engineering (RE) process is the most suitable phase for managing these conflicts amidst different stakeholders' views, opinions, desires, expectations, and goals [3]. Stakeholders, in this sense, can either be the people, groups, or organizations that influence or are influenced by a system [4].

However, using the proper RE methodology is essential for creating high-quality software-intensive systems. In goal-oriented requirements engineering (GORE), the stakeholders' statements about the intended system are expressed as goals that the system must meet [5]. These goals represent the needs and intentions of the different stakeholders, which can further be simplified in the form of a user story to achieve the specified goal. The user story is a better way to capture the stakeholders' requirements for a specific goal, especially in the agile development methodology. In sociotechnical systems (STS), the goals are accomplished through the mutual collaboration of the different stakeholders involved. Remarkably, addressing conflicts in GORE can make the development process of STSs easier in many areas, as they (STSs) are designed to meet business goals [6]. In particular, as stakeholders frequently pursue mismatching goals, conflicts in requirements must be identified and resolved. This is a necessary component of GORE [7] [8] [9].

In this thesis, we noted that the rapid improvement of agile software development methodologies and practices is prone to conflicts, especially as different stakeholders' goals and expectations need to be mutually harmonized [10]. Thus, the new developments and application of the agile methods in practice must be considered for conflict management, and requirements must be improved [11]. Notably, the agile methodology is one of the best and most popular methodologies for achieving high-quality software applications [12] and for getting benefits, especially in team collaboration and meetings [13].

Therefore, this thesis's primary research question (RQ) is: **How to manage conflicts within an agile development framework in GORE from STSs perspective in an actual software development project?** To identify the problem and to answer the above question, we use the following three sub research questions:

- RQ1: How are functional goals created or defined in an agile development project?
- RQ2: How to identify conflicts in the user stories specified for the functional goals?

- RQ3: How to resolve the conflicts identified in RQ2?
- RQ4: How to validate the approach in RQ2 and RQ3?

1.1 Research Motivation

Conflict management aims to get conflict-free requirements which is one of the integral parts of delivering a high-quality software system [3]. The primary motivation for this thesis is that it is based on real practices and applies to most software development companies or organizations. At the same time, we seek to introduce a more cost-effective approach to conflict management than existing approaches.

1.2 Research Goal

The main focus of this study is to provide a more realistic strategy for identifying and resolving conflicts in the GORE of STS. The practical approach can best be used in the industry where the agile development methodology is practiced.

Therefore, the research goal of this study is twofold: Firstly, we aim to give a deeper explanation of how to formulate functional goals in the agile development methodology from an STSs perspective and capture related user stories for the specified goal(s). Secondly, we aim to address conflicts by identifying and resolving them using a suitable and more reliable software tool that can be useful for practitioners in the industry and researchers in the RE community. To achieve the research goals, we presented a real case study further to prove the adequacy of our approach and tool.

1.3 Research Contribution

Our contributions in this thesis are summarized as follows:

- **Novel method:** Under the umbrella of the agile methodology, we developed a software tool for the conflict management process. Regarding that, we used the analytical decision-making technique to calculate the relative importance weights of requirements in the resolution process to aid mutual satisfaction and acceptance by stakeholders (end-users).
- **Novel insights:** Current state-of-the-art techniques are mainly theoretical, which the industry finds difficult and expensive to use and implement in real-life development projects. Regarding that, we highlight real software engineering practices and demonstrate suitable and more realistic approaches that can be applied in real industry development projects.

- **Improved performance:** We are introducing an approach that considers the company or organization's goal and end-users desires and presents the most relevant results with the help of the analytical decision-making method.

1.4 Outline

The thesis has been structured as follows:

- **Chapter 2** reviews the state-of-the-art about requirements engineering, GORE, STS, conflict management, agile methodology, and user story. Moreover, we also present related work and uncover the existing gaps.
- **Chapter 3** describes the research approach and the case study used for the research and provides detailed information about the methods for answering each research question.
- **Chapter 4** presents the results after implementing the approaches in Chapter 3.
- **Chapter 5** discusses the findings and implications of the research based on the results in chapter 4.
- **Chapter 6** discusses threats to the validity of the work.
- **Chapter 7** presents the conclusion and future work.

2 State of the Art

This chapter examines the fundamental concepts of this thesis, such as RE, GORE, STS, conflict management, agile methodology, user stories, and an overview of recent research that we believe is pertinent to this subject. We begin by demonstrating a basic understanding of RE, concentrating on the essential activities and processes and how conflicts develop. Next, we explain GORE, where we describe the concept of “goals” in RE and goal orientation in general. We also define the STS perspective in GORE before introducing the concept of conflict management, explicitly identifying and resolving conflicts in GORE within the scope of STS. We then review previous research on conflict management in GORE for STS to identify their strengths and limitations.

2.1 Requirements Engineering

Requirements engineering (RE) is commonly acknowledged as the first stage of the software engineering process. It is a critical activity that engenders quality development of software-intensive systems in software engineering practices [14]. Remarkably, RE is a well-established field that incorporates a wide range of skills, procedures, methodologies, techniques, and instruments. Hübner et al. [15] define it as "a component of software engineering that integrates several methodologies for obtaining, describing, analysing, maintaining, and utilising user needs."

In this context, the user needs are referred to as requirements, which can be considered goals in GORE. The requirements are the feature that a system must exhibit to solve a problem. Ernst [16] observed that these requirements are the desirable characteristics of a new system that the specification must achieve. In particular, they are the goals that the stakeholder wants to see accomplished for them to embrace the system.

2.2 Goal Oriented Requirements Engineering

These days implementing GORE approaches is prevalent in the software engineering process [17]. At various levels of abstraction, goals take the different objectives the system under consideration should arrive at [18]. GORE is concerned with using goals for obtaining, elaborating, developing, identifying, investigating, arranging, documenting, and revising requirements. Shortly, GORE is mainly focused on the problem-solving side. During the creation of the goals, 'why' is always asked and tried to identify 'what' is at the root and why this goal must be created. In GORE, plenty of various methods are used during the development of the goals. Some of these methods are as follows:

- **Knowledge Acquisition in automated Specification (KAOS):** This is one of the formal ways to develop goals, and this methodology consists of three main steps. First of all, creating the specific language (like the agent, action, constraint, goal,

etc.), secondly elaborating goals from goal to requirements, and another process guiding decisions during that elaboration [19].

- **The Goal Based Requirements Analysis Method (GBRAM):** This method was created because there was a gap in goal identification, and this method aims to solve that gap.

2.3 Socio-Technical System

The term STS was first used in the 1950s [20]. STS is a system that is a combination of social, human, organisational, and technical approaches [21]. It can be seen like humans (agents) are making tasks and different processes with machines with the help of the software for customers. In other words, it can be seen as a collaboration work among humans and/or software and/or hardware [22]. STS is a general concept and has plenty of applications in the industry. If the aviation industry is an example, the first contemporary aeroplane can be an excellent example of STS in this industry [23]. In the implementation of the STS in RE and GORE, requirements and goals must be designed based on all social, behavioural, and technical aspects. One of the ways to implement this is by using a viewpoint framework approach which is creating a 3x3 matrix with problem domain analysis, implementation, and design rows and symbolising the perspectives, information, and behaviour columns [24]. The viewpoint framework was explained by Sterling and Taveter [24]. Interestingly, successful implementation of STS will require adequate conflict management considering its collaborative nature where humans are involved, and their different mismatching goals need to be harmonised to guarantee stakeholders' satisfaction.

2.4 Conflict Management

Conflict management entails identifying and resolving conflicts arising due to the divergence of views and goals in an organisation [25]. Technically, it is crucial in the present digital world and the information-intensive system, where complex multi-stakeholder design problems for STSs are inevitable [3]. On the one hand, managing conflicts is expected in the GORE process of STSs, since both (GORE and STSs) are collaborative in nature, involving many different stakeholders with mismatching goals [26, 27]. These goals significantly play a crucial role in the RE process, as they provide a clear direction when eliciting, creating, and specifying the requirements needed for achieving the goals. They influence the development process of this requirements [28, 29]. Moreover, the goals serve as a completeness criterion for the requirements specification [30]. As such, the specification is complete if all stated goals are satisfied by the specification [31].

In essence, as Van Lamsweerde et al. [30] observed, the requirements "implement" goals in a similar manner to how programs "implement" design requirements, and

the existence of these requirements are linked to the goals [32]. Therefore, conflicts impact the goal-driven process due to stakeholders' involvement, [33], and dealing with stakeholders' goals, particularly the functional and quality goals, will inevitably provide difficulties. On the other hand, managing conflicts in a software development project for a specific problem domain can be challenging. For that, requirements engineers need to suggest best practices for conflict management in the goal-driven process [30].

This thesis focuses on conflict management in the GORE process of STSs. Taking into consideration stakeholders' goals, interests, and desires is one of the critical points of GORE because it helps reach the system's overall goal earlier and also helps develop a high-quality software-intensive system [34]. At the same time, having conflicting goals and desires between stakeholders causes conflicts. Still, at the same time, it shows the weak sides of the system, and rather than avoiding or closing these kinds of conflict situations, those kinds of concerns must be found and solved [35].

When conflicts are not managed, it can result in a bad software product or system, and as such, stakeholders can lose their motivation and support for the software, and at the end of this, the software can be left unused tool [36]. Thus, managing conflicts, firstly, must be identified, and then, different kinds of resolution methods must be stated.

2.4.1 Identification of conflicts

Conflict identification is the first phase of conflict management, and various techniques exist to identify the conflicting requirements (goals). One of these techniques is negotiation [37]. In this technique, developers and owners analyse and discuss goals and try to find the conflicting goals [38]. The automation technique is also one of the ways to identify conflicts. In this method, developers try to implement special tools based on some algorithms and identify conflicts with the help of these tools. In other research, these approaches are called in different ways. For example, the methodology and technology used in [39] are the same approaches to negotiation and automation. The ontological approach is also one way to find conflicts, and as seen in its name, it uses ontology to identify inconsistencies. Three different approaches are also stated in [40] for identifying conflicts. The first is the formalisation technique which identifies conflicts based on formal characteristics. The second one is the stakeholder prioritisation technique which totally depends on the stakeholder's priority index in short, depending on their preferences. The third one is a model-based technique that creates a specific model for requirements and based on this model, the conflicts are identified.

2.4.2 Resolution of conflicts

This is the second phase of the conflict management process. After identifying the conflicting goals, some techniques must be implemented to resolve these conflicts, and various types of conflict resolution methods exist. Some of them are stated as follows:

- *Pair Wise Comparison (PCM)*: The PCM technique [41] is one of the best techniques for conflict resolution. This methodology has three main steps, which are stated in detail by Rashid et al. [42]. Firstly the contribution matrix is created; in the second step, weights are given to the requirements, and at the end, with the help of prioritisation, the conflicts are resolved.
- *Joint Application Development (JAD)*: JAD is one of the most popular techniques to resolve conflicts in the RE process. This is a methodology in which the business goals are collected during the development of the software with agile methodology [43]. From end-users to software developers and stakeholders, everyone who is part of the development or usage of the application is a resource for collecting business goals.
- *Win-Win approach*: Win-Win is one methodology used in identification and resolution processes. This methodology considers all parties' thoughts and claims, trying to give a solution that is met for all opinions [44]. That is why sometimes it is also called win-win collaboration. As seen from its name in this approach, the focus is to gain collaboration between all parties [45].

2.5 Agile Methodology

The agile methodology became increasingly popular in the software industry because of its dynamic nature of accommodating changes in the development process. Different frameworks were formed based on agile principles to make this methodology flexible and usable in various information technology (IT) industries. Some famous examples are SCRUM, Extreme Programming (XP), and Lean [46]. There are many advantages of this methodology based on [47] which are listed below:

- Continuous delivery
- Process is dividable into iterations
- Less defects
- Continuous testing
- Collaboration
- Maximum ROI

Based on this list, we can state that one of the main aims of this methodology is to deliver high-quality software. Remarkably, conflict-free software functional goals are one of the most critical steps to achieving high-quality software. Moreover, team collaboration, team-stakeholder collaboration, collaboration between different teams,

and meetings for product delivery such as kick-off and grooming are the main key points to achieving conflict-free functional goals [48].

2.5.1 Project manager role in Agile

The role of the project managers is significant in delivering high-quality software. In practice, most companies are still using this position name but based on agile methodology; it is not valid anymore. Instead, it is divided into two positions Scrum Master and Product Owner, and some frameworks use the coach keyword as an alternative to Scrum Master [49]. The Product Owner focuses on working with stakeholders and end-users and representing them. Scrum Master aims to manage internal task preparation, and task discussion [50]. Notably, collaboration is one of the key points to delivering conflict-free requirements, and Scrum Master and Product Owner are helping to create the calibration between stakeholder and team.

2.6 User Stories

The colossal application of agile methodologies such as SCRUM and Kanban frameworks make user stories increasingly popular [51]. User stories are textual explanation of the functional goals of the software system, which is readable and understandable by non-technical people. There are some different options to create user stories. Still, the most used one based on a survey conducted in [52] is called Connextra, and its structure of it is as follows [53]:

“As a < role >, I want < goal >, so that < benefit >”

The user story consists of three segments as follows [54]:

- Role-defines system user who wants to do an action in the system
- Goal-defines the action, which is the desire of the system user
- Benefit-defines the benefit for the system user after doing the goal

2.7 Related Works

Managing conflicts in RE is one of the key processes in the software development lifecycle. That is why plenty of research has already been done to address this issue from a different perspective. Several studies exist with promising approaches and methodologies to identify and solve conflicts. However, the best approach and methodology suitable for real software development practices is still missing from an industrial perspective. In

this section, we discuss all aspects of existing works in this area and the weaknesses and gaps in these existing works.

The paper [3] gives a high overview of conflict management in GORE. In [55], the researcher stated the importance of collaboration with stakeholders during solving conflicts in the RE process. In [56] and [57], the importance of stakeholders' collaboration and the need to harmonise their goals and views for the software-to-be were stated. This collaboration can be between requirements engineers and stakeholders or with the help of different tools and techniques.

Further, the importance of machine learning clustering algorithms in RE and conflicts management was clearly explained in [58]. At the same time, [59] explains how clustering can be used to get high-quality requirements, and the K-means algorithm was used by Gambo and Taveter [36] for clustering. Unfortunately, these approaches seem complex for industrial practices, especially within the agile development methodology. Differently, this thesis introduced a more realistic approach that is cost-effective and can be useful at the industrial level.

The work done in [18] only helps to model the conflicts, but conflict identification and resolution issues were not considered. However, in [30] not only modelling of conflicts but identification of conflicts also was taken into consideration. Similarly, the work in [60] stated approaches only for modelling and identifying. But [61] is stating a resolution framework that also simultaneously states ways for modelling and identifying. In [62], different methodologies are stated by the authors, but unfortunately, these methodologies are not proven in practice. The work done in [36] shows in detail with a table the weaknesses of some research, and this thesis leverage these weaknesses for a more realistic approach to managing conflicts in the GORE of STSs.

3 Methodology

This chapter explains the methodological approaches to answer each research question.

3.1 Research Approach

This thesis is based on a case study and states the research approaches for conflict management in GORE [9] with the case study. Since conflict management consists of two main parts, conflict identification and conflict resolving, we have developed a software tool as a web application for the full conflict identification flow and applied the analytical hierarchical process (AHP) [63] decision-making approach to get relative importance score. We used these scores in the prototype tool to solve the conflicts. AHP is one of the most used and popular methods created by Saaty [64] which helps to make decisions based on expressing the quantity of relative priorities for a given set of options on a ratio scale [65]. The complete overview of the research approach is described in Figure 1 below.

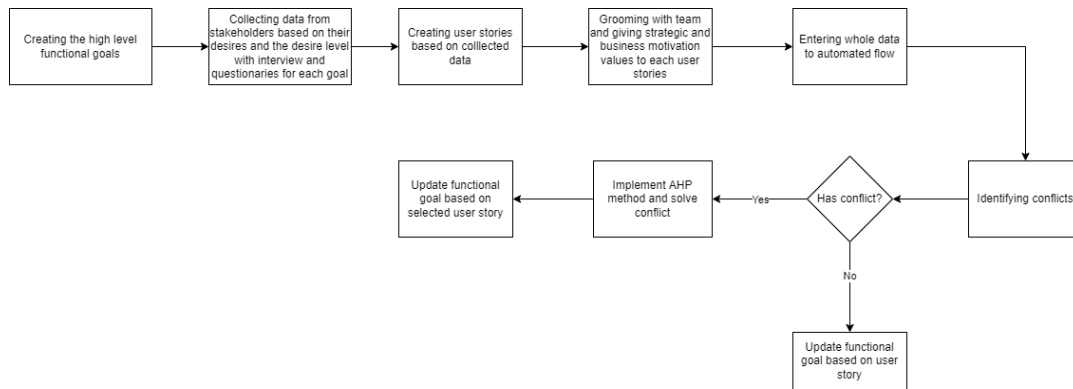


Figure 1. Main flow of Conflict management

As shown in Figure 1 firstly, we define high-level functional goals for the system, then collect information from end-users (stakeholders) about their desires and motivation for each functional goal using semi-structured interviews and questionnaires. After that, we formulate the user stories based on the data collected from the end-users. The user stories are groomed, strategic and business motivation added to each user story, and then the data passes through the conflict management flow.

3.2 The Case Study - Social Media for Developers

Our case study is based on the development of a social media platform for software developers where they can share different posts and communicate with each other via messages and calls. We consider the case scenario of building a new startup company where the primary business strategy is formed, and a high overview of functional goals is created for the software and/or system. Since it is social media, we focus on end-users desires and accept each end-user as one stakeholder. Notably, this platform is only for software developers, so our target end-user group is the software developers. For that, we made an online questionnaire and conducted face-to-face interviews with end-users (developers) and collected their opinions about each functional goal of the system, and from which perspectives would they prefer this functional goal and what is their motivation in other words, what is their level of desires for that functionality. The process is as shown in Figure 2. The values for the stakeholders' motivation are based on Table 1.

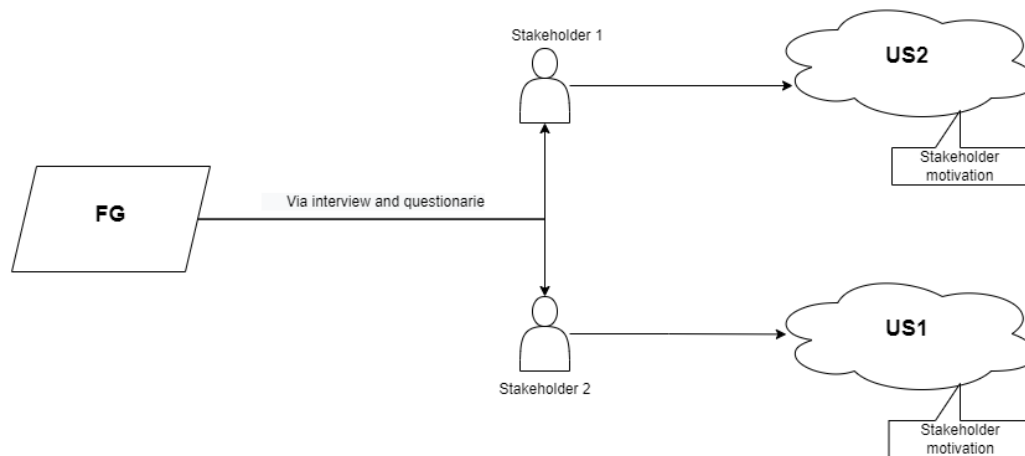


Figure 2. User story collection flow

For instance, as Figure 2 reflects, if we have one functional goal of the system like *"System must have LogIn flow,"* and based on our questionnaires and interviews, we define that for LogIn flow, we can get a number of different flow preferences from the end-users. Ultimately, the aim is the same: the user must log in to the system from different perspectives. In Figure 2, the FG stands for Functional Goal, and the US stands for the user story.

Table 1. Names and values for motivation levels.

Motivation level name	Motivation level value
Very high	5
High	4
Medium	3
Low	2
Very low	1

After collecting all data from the users, the Product Owner makes the user stories based on their responses. When all user stories are ready, the team again makes the meetings and gives two motivation levels to the user stories. The first one is the *business motivation* where the business costs and all impacts are discussed, and the motivation value is given based on values described in Table 1. Another motivation level is strategic motivation, which is the company's desire to complete each user story based on the company's strategy and values. The strategic motivation level values are also based on Table 1.

So from the flow mentioned above, the importance of team collaboration, in other words, the importance of the agile methodologies, has been clarified and justified.

3.3 Approach to Answering Research Questions

3.3.1 Approach for RQ1

To answer RQ1 (*How are functional goals created or defined in an agile development project?*), for every beginner company, firstly, the aim is to launch the Minimum Viable Product (MVP) version of the application. To do that, firstly, the strategic goals must be decided, and after that, the functional goals are created based on company strategy. But how? At that point, we leverage the agile methodologies, such as cross-functional team collaboration, internal team collaboration, and continuous delivery.

Firstly, the existing applications (mobile and web) for social media have been listed and analysed based on the question of which social media has what kind of functionalities. After analysing the existing applications and brainstorming about the features of the new product, new meetings were arranged, the strategy of the new product was decided, and the team got the company's strategic goals. After that, the goals are discussed in internal team meetings, and the necessary functional goals for the MVP are decided. Still, the functional goals can be called the high-level functional goals because after getting the inputs from the stakeholders, the functional goals will be updated to their newer and more exact versions, as explained in the user story collection flow.

So, considering all notes, our approach to get an answer for RQ1 is that use the benefits of Agile, do brainstorming or kick-off meetings, as it is in agile words, and

decide on the MVP of the product and generate necessary functional goals for the MVP.

3.3.2 Approach for RQ2

To answer RQ2 (*How are the conflicts identified in the user stories specified for the functional goals?*), we followed the agile methodology and developed a software tool to identify the conflicts. Figure 3 describes a high overview of the conflict identification process.

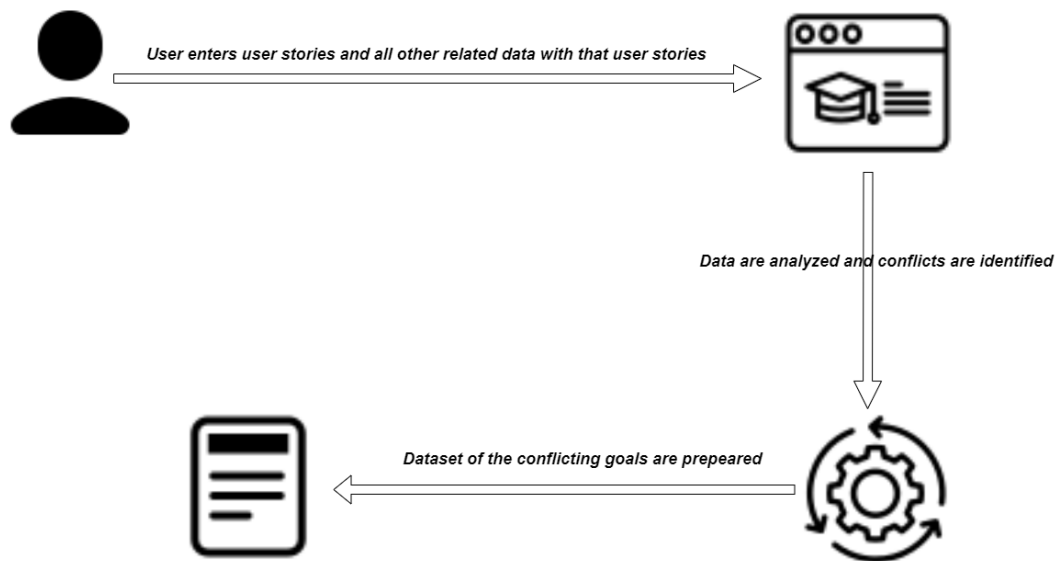


Figure 3. Overview of Conflict identification part

Based on Figure 3, firstly, the user (Scrum master in our case) enters the functional goals into the system (a developed software tool for conflict management), and after that, every user story is entered into the system. While adding every user story to the system, the user must select the related goal from the drop-down menu. The reason for that is explained in Figure 4 which presents the database table structure for user stories and functional goals. From that database structure, it is clear that each user story is joined with one functional goal, which will help to identify the conflicts. But the next question is, how can this be achieved? The following paragraph explains.

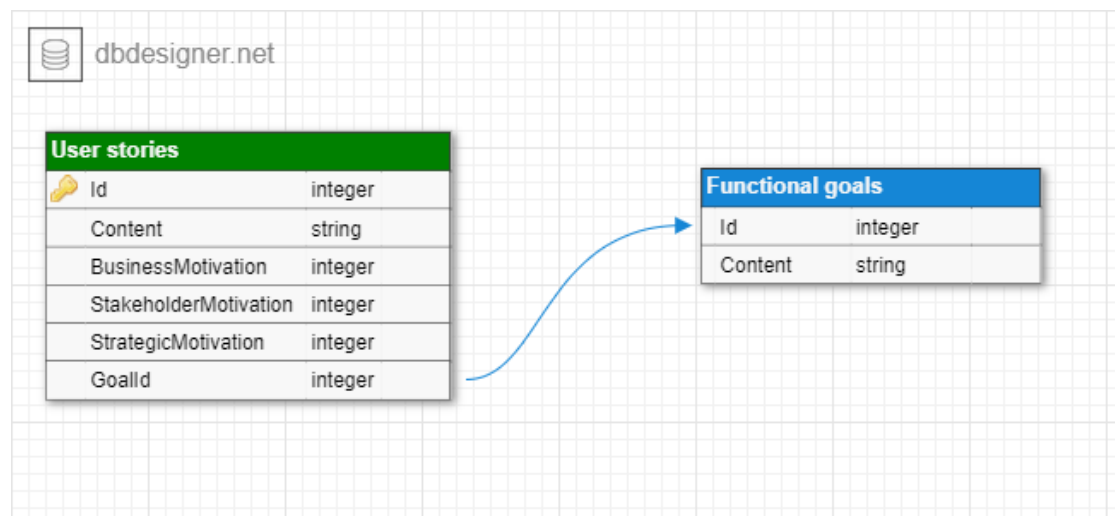


Figure 4. Database tables: User stories and Functional goals

After entering all data into the system, we enter the "/conflict-management" route of the software and select one of the functional goals from the system. When the goal has been selected, the code snippet, added as a screenshot in Figure 5 has been executed. The explanation of this code snippet written in JavaScript is as follows: "filter" is a JavaScript-specific built-in method that does a loop and conditional check operations together and returns the suitable data-set based on the condition as an array in our case array of objects (user-stories). Here, the importance of joining two tables (user stories and functional goals) is established in that code snippet. Actually, the condition is checking the case where the Id of the selected (which has been selected from the drop-down) functional goal is in any object (userStory) inside the array (userStories).

Therefore, considering all the processes above in this section, we can say that the conflict identification process is based on the collaboration of human and machine calculation which justifies our notion of STSs. In this case, the human enters the data into the system, and the system identifies conflicts and shows them to the user.



Figure 5. Code snapshot which made conflict identification

3.3.3 Approach for RQ3

To answer RQ3 (*How to resolve the conflicts which are identified in RQ2?*), we have 2 phases for that process, and both phases are defined in the following paragraphs.

In the first phase, firstly, we created the AHP metrics for user story motivations (stakeholder motivation, business motivation, and strategic motivation) as seen in Table 3. In Table 3, Mot1, Mot2, and Mot3 are stakeholder motivation, business motivation, and strategic motivation accordingly, and the variables with X_{nn} are the values of relative importance comparison between the two values. For example, X_{12} means how much the first criteria is more important than the second one; in that case, the value for X_{21} is 1 divided by the value of the X_{12} ; in other words, they are reciprocal numbers. The values are based Table 2. Stakeholder motivation is asked from stakeholders during the data collection for user stories. The other two, business motivation and stakeholder motivation, are decided by the team based on reviewing the content of each user story.

Table 2. Importance level names and values

Importance level name	Importance level value	Reciprocal value
Very highly preferred	7	1/7
highly preferred	5	1/5
Slightly preferred	3	1/3
Equally preferred	1	1

Table 3. Sample for AHP matrices

	Mot1	Mot2	Mot3	...	Motn
Mot1	X_{11}	X_{12}	X_{13}	\dots	X_{1n}
Mot2	X_{21}	X_{22}	X_{23}	\dots	X_{2n}
Mot3	X_{31}	X_{32}	X_{33}	\dots	X_{3n}
\dots	\dots	\dots	\dots	\dots	\dots
Motn	X_{n1}	X_{n2}	X_{n3}	\dots	X_{nn}

After creating the metrics, we calculated the relative importance scores (weights) based on these metrics. The relative importance score helps us define which motivation level and how much it influences the overall score of the user story. To find the weights, we use the linear algebra method. Firstly calculate the geometrical mean (M) of the values of each row. Geometric Mean Value is one of the Linear Algebra methods to find the average value. To calculate this value, we multiply the values, and after that, get the nth (in our case, n is equal to 3) root, and the formula is shown in Equation 1.

$$M_1 = \sqrt[n]{X_{11} * X_{12} * X_{13} * \dots * X_{1n}} \quad (1)$$

After getting the geometrical mean for all rows, we calculate each row's weight (W). To find this weight, we divide the geometric mean value of the desired row by the summation of all geometric mean values. This procedure is called normalization, and the formula is described in Equation 2. After this procedure sum of each weight is equal to the value 1, or with a percentage of 100. It is needed because all these criteria (motivations) are part of one decision.

The whole process explained above for calculating the Geometric Mean Value and Weight for each row has been described as an algorithm (See Algorithm 1).

$$W_1 = \frac{M_1}{M_1 + M_2 + M_3 + \dots + M_n} \quad (2)$$

Algorithm 1: Calculation of weight of each row

Input: Metrics as $2D$ array**Result:** List of weights

```
1 meansList  $\leftarrow$  Empty List ;
2 weightList  $\leftarrow$  Empty List ;
3 for row in array do
4   meanValue  $\leftarrow$  Geometrical Mean of row ;
5   meansList  $\leftarrow$  New meanValue ;
6 for mean in meansList do
7   weight  $\leftarrow$  mean divide by meansList.sum() ;
8   weightList  $\leftarrow$  New weight
9 return weightList
```

As a final step, the weights of each row (in our case, the rows are stakeholder, business, and strategic motivation) are multiplied by every value of their actual value in the user story. After that, we calculated the overall score of each user story with a summation. The AHP methodology decides the row with the highest score and, in our case, the selected user story for the functional goal of the system. (See Table 4 and Algorithm 2)

But in our case, we automated that flow, and the user just entered the weight of each row. After that, our software tool calculates all these and, in the end, shows the selected user story as a popup alert in the user interface (UI).

Table 4. AHP decision making process

	Mot1	Mot2	Mot3	...	Motn	Sum
Userstory1	$X_{11} * W_1$	$X_{12} * W_1$	$X_{13} * W_1$...	$X_{1n} * W_1$	Sum of values after multiplication
Userstory2	$X_{21} * W_2$	$X_{22} * W_2$	$X_{23} * W_2$...	$X_{2n} * W_2$	Sum of values after multiplication
Userstory3	$X_{31} * W_3$	$X_{32} * W_3$	$X_{33} * W_3$...	$X_{3n} * W_3$	Sum of values after multiplication
...	Sum of values after multiplication
Userstoryn	$X_{n1} * W_n$	$X_{n2} * W_n$	$X_{n3} * W_n$...	$X_{nn} * W_n$	Sum of values after multiplication

Algorithm 2: Decision making by AHP method for user stories

Input: userStories as an array of object(userStory) weights as an array of weights

Result: Decided user story

```
1 for userStory in userStories do
2   for weight in weights do
3     |  $\_userStories \leftarrow weight * userStory[motivation_n]$ 
4 for userStory in userStories do
5   |  $userStories \leftarrow$  find summation of motivations and add to each userStory
   | object as sum field;
6  $selectedUserStory \leftarrow$  Find userStory with maximum value of sum field;
7 return selectedUserStory
```

3.3.4 Approach for RQ4

We again focused on team activities to answer RQ4 (*How to validate the approach in RQ2 and RQ3?*). In the thesis, we tried to approach the whole process more practically. So, we never fully believe in the automated flow because every decision is artificial. Again, the answer consists of two parts: the first is for conflict identification, and the next is for conflict resolution.

In the conflict identification part, we mainly focused on team activities, and if the user will add every user story and goal correctly to the system, the success percentage is 99. But to ensure we have conflict-free and high-quality software, we must consider every probability, even if it is just one percent. When the user selects the functional goal in the UI, the user can see all conflicting user stories about the functional goal. The user (Scrum Master) is reading through all user stories again and checking if all these user stories are related to the selected functional goal. So the answer for validating the result for RQ2 is based on that flow.

We leverage team collaboration to validate the conflict resolution approach. As explained in the approach for answering RQ3, the user sees the selected user story at the end. During conflict resolution for the functional goal, the Scrum Master has listed all selected user stories. In the end, the team collected and reviewed all the selected user stories and checked manually whether the decisions were logical or not. The high-level functional goal is based on the selected user story if the decisions are logical and updated at the end.

4 Results

This chapter explains the results of our methodological approaches while answering each research question.

4.1 Methodology Implementation

To implement our methodology, as mentioned above also, we used human and machine collaboration as a perspective of STSs. In other words, we got our results with the help of the agile development methodology and software tool.

4.2 Development of the Software Tool

We created a web application software tool to automate the conflict management flow. In the development of the tool, the following technologies have been used:

- Front-end development: React JS
- Back-end development: Node JS Express JS
- Database: SQLite

The tool consists of three pages. The first page is for creating the user story, where the Create-Read-Update-Delete (CRUD) operations are executed. The second page is for entering the functional goal, and the last page is for conflict management. Moreover, the source file for the development is provided in the GitHub¹.

4.3 Development of the Functional Goals and User Stories

In the meetings, the team discussed the MVP and defined the main functional goals needed for the product's MVP stage. Three examples of the defined high-level functional goals (FG) are as follows:

- FG1: System must have Login flow.
- FG2: User can share a post on the system.
- FG3: User gets notifications from the system

¹<https://github.com/rasulagarzayev1/conflict-management>

After defining the functional goals, the questions for interviews and questionnaires to form user stories are discussed and decided for each functional goal. To collect data, a Google form was created. Still, people were usually not eager to fill out the form, so the interviews were made with developers simultaneously to collect their opinions.

For example, for the FG1, the end-users required 37 different ways (user stories), which are shown in Table 5. But while asking the flow from the user, their motivation level (How much they really want for the future?) also was asked, and the levels are defined in Table 1. Additionally, for the FG2 there are 13 user stories presented in Table 7, and for the FG3, there are 19 different user stories shown in Table 8. After collecting all the data from the end users, the team discussed all user stories. It gave them motivation levels as follows: Business motivation and strategic motivation, and the values of these motivations also are based on Table 1.

In Table 5 and the formulas, the following abbreviations are used: BM - Business motivation, SM - Stakeholder motivation, StM - Strategic motivation, US - User Story.

Table 5. User stories for Log In flow

Begin of Table				
Id	Content	SM	BM	StM
US1	As a user I want to enter the system with my phone number and password so I can access the whole content and functionality of the system.	3	5	3
US2	As a user I want to enter the system 2 different ways with my phone number and password or email and password so I can access the whole content and functionality of the system.	4	4	4
US3	As a user I want to enter the system 2 different ways with my phone number and password or username and password so I can access the whole content and functionality of the system.	4	5	3
US4	As a user I want to enter the system 3 different ways with my phone number and password or username and password or email and password so I can access the whole content and functionality of the system.	3	2	2
US5	As a user I want to enter the system 2 different ways with my phone number and password or with my Google account so I can access the whole content and functionality of the system.	2	3	3

Continuation of Table 5				
Id	Content	SM	BM	StM
US6	As a user I want to enter the system 3 different ways with my phone number and password or with my Google account or with my LinkedIn account so I can access the whole content and functionality of the system.	3	2	4
US7	As a user I want to enter the system 4 different ways with my phone number and password or with my Google or with my LinkedIn account or with my GitHub account so I can access the whole content and functionality of the system.	4	2	4
US8	As a user I want to enter the system 4 different ways with my phone number and password or with my Google or with my Facebook account or with my Twitter account so I can access the whole content and functionality of the system.	3	2	4
US9	As a user I want to enter the system 4 different ways with my phone number and password or with my Google or with my LinkedIn account or with my Twitter account so I can access the whole content and functionality of the system.	5	2	4
US10	As a user I want to enter the system 4 different ways with my phone number and password or with my GitHub or with my LinkedIn account or with my Twitter account so I can access the whole content and functionality of the system.	5	2	4
US11	As a user I want to enter the system 4 different ways with my phone number and password or username and password or email and password or with my Google account so I can access the whole content and functionality of the system.	4	1	3
US12	As a user I want to enter the system 5 different ways with my phone number and password or username and password or email and password or with my Google account or with my LinkedIn account so I can access the whole content and functionality of the system.	4	1	4

Continuation of Table 5				
Id	Content	SM	BM	StM
US13	As a user I want to enter the system 6 different ways with my phone number and password or username and password or email and password or with my Google account or with my LinkedIn account or with my GitHub account so I can access the whole content and functionality of the system.	5	1	5
US14	As a user I want to enter the system 6 different ways with my phone number and password or username and password or email and password or with my Google or with my Facebook account or with my Twitter account so I can access the whole content and functionality of the system.	3	2	4
US15	As a user I want to enter the system 6 different ways with my phone number and password or username and password or email and password or with my Google or with my LinkedIn account or with my Twitter account so I can access the whole content and functionality of the system.	2	1	4
US16	As a user I want to enter the system 6 different ways with my phone number and password or username and password or email and password or with my GitHub or with my LinkedIn account or with my Twitter account so I can access the whole content and functionality of the system.	5	2	5
US17	As a user I want to enter the system with my email and password so I can access the whole content and functionality of the system.	1	5	3
US18	As a user I want to enter the system 2 different ways with my email and password or username and password so I can access the whole content and functionality of the system.	2	4	3
US19	As a user I want to enter the system 2 different ways with my email and password or with my Google account so I can access the whole content and functionality of the system.	1	3	2

Continuation of Table 5				
Id	Content	SM	BM	StM
US20	As a user I want to enter the system 3 different ways with my email and password or with my Google account or with my LinkedIn account so I can access the whole content and functionality of the system.	3	2	4
US21	As a user I want to enter the system 4 different ways with my email and password or with my Google or with my LinkedIn account or with my GitHub account so I can access the whole content and functionality of the system.	5	2	4
US22	As a user I want to enter the system 4 different ways with my email and password or with my Google or with my Facebook account or with my Twitter account so I can access the whole content and functionality of the system.	3	2	3
US23	As a user I want to enter the system 4 different ways with my email and password or with my Google or with my LinkedIn account or with my Twitter account so I can access the whole content and functionality of the system.	2	2	3
US24	As a user I want to enter the system 4 different ways with my email and password or with my GitHub or with my LinkedIn account or with my Twitter account so I can access the whole content and functionality of the system.	4	2	3
US25	As a user I want to enter the system with my username and password so I can access the whole content and functionality of the system.	2	5	2
US26	As a user I want to enter the system 2 different ways with my username and password or with my Google account so I can access the whole content and functionality of the system.	3	4	3
US27	As a user I want to enter the system 3 different ways with my username and password or with my Google account or with my LinkedIn account so I can access the whole content and functionality of the system.	4	3	3

Continuation of Table 5				
Id	Content	SM	BM	StM
US28	As a user I want to enter the system 4 different ways with my username and password or with my Google or with my LinkedIn account or with my GitHub account so I can access the whole content and functionality of the system.	4	2	4
US29	As a user I want to enter the system 4 different ways with my username and password or with my Google or with my Facebook account or with my Twitter account so I can access the whole content and functionality of the system.	1	2	3
US30	As a user I want to enter the system 4 different ways with my username and password or with my Google or with my LinkedIn account or with my Twitter account so I can access the whole content and functionality of the system.	2	2	4
US31	As a user I want to enter the system 4 different ways with my username and password or with my GitHub or with my LinkedIn account or with my Twitter account so I can access the whole content and functionality of the system.	3	2	5
US32	As a user I want to enter the system with my Google account so I can access the whole content and functionality of the system.	3	5	3
US33	As a user I want to enter the system with my Facebook account so I can access the whole content and functionality of the system.	3	5	1
US34	As a user I want to enter the system with my Twitter account so I can access the whole content and functionality of the system.	2	1	1
US35	As a user I want to enter the system with my LinkedIn account so I can access the whole content and functionality of the system.	4	4	2
US36	As a user I want to enter the system with my GitHub account so I can access the whole content and functionality of the system.	3	5	2

Continuation of Table 5				
Id	Content	SM	BM	StM
US37	As a user I want to enter the system 4 different ways with my phone number and password or username and password or email and password or with my GitHub account so I can access the whole content and functionality of the system.	5	3	5
End of Table				

4.4 Conflict Identification Flow Implementation

After completing the formation of the initial data, the second step (conflict identification) part starts.

To identify the conflicts, firstly, the Scrum Master adds all data to the system (software tool for conflict management) and, after that, chooses conflict management in the UI, as shown in Figure 6.

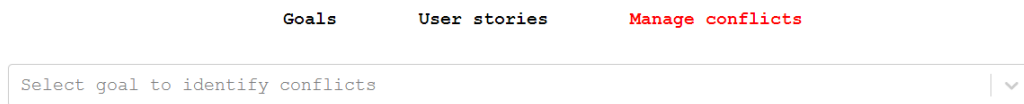


Figure 6. Part of UI as a first step of Conflict Identification

In the conflict management stage of the application, the Scrum master selects the functional goal from the drop-down, which is also shown in the same Figure 7.

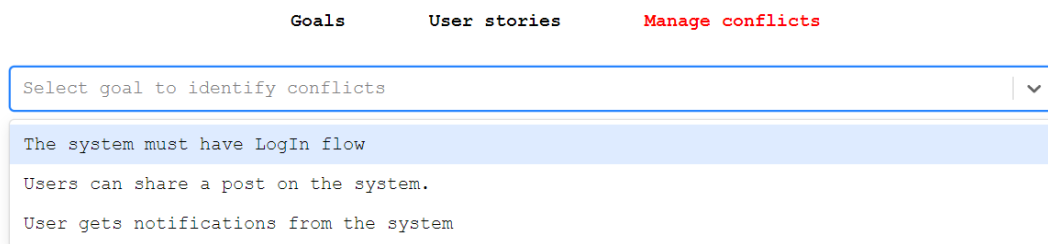


Figure 7. Part of UI to select goal for identification

When the user selects the functional goal, all user stories related to that functional goal are shown on the UI, which is also clearly seen in Figure 8.

If only one user story is related to the functional goal, the following message is shown: "There is only one user story related to the functional goal, so there is no conflict." User story also is presented under the message.

In conclusion, to implement an approach for answering RQ2, firstly, we must enter all data into the system and, after that, just follow the UI commands. With simple UI touches, we can identify the conflicting user stories.

Goals

User stories

Manage conflicts

The system must have LogIn flow

There is 37 conflicting user stories as shown in table

Enter relative importance score to solve conflicts

Content	Stakeholder Motivation	Business Motivation	Strategic Motivation
As a user I want to enter the system with my phone number and password so I can access the whole content and functionality of the system.	3	5	3
As a user I want to enter the system 2 different ways with my phone number and password or email and password so I can access the whole content and functionality of the system.	4	4	4
As a user I want to enter the system 2 different ways with my phone number and password or username and password so I can access the whole content and functionality of the system.	4	5	3

Figure 8. Part of UI as a result of Conflict Identification

4.5 Conflict Resolution Flow Implementation

When we finish identifying the conflicts, we have a data set showing conflicting user stories. To solve the conflict, we decided on one among them, and as understandable from the word "decide," we need a decision-making approach, and we are implementing the AHP decision-making approach.

To implement the AHP method, firstly, we need the metrics, and the metrics are based on PWC. So, we have created the metrics shown in Table 6 and the metrics values are based on Table 2.

Table 6. Sample for AHP metrics for Login user stories

	SM	BM	StM
SM	1	1/3	1/7
BM	3	1	1/5
StM	7	5	1

After forming the metrics, we calculate the geometric mean (M) for the sum of the values of each row separately, and Equations 3, 4, and 5 represent these calculations.

$$M_{SM} = \sqrt[3]{1 * \frac{1}{3} * \frac{1}{7}} = 0.36 \quad (3)$$

$$M_{BM} = \sqrt[3]{3 * 1 * \frac{1}{5}} = 0.84 \quad (4)$$

$$M_{StM} = \sqrt[3]{7 * 5 * 1} = 3.27 \quad (5)$$

When the geometrical mean for each row is ready, we calculated the weights for each row as a second step. This is called normalization. Also, after calculations, we get values for weights between zero and one, and the sum of these weights must equal 1, as shown in Equation 9.

$$W_{SM} = \frac{M_{SM}}{M_{SM} + M_{BM} + M_{StM}} = \frac{0.36}{0.36 + 0.84 + 3.27} = 0.08 = 8\% \quad (6)$$

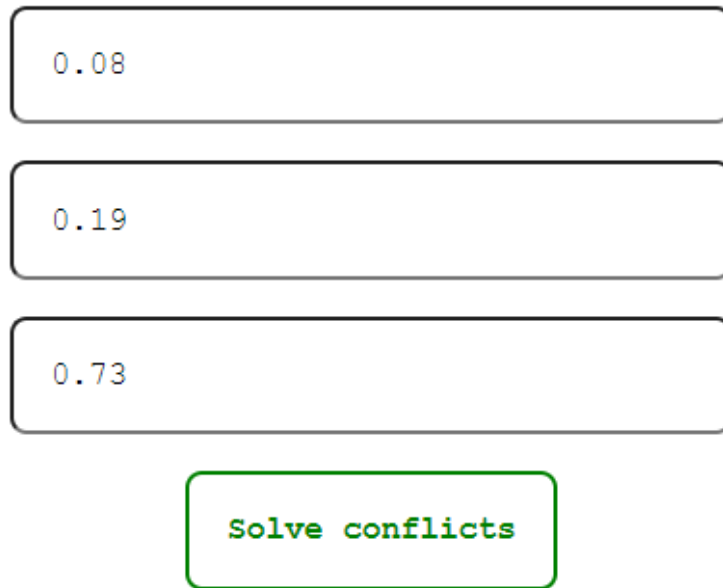
$$W_{BM} = \frac{M_{BM}}{M_{SM} + M_{BM} + M_{StM}} = \frac{0.84}{0.36 + 0.84 + 3.27} = 0.19 = 19\% \quad (7)$$

$$W_{StM} = \frac{M_{StM}}{M_{SM} + M_{BM} + M_{StM}} = \frac{3.27}{0.36 + 0.84 + 3.27} = 0.73 = 73\% \quad (8)$$

$$W_{SM} + W_{BM} + W_{StM} = 0.08 + 0.19 + 0.73 = 1 = 100\% \quad (9)$$

As the last step, we enter the relative importance scores into our software tool, which is also clearly presented in Figure 9. In the end, we click on the button to *solve the conflicts*.

Enter relative importance score to solve conflicts



The image shows a user interface for entering relative importance scores. It consists of three vertically stacked input fields, each containing a numerical value: 0.08, 0.19, and 0.73. Below these fields is a green button with the text "Solve conflicts".

Figure 9. Part of UI for entering weights which gained from PWC

As a result, the selected user story is shown as a popup in the UI after clicking the button *resolve conflicts*. The snapshot of this flow is as shown in Figure 10.

Goals
User stories
Manage conflicts

The system must have LogIn flow

There is 37 conflicting user stories as shown in table

Enter relative importance score to solve conflicts

0.08

The following user story has been selected:

As a user I want to enter the system 4 different ways with my phone number and password or username and password or email and password or with my GitHub account so I can access the whole content and functionality of the system.

End process

Content	Stakeholder Motivation	Business Motivation	Strategic Motivation
As a user I want to enter the system with my phone number and password so I can access the whole content and functionality of the system.	0.24	0.95	2.19
As a user I want to enter the system 2 different ways with my phone number and password or email and			

Figure 10. Part of UI as a result of Conflict resolution

After finding the most suitable user story, the Scrum Master updates the Functional Goal based on the selected user story, shown in Figure 15.

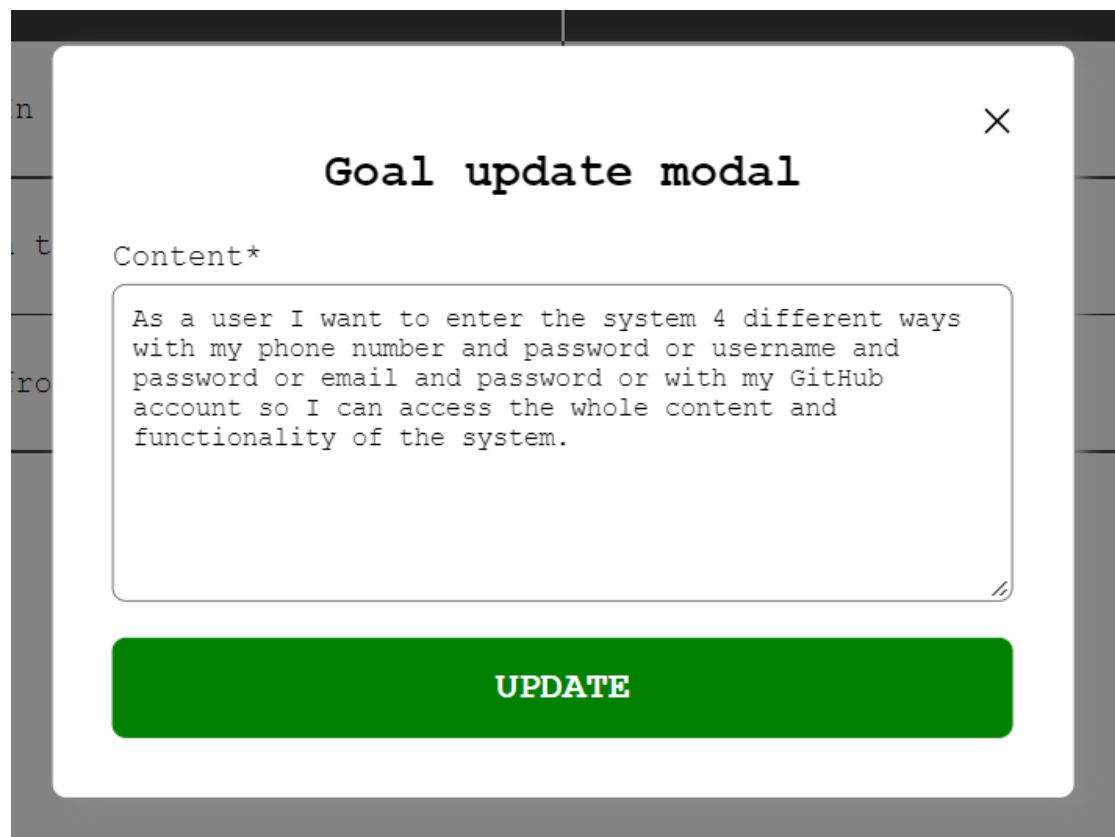


Figure 11. Functional goal update popup

5 Discussions

The following chapter is discussing the results of the research.

5.1 Formation of the Functional Goals

The research is about conflict management in GORE from STS perspective and focused on real practices, and our case study is developing social media platform for developers. In Chapter 3 we explained out full research approach and in Chapter 4 we presented the results of this research approach. Based on the results we understood the role of the goals in the Software Engineering life-cycle, and the way of how to form the goals. The main key points during the development of the functional goals for our case-study which have been found are as follows agile methodology, team collaboration, project managers.

5.2 Conflict Management

After implementing the software tool to identify conflicts among user stories we understood that in most cases one functional goal of the system has more than one user story and every user-story are provided by different stakeholders. For example, for login flow, post sharing flow, and push notification flow there are 37, 19, and 13 different user stories which are shown in Table 5, 8 and 7 accordingly. So in that case, the best decision must be done among these user stories and this decision must not disappoint the stakeholders. To resolve conflicts in the user stories, the software tool, the AHP method is implemented as explained in detailed in the above chapters.

For conflict resolution, our approach is deciding on one user story among all user stories and this decision is based on the weights of the motivation levels of user stories from various perspectives. Two motivational weights out of the three are from the company side and one motivation is based on stakeholders' input. To resolve conflicts we must decide on the user story which is the most suitable for the company's strategic goals, which has a less negative business impact and has as much as possible higher stakeholder motivation. After our implementation, 0.73 (73%), 0.19 (19%), and 0.08 (8%) are the values of the weights for strategic (W_{StM}), business (W_{BM}), and stakeholder (W_{SM}) motivations consequently. As described in the above values also the main focus must be a strategic goal for the company, especially which is on the early stage of the company's foundation, and after that, we must focus on business impact, especially for financial aspects as a newly created company and that is why second importance is given to business motivation and lastly, we must consider stakeholders' needs and deliver a product based on their needs.

5.3 Validation of the Findings

After resolving the conflicts, the functional goal is updated based on its selected user story. When all conflicts are resolved, the team again reviews all final functional goals and discusses whether they are the logical choice or not. For example, in our results for the login, the following user has been selected: "As a user, I want to enter the system 4 different ways with my phone number and password or username and password or email and password or with my GitHub account so that I can access the whole content and functionality of the system." This user story is also the logical choice because email, phone number, and username are some of the most used login flows by all platforms. On the other side, since our case scenario is on the development of social media platforms, all other social media platforms are competitors, and it is not a logical way to login with other social media accounts. It is against the company's strategic goals.

6 Threat to Validity

The following describes the threats to validity of this research:

1. The first threat is the **elicitation** of the user stories from the end-users. Firstly, we decided the best way to collect data via online questionnaire forms is enough. But we observed that usually, end-users are not eager to fill those forms, or they are irresponsibly or not accurately filling them. And this cause to have wrong data and incorrect decisions in the end. To mitigate this threat, we decided to also use face-to-face interviews with end-users.
2. The second threat is about the **applicability** of our research in real software development because our primary motivation is focusing on the industry. To reduce this threat, we presented our scientific approach and implementation to the problem domain experts and explained the conflict management process. The experts reviewed the approach and the results, confirming the method's applicability in the industry and the results.
3. The third threat is to ascertain the **reliability** of the software tool to know if it is working and presenting the results correctly. To mitigate this threat, the framework's source-code has been provided in the research, and every step to implement the methodology is explained in chapter 3. Every step has been provided in the application documentation for running the framework.

7 Conclusion and Future Work

This thesis focused on conflict management in GORE from an STS perspective. The research questions answered focused on a case study of the social media platform. The importance of the agile methodology, human-machine interaction, and decision-making methods was described during the implementation. Also, we implemented a prototype software tool as a web application for conflict management.

Machine learning methods to detect emotion levels from the voice and writings can be applied to that methodology for further work. In that case, asking about the stakeholder's motivation level can be replaced with ML techniques.

On the other hand, the approach which we implemented in this thesis can be applied in different areas where one goal has more than one way to gain.

References

- [1] L. Capra, W. Emmerich, C. Mascolo, A micro-economic approach to conflict resolution in mobile computing, *ACM SIGSOFT Software Engineering Notes* 27 (6) (2002) 31–40.
- [2] G. M. Vecchi, V. B. Van Hasselt, S. J. Romano, Crisis (hostage) negotiation: Current strategies and issues in high-risk conflict resolution, *Aggression and Violent Behavior* 10 (5) (2005) 533–551.
- [3] I. Gambo, K. Taveter, A pragmatic view on resolving conflicts in goal-oriented requirements engineering for socio-technical systems, in: *Proceedings of the 16th International Conference on Software Technologies, ICSOFT 2021, July 6-8, 2021, 2021*, pp. 333–341. doi:10.5220/0010605703330341.
- [4] R. E. Freeman, *Strategic management: A stakeholder approach*, Cambridge university press, 2010.
- [5] H. Eridaputra, B. Hendradjaya, W. D. Sunindyo, Modeling the requirements for big data application using goal oriented approach, in: *2014 international conference on data and software engineering (ICODSE), IEEE, 2014*, pp. 1–6.
- [6] S. I., *Software Engineering*, Pearson Education, 2010.
- [7] J. Horkoff, F. B. Aydemir, E. Cardoso, T. Li, A. Maté, E. Paja, M. Salnitri, L. Piras, J. Mylopoulos, P. Giorgini, Goal-oriented requirements engineering: an extended systematic mapping study, *Requirements engineering* 24 (2) (2019) 133–160.
- [8] L. Vijayasarathy, D. Turk, Agile software development: A survey of early adopters, *Journal of Information Technology Management* 19 (2) (2008) 1–8.

- [9] J. Horkoff, F. B. Aydemir, E. Cardoso, T. Li, A. Maté, E. Paja, M. Salnitri, J. Mylopoulos, P. Giorgini, Goal-oriented requirements engineering: a systematic literature map, in: 2016 IEEE 24th International Requirements Engineering Conference (RE), IEEE, 2016, pp. 106–115.
- [10] V. One, 9th annual state of agile survey, Survey. Accessed online 15 (2015).
- [11] R. van Dijk, Determining the suitability of agile methods for a software project, 2011.
- [12] A. R. Amna, G. Poels, Ambiguity in user stories: A systematic literature review, *Information and Software Technology* 145 (2022) 106824. doi:<https://doi.org/10.1016/j.infsof.2022.106824>.
URL <https://www.sciencedirect.com/science/article/pii/S0950584922000040>
- [13] W. Ruchira Prasad, G. Perera, K. Jeeva Padmini, H. Dilum Bandara, Adopting design thinking practices to satisfy customer expectations in agile practices: A case from sri lankan software development industry, in: 2018 Moratuwa Engineering Research Conference (MERCon), 2018, pp. 471–476. doi:10.1109/MERCon.2018.8422006.
- [14] L. Ramingwong, A review of requirements engineering processes, problems and models, *International Journal of Engineering Science and Technology* 4 (06 2012).
- [15] U. Hübner, E. Cruel, M. Gök, M. Garthaus, M. Zimansky, H. Remmers, O. Rienhoff, Requirements engineering for cross-sectional information chain models, in: NI 2012: 11th International Congress on Nursing Informatics, June 23-27, 2012, Montreal, Canada., Vol. 2012, American Medical Informatics Association, 2012.
- [16] N. A. Ernst, *Software Evolution: a Requirements Engineering Approach*, University of Toronto (Canada), 2012.
- [17] A. Lapouchnian, Goal-oriented requirements engineering: An overview of the current research, *University of Toronto* 32 (2005).
- [18] A. Van Lamsweerde, Goal-oriented requirements engineering: A guided tour, in: *Proceedings fifth ieee international symposium on requirements engineering*, IEEE, 2001, pp. 249–262.
- [19] G. Regev, A. Wegmann, Where do goals come from: the underlying principles of goal-oriented requirements engineering, in: 13th IEEE International Conference on Requirements Engineering (RE'05), Ieee, 2005, pp. 353–362.

- [20] F. Emery, Characteristics of socio-technical systems (document no. 527), London: Tavistock (1959).
- [21] G. Baxter, I. Sommerville, Socio-technical systems: From design methods to systems engineering, *Interacting with Computers* 23 (1) (2010) 4–17. arXiv:<https://academic.oup.com/iwc/article-pdf/23/1/4/2038336/iwc23-0004.pdf>, doi:10.1016/j.intcom.2010.07.003. URL <https://doi.org/10.1016/j.intcom.2010.07.003>
- [22] N. Zimmer, K. Taveter, Requirements engineering for sociotechnical systems: Case study of an airline operations control center., in: RESOSY@ APSEC, 2021.
- [23] P. M. Salmon, G. H. Walker, N. A. Stanton, Pilot error versus sociotechnical systems failure: a distributed situation awareness analysis of air france 447, *Theoretical Issues in Ergonomics Science* 17 (1) (2016) 64–79.
- [24] L. Sterling, K. Taveter, *The art of agent-oriented modeling*, MIT press, 2009.
- [25] K. A. Jehn, C. Bendersky, Intragroup conflict in organizations: A contingency perspective on the conflict-outcome relationship, *Research in organizational behavior* 25 (2003) 187–242.
- [26] D. Damian, L. Izquierdo, J. Singer, I. Kwan, Awareness in the wild: Why communication breakdowns occur, in: *International Conference on Global Software Engineering (ICGSE 2007)*, 2007, pp. 81–90. doi:10.1109/ICGSE.2007.13.
- [27] I. Kwan, D. Damian, The hidden experts in software-engineering communication: Nier track, in: *2011 33rd International Conference on Software Engineering (ICSE)*, 2011, pp. 800–803. doi:10.1145/1985793.1985906.
- [28] D. T. Ross, K. E. Schoman, Structured analysis for requirements definition, *IEEE transactions on Software Engineering* (1) (1977) 6–15.
- [29] A. Dardenne, S. Fickas, A. van Lamsweerde, Goal-directed concept acquisition in requirements elicitation, in: *Proceedings of the Sixth International Workshop on Software Specification and Design*, IEEE Computer Society, 1991, pp. 14–15.
- [30] A. Van Lamsweerde, R. Darimont, E. Letier, Managing conflicts in goal-driven requirements engineering, *IEEE transactions on Software engineering* 24 (11) (1998) 908–926.
- [31] K. Yue, What does it mean to say that a specification is complete?, in: *Proc. IWSSD-4, Fourth International Workshop on Software Specification and Design*, 1987.

- [32] I. Sommerville, P. Sawyer, *Re: a good practice guide*, John Wiley and Sons 113 (1997) 114.
- [33] B. Curtis, H. Krasner, N. Iscoe, A field study of the software design process for large systems, *Communications of the ACM* 31 (11) (1988) 1268–1287. doi: <https://doi.org/10.1145/50087.50089>.
- [34] R. Lutz, S. Schäfer, S. Diehl, Using mobile devices for collaborative requirements engineering, in: *Proceedings of the 27th IEEE/ACM International Conference on Automated Software Engineering*, 2012, pp. 298–301.
- [35] M. Deutsch, *The resolution of conflict: Constructive and destructive processes*, Yale University Press, 1973.
- [36] I. Gambo, K. Taveter, Stakeholder-centric clustering methods for conflict resolution in the requirements engineering process, in: *International Conference on Evaluation of Novel Approaches to Software Engineering*, Springer, 2021, pp. 183–210.
- [37] M. Aldekhail, A. Chikh, D. Ziani, Software requirements conflict identification: review and recommendations, *International Journal of advanced computer science and applications* 7 (10) (2016).
- [38] T. Moser, D. Winkler, M. Heindl, S. Biffl, Requirements management with semantic technology: An empirical study on automated requirements categorization and conflict analysis, in: *International Conference on Advanced Information Systems Engineering*, Springer, 2011, pp. 3–17.
- [39] W. N. Robinson, Surfacing requirements interactions, in: *Perspectives on Software Requirements*, Springer, 2004, pp. 69–90.
- [40] A. Sardinha, R. Chitchyan, J. Araújo, A. Moreira, A. Rashid, Conflict identification with ea-analyzer, in: *Aspect-oriented requirements engineering*, Springer, 2013, pp. 209–224.
- [41] M. Amroune, J. M. Inglebert, P. J. Charrel, N. Zarour, A conflict resolution process in aspecis approach, *International Journal of Computer Applications* 44 (10) (2012) 14–21.
- [42] A. Rashid, A. Moreira, J. Araújo, Modularisation and composition of aspectual requirements, in: *Proceedings of the 2nd international conference on Aspect-oriented software development*, 2003, pp. 11–20.
- [43] A. De Lucia, A. Qusef, Requirements engineering in agile software development, *Journal of emerging technologies in web intelligence* 2 (3) (2010) 212–220.

- [44] B. Boehm, P. Bose, E. Horowitz, M. J. Lee, Software requirements negotiation and renegotiation aids: A theory-w based spiral approach, in: 1995 17th International Conference on Software Engineering, 1995, pp. 243–243. doi:10.1145/225014.225037.
- [45] W. Werther, H. Weihrich, Refining mbo through negotiations, *MSU Business Topics* 23 (3) (1975) 53–59.
- [46] W. Prasad, I. Perera, K. Padmini, D. Bandara, Adopting design thinking practices to satisfy customer expectations in agile practices: A case from sri lankan software development industry, 2018, pp. 471–476. doi:10.1109/MERCon.2018.8422006.
- [47] R. Shankarmani, R. Pawar, S. Mantha, V. Babu, Agile methodology adoption: Benefits and constraints, *International Journal of Computer Applications* 58 (10 2012). doi:10.5120/9361–3698.
- [48] W. Ruchira Prasad, G. Perera, K. Jeeva Padmini, H. Dilum Bandara, Adopting design thinking practices to satisfy customer expectations in agile practices: A case from sri lankan software development industry, in: 2018 Moratuwa Engineering Research Conference (MERCon), 2018, pp. 471–476. doi:10.1109/MERCon.2018.8422006.
- [49] Y. Shastri, R. Hoda, R. Amor, The role of the project manager in agile software development projects, *Journal of Systems and Software* 173 (2021) 110871. doi:https://doi.org/10.1016/j.jss.2020.110871.
URL <https://www.sciencedirect.com/science/article/pii/S0164121220302612>
- [50] J. Highsmith, *Agile project management: creating innovative products*, Pearson education, 2009.
- [51] F. Dalpiaz, S. Brinkkemper, Agile requirements engineering: From user stories to software architectures, in: 2021 IEEE 29th International Requirements Engineering Conference (RE), 2021, pp. 504–505. doi:10.1109/RE51729.2021.00076.
- [52] G. Lucassen, F. Dalpiaz, J. M. E. van der Werf, S. Brinkkemper, The use and effectiveness of user stories in practice, in: *International working conference on requirements engineering: Foundation for software quality*, Springer, 2016, pp. 205–222.
- [53] M. Cohn, *User stories applied: For agile software development*, Addison-Wesley Professional, 2004.

- [54] A. R. Amna, G. Poels, Ambiguity in user stories: A systematic literature review, *Information and Software Technology* 145 (2022) 106824. doi:<https://doi.org/10.1016/j.infsof.2022.106824>.
URL <https://www.sciencedirect.com/science/article/pii/S0950584922000040>
- [55] H. Bendjenna, P. Charrel, N. Zarour, Using ahp method to resolve conflicts between non-functional concerns, in: *International conference on education, applied sciences and management (ICEASM'2012)*, 2012, pp. 26–27.
- [56] M. Derntl, D. Renzel, P. Nicolaescu, I. Koren, R. Klamma, Distributed software engineering in collaborative research projects, in: *2015 IEEE 10th International Conference on Global Software Engineering*, IEEE, 2015, pp. 105–109.
- [57] I. Mistrík, J. Grundy, A. v. d. Hoek, J. Whitehead, Collaborative software engineering: challenges and prospects, *Collaborative software engineering* (2010) 389–403.
- [58] A. Vogelsang, M. Borg, Requirements engineering for machine learning: Perspectives from data scientists, in: *2019 IEEE 27th International Requirements Engineering Conference Workshops (REW)*, IEEE, 2019, pp. 245–251.
- [59] W. Zhang, H. Mei, H. Zhao, Feature-driven requirement dependency analysis and high-level software design, *Requirements Engineering* 11 (3) (2006) 205–220.
- [60] T. Hassan, M. Z. Hussain, M. Z. Hasan, Z. Ullah, N.-u. Qamar, Quantitative based mechanism for resolving goals conflicts in goal oriented requirement engineering, in: *International Conference on Intelligent Technologies and Applications*, Springer, 2018, pp. 822–831.
- [61] P. K. Murukannaiah, A. K. Kalia, P. R. Telangy, M. P. Singh, Resolving goal conflicts via argumentation-based analysis of competing hypotheses, in: *2015 IEEE 23rd International Requirements Engineering Conference (RE)*, IEEE, 2015, pp. 156–165.
- [62] T. Viana, A. Zisman, A. K. Bandara, Identifying conflicting requirements in systems of systems, in: *2017 IEEE 25th International Requirements Engineering Conference (RE)*, IEEE, 2017, pp. 436–441.
- [63] K. M.-S. Al-Harbi, Application of the ahp in project management, *International Journal of Project Management* 19 (1) (2001) 19–27. doi:[https://doi.org/10.1016/S0263-7863\(99\)00038-1](https://doi.org/10.1016/S0263-7863(99)00038-1).
URL <https://www.sciencedirect.com/science/article/pii/S0263786399000381>

- [64] T. L. Saaty, Decision making for leaders: the analytic hierarchy process for decisions in a complex world, RWS publications, 2001.
- [65] T. L. Saaty, What is the analytic hierarchy process?, in: Mathematical models for decision support, Springer, 1988, pp. 109–121.

Appendix

Table 7. User stories for notification flow

Begin of Table				
Id	Content	SM	BM	StM
US1	As a user I want to get notifications only through the application, so I can get up to date about all news.	1	5	2
US2	As a user I want to get notifications only with email, so I can get up to date about all news.	3	5	1
US3	As a user I want to get notifications only with SMS, so I can get up to date about all news.	2	5	1
US4	As a user I want to get notifications from the system and with email at the same time, so I can get up to date about all news.	5	3	5
US5	As a user I want to get notifications from the system and with SMS at the same time, so I can get up to date about all news.	3	3	4
US6	As a user I want to get notifications with email and with SMS at the same time, so I can get up to date about all news.	3	3	1
US7	As a user I want to get notifications with email and with SMS at the same time, so I can get up to date about all news.	3	3	1
US8	As a user I want to get notifications through application and with email and with SMS at the same time, so I can get up to date about all news.	2	2	4
US9	As a user I want to get notifications from the system or with email at the same time, so I can get up to date about all news.	4	1	2
US10	As a user I want to get notifications from the system or with SMS at the same time, so I can get up to date about all news.	4	1	2
US11	As a user I want to get notifications with email or with SMS at the same time, so I can get up to date about all news.	3	1	1
US12	As a user I want to get notifications with email or with SMS at the same time, so I can get up to date about all news.	3	1	1

Continuation of Table 7				
Id	Content	SM	BM	StM
US13	As a user I want to get notifications through application or with email or with SMS at the same time, so I can get up to date about all news.	5	1	1
End of Table				

There is 13 conflicting user stories as shown in table

Enter relative importance score to solve conflicts

0.08

The following user story has been selected:

As a user I want to get notifications from the system and with email at the same time, so I can get up to date about all news.

End process

Content	Stakeholder Motivation	Business Motivation	Strategic Motivation
As a user I want to get notifications only through the application, so I can get up to date	0.08	0.95	1.46

Figure 12. Selected user story for notification flow

Table 8. User stories for sharing a post flow

Begin of Table				
Id	Content	SM	BM	StM
US1	As a user I want to share a post as a single text without any additional features so I can share my thoughts and feelings with others.	1	5	1
US2	As a user I want to share a post as a rich textarea text so I can share my thoughts and feelings with others.	2	4	1
US3	As a user I want to share a post as a single text and at the same I want to tag someone to my post so I can share my thoughts and feelings with others.	2	5	2

Continuation of Table 8				
Id	Content	SM	BM	StM
US4	As a user I want to share a post as a rich textarea text and at the same I want to tag someone to my post so I can share my thoughts and feelings with others.	2	4	2
US5	As a user I want to share a post as a single media without any additional features so I can share my thoughts and feelings with others.	2	5	1
US6	As a user I want to share a post as a media collection without any additional features so I can share my thoughts and feelings with others.	2	4	1
US7	As a user I want to share a post as a single media and at the same I want to tag someone to my post so I can share my thoughts and feelings with others.	2	4	1
US8	As a user I want to share a post as a media collection and at the same I want to tag someone to my post so I can share my thoughts and feelings with others.	2	3	1
US9	As a user I want to share a post as a single media and single text so I can share my thoughts and feelings with others.	3	3	2
US10	As a user I want to share a post as a single media or as a single text so I can share my thoughts and feelings with others.	4	3	2
US11	As a user I want to share a post as a single media and rich textarea text and at the same I want to tag someone to my post so I can share my thoughts and feelings with others.	4	2	5
US12	As a user I want to share a post as a single media or rich textarea text and at the same I want to tag someone to my post so I can share my thoughts and feelings with others.	4	1	5
US13	As a user I want to share a post as a media collection and text so I can share my thoughts and feelings with others.	4	1	3
US14	As a user I want to share a post as a media collection or text so I can share my thoughts and feelings with others.	2	2	3

Continuation of Table 8				
Id	Content	SM	BM	StM
US15	As a user I want to share a post as a media collection and rich text area text and at the same I want to tag someone to my post so I can share my thoughts and feelings with others.	4	1	3
US16	As a user I want to share a post as a media collection or rich text area text and at the same I want to tag someone to my post so I can share my thoughts and feelings with others.	3	1	3
US17	As a user I want to share a post as a single hashtag so I can share my thoughts and feelings with others.	2	5	1
US18	As a user I want to share a post as a media(s) or text and I want to add a hashtag to the post so I can share my thoughts and feelings with others.	2	4	1
US19	As a user I want to share a post as a single media or single text and I want to add a hashtag to the post and at the same time tag someone to the post so I can share my thoughts and feelings with others.	5	2	5
End of Table				

There is 19 conflicting user stories as shown in table

Enter relative importance score to solve conflicts

0.08

The following user story has been selected:

As a user I want to share a post as a single media or single text and I want to add a hashtag to the post and at the same time tag someone to the post so I can share my thoughts and feelings with others.

End process

Content	Stakeholder Motivation	Business Motivation	Strategic Motivation
a user I want to share post as a single text without any additional features so I can share thoughts and feelings	0.08	0.95	0.73

Figure 13. Selected user story for post sharing flow

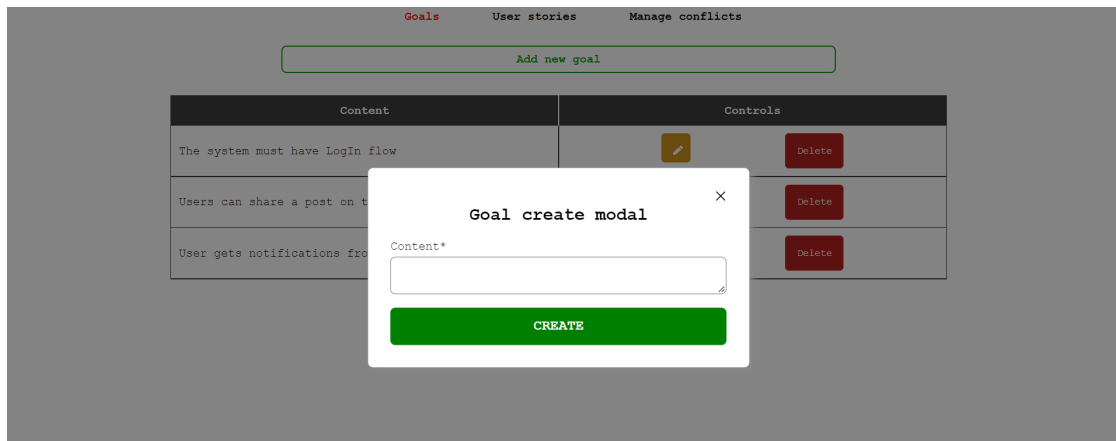


Figure 14. Modal for creating functional goal in the UI

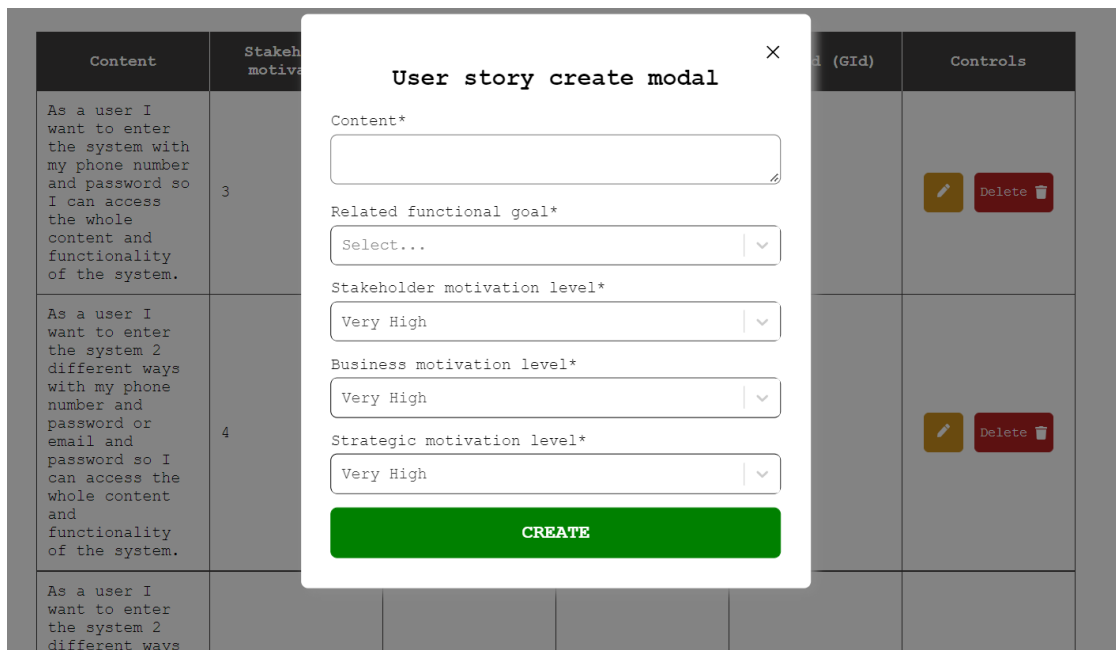


Figure 15. Modal for creating user story in the UI

Lisad

I. Glossary

Table 9. List of the acronyms in the research

Acronym	Description
RE	Requirement Engineering
GORE	Goal oriented Requirement Engineering
STS	Socio Technical Systems
RQ	Research Question
KAOS	Knowledge Acquisition in automated Specification
GBRAM	Goal Based Requirements Analysis Method
PWC	Pair Wise Comparison
JAD	Joint Application Development
IT	Information technology
XP	Extreme Programming
AHP	Analytical hierarchical process
FG	Functional goal
US	User story
MVP	Minimum viable product
UI	User Interface

II. Licence

Non-exclusive licence to reproduce thesis and make thesis public

I, Rasul Agharzayev,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

Conflicts Management in Goal-Oriented Requirements Engineering: Socio-Technical Systems Perspective,

supervised by Ishaya Peni Gambo, PhD.

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.
3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.
4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Rasul Agharzayev

07/08/2022