A Proactive Approach to Reduce Requirements Ambiguity via Gamify4Req

Hafsa Shareef Dar[*], Salma Imtiaz[**], Muhammad Ikramullah Lali[***]

[*] Department of Software Engineering, University of Gujrat; International Islamic University Islamabad, Pakistan
[**] Department of Software Engineering, International Islamic University Islamabad, Pakistan
[***] Department of Information Sciences, University of Education Lahore, Pakistan

Abstract- Ambiguity in natural language requirements is an inherent challenge. Ambiguity arises when the user does not participate in the system development. Ambiguity once incorporated seeps into later stages of system development. In traditional software development approaches ambiguity is addressed at the inspection phase, which is manual, labour intensive and costly activity. Researchers have proposed several methods, techniques, approaches, tools etc. for avoidance, detection, and removal of ambiguity once requirements are specified. However, all the approaches utilize time and effort and require manual intervention. To the best of our knowledge, there are very few studies present in literature that focus on reducing ambiguity during requirements elicitation. Traditional techniques are reactive and address ambiguity once requirements are specified. We propose that a pro-active approach is needed for addressing ambiguity early i.e., during elicitation. Doing so will result in unambiguous requirements that can lead to successful development. The proposed approach uses gamification i.e., use of game elements to engage users and keep them interested while performing the task at hand i.e., elicitation of unambiguous requirement. The approach is supported via Gamified tool *Gamify4Req*, which uses the game concepts of avatar, leaderboard and points and badges. The validation of the tool is performed on two case studies selected from software market, and user involvement is measured with help of a questionnaire. The findings show that the gamified tool outperforms the existing methods used in the industry for reducing ambiguity in requirements both in terms of number of identified ambiguities and time spent on the task. Moreover, the tool ensures user involvement which is measured with help of user involvement survey. usage activity and survey of users.

Index Terms- Requirements elicitation, semantic ambiguity, ambiguity rules, gamification, game elements, PBL (points, badges, leaderboards), game rules, POS tags

I. INTRODUCTION

Requirements written in Natural Language (NL) are prone to serious problems such as incompleteness, vagueness, and ambiguity due to the nature of English language. Other ways to collect requirements are present in the form of formal languages that reduce ambiguity. The formal languages elicit requirements with help of mathematical expressions and predicates, but lack the depth of concept expression [1]. Moreover specifying requirements via formal methods requires are time taking and costly [2]. According to the literature, ambiguity in requirements is a more complex issue when compared to other related problems such as incomplete requirements [3]. Ambiguity in requirements is characterized by a statement that has multiple meanings, regardless of the reader's familiarity with the RE context. In traditional methods of ambiguity handling, ambiguity is addressed during inspection phase, where requirements are specified in Software Requirements Specification (SRS) document. During ambiguity identification and removal using SRS document, users are not involved in the inspection [4]. Due to which, ambiguity remains the critical challenge.

A. Requirements Ambiguity

Ambiguity in requirements specification can lead to incompleteness, inconsistency, and misunderstanding [2], caused by NL expressions that stem from human judgment of real-world scenarios. These inconsistencies carry over to later stages of software development [5]. However, pinpointing the source of ambiguity can help mitigate its impact during initial phases of software development. Addressing ambiguity can also help in removing inconsistencies and incompleteness. Various researchers have categorized ambiguity into different types and taxonomies [6] [7]. One classification of ambiguity categorizes ambiguity into lexical, semantic, syntactic, pragmatic, and language errors.

Each type of ambiguity caters for different ambiguity problems. *Lexical* ambiguity occurs when a word contains multiple meanings [8] [9]. It is further divided into two sub-types: homonymy and polysemy. *Syntactic* ambiguity occurs due to parsing of a sentence in multiple ways having different meanings [10] [11]. It has one major sub-type of attachment ambiguity [9]. *Semantic* ambiguity occurs when a sentence has multiple interpretations without any syntactic or lexical ambiguity [10]. Referential, coordination [12], and scope ambiguity [13] are the sub-types of sematic ambiguity. *Pragmatic* ambiguity occurs when human contextual knowledge and common-sense knowledge are uncertain [8] [11]. Whereas *Language Errors* are caused due to poor grammatical structure [14].

As requirements in NL tend to be ambiguous, thus pre-processing of software requirements specifications (SRS) document is required along with natural language processing (NLP) techniques for ambiguity detection and resolution [15]. However, manual ambiguity resolution of software requirements is not only time consuming, error prone and but also a costly process [16] [17].

B. Requirements Elicitation Techniques

In requirements elicitation, requirements are defined in a more broader context focusing on the knowledge area where system is applied, customer problems, system interaction environment, user needs, and constraints of system under development [19]. Requirements elicitation involves different types of elicitation methods such as traditional [20] [21], observational, contextual or social analysis, [22], and cognitive/analytical to collect the requirements. Each type further consists of different elicitation techniques such as interviews, questionnaire, focus groups, workshops, brainstorming, social analysis, and observation etc., which are used in accordance with the project nature and choice of the technical team.

Elicitation becomes a time taking activity because it involves stakeholders from different domains, so it becomes difficult to make them understand each other perspective. Traditional elicitation techniques comprise of challenges that affect the quality of collected requirements. These challenges include problems of scope, requirements understanding and volatility of requirements [23]. Ensuring stakeholder's presence in the elicitation activity is critical for the success of development [21]. Users are reluctant to give information as they are busy or consider this activity a burden. Similarly, prototyping is another way of getting user feedback. It is typically developed using preliminary requirements [24]. Prototype is challenging due to insufficient analysis and excessive development time [25]. Storyboarding describes system functionality for a specific scenario that includes interaction between user and the system [26]. One of the biggest problems with storyboarding is that they outdate very quickly, and user interfaces often change over time creating a maintenance burden.

Requirements requires users' interest to elaborate system functionalities and behavior [43]. None of the existing elicitation methods focus on active participation of users in the process. For this reason, software systems can fail due to lack of user input [27], erroneous, ambiguous and misunderstood requirements [28]. Table 1 presents some of the challenges of traditional elicitation techniques.

Table 1: Challenges of Requirements Elicitation Techniques

No.	Challenges of Elicitation Techniques	Reference
1.	Lack of user involvement	[29]
2.	Active participation of the users	[30]
3.	Limitation of responsiveness and sensitivity of observation to real environment	[31]
4.	Much effort is required on conducting the activity	[21] [32]
5.	Complex communication between stakeholder and analyst	[33] [34]
6.	Setting up meeting environment every time	[21] [32]
7.	More time consuming than the usual activities even in tight schedule	[31]
8.	Proper analysis of records of interaction in meeting is required	[21] [32]
9.	Ensuring the presence of stakeholders	[21] [32]
10.	Hard to specify and analyse the perception of observer	[31]
11.	Domain knowledge must be clear because analysis is based on expert's knowledge about the process and system	[20]
12.	Scheduling the meeting every time	[21] [32]

One of the important challenges is lack of user involvement in the activity leading to vague and ambiguous requirements. Researchers have explored more interactive interfaces [35] [36] [37] [38] to involve and engage users in requirements elicitation.

C. Gamification in Requirements Elicitation

Gamification helps to engage users in the elicitation activity [31] [32] [33] by making it an interesting task. Gamification uses game elements in non-gaming systems [34] [35]. Past studies [36] reflect that the participants involved in gamification agree to the significance of this approach and recommend it where passiveness, boredom, and repetition are common factors. To overcome these factors, gamification uses interactive and interesting gaming features to encourage and motivate users to participate in the system. The commonly used game elements discussed in literature are leaderboard [37], points and badges [38], awards, rewards, levels, quests, avatars [39], and stories [40]. The selection of game elements majorly depends on user roles and designer's instinct [41]. Gamification is used to elicit user requirements and help in goals identification [42].

We have used 'gamification' for the purpose of engaging users in the system. Gamification enhances user involvement with unique game elements and game rules. We have designed and developed a web-based gamified tool for reducing ambiguity in requirements during elicitation. Currently the scope of this work is limited to semantic ambiguity only. The tool is validated on two case studies conducted from local IT companies. Furthermore, to evaluate whether users participated and enjoyed the activity, a user feedback survey was conducted. The tool provides support in involving and engaging users, and documenting requirements throughout the process.

The rest of the paper is organized into different sections. Review of literature covering related studies is presented in section 2. Section 3 elaborates the proposed approach for reducing requirements ambiguity during elicitation via gamification and development of the gamified tool. Case studies are discussed in section 4. Section 5 presents the results and discussion, whereas section 6 concludes the work and presents future directions.

II. BACKGROUND AND MOTIVATION

The scope of this study is limited to semantic ambiguity. Semantic ambiguity occurs when a sentence has more than one meaning or ways of reading it within the given context, and it does not contain any lexical or structural ambiguity [5] [9]. It means, under this ambiguity there can be multiple expressions of a sentence if translated [43]. Semantic ambiguity has three ambiguity types i.e. referential, coordination, and scope ambiguity [8].

Coordination ambiguity has further two kinds: 1) when "*and"* or "*or"*, more than one conjunction is used in a sentence. For example: *I saw John and Jim and Larry saw me*. This sentence is ambiguous to the reader and can be corrected by using a well-placed comma, and 2) when one conjunction is used with a modifier [5]. For example: *young man and woman* This sentence can be read as either young (man and woman) or as (young man) and woman.

Referential ambiguity arises when an anaphor takes its reference from more than one element, each element acting as antecedent

[44]. For example: *The trucks shall treat the roads before **they** freeze.* The antecedent of anaphor '*they'* is either trucks or roads.

A. Reducing Requirements Ambiguity during Elicitation

Previous studies have suggested that ambiguity occurs due to the difference of customer's meaning of information and assignment of meaning by the analyst [13]. This articulation of meaning involves any word or set of words in a sentence (spoken form), articulated by the user. This articulation becomes more important when NL based requirements are involved. Requirements specified in NL may have issues of ambiguity, which is mainly handled by four approaches: avoidance, detection, reduction, and removal [2] [45] [46] [47]. There is less work present in literature that focuses on reducing requirements ambiguity.

A three-step, semi-automatic method is proposed along with a prototype tool to identify ambiguities and inconsistencies in SRS written in NL [48]. The proposed method works in a combination of human reasoning and strengths of automation for reviews and inspections. In the first step, tool parses SRS with the help of constraining grammar. During parsing, based on the exposed relationship, the tool creates classes, methods, variables, and associations for an object-oriented analysis model. In the third step, the model is diagrammed so that human reviewer can use the model for detection of ambiguity and inconsistency. The researchers use case studies to demonstrate the proposed model. The approach is designed to identify syntactic and semantic ambiguities, by the model. There is no user/client involvement in reducing requirements ambiguity. In a similar work [49], the authors presented an approach to minimize ambiguity in NL based SRS. The work addresses the obstacle that occurs due to informal nature of English language. A controlled NL representation for requirements if considered for generating accurate and consistent software models. An automated approach is proposed to generate Semantic of Business Vocabulary and Rules (SBVR), which is processed by the machine and results in accurate generation of software models. The approach focuses on lexical, syntactic, and semantic ambiguities, and considers NL SRS as an input. Like the previous approach, the approach is not pro-active, hence no user/client is involved during elicitation.

The work reduces ambiguity in requirements provided by the customer through historical rule-based knowledge and a scripted process [50]. The scripted process is a guide that encourages a novice developer to reduce ambiguity, whereas rule-based knowledge involves the requirements of previously implemented web-based applications. This approach is used on web-forms only. The approach is validated via exploratory case study. Performance evaluation of the conceptual model proposed in this work is not validated, due to which accuracy is highly dependent on the collected data. The measurements to identify ambiguity in requirements are also employed in the same way, and can be biased. Similarly, an approach using multilingual word sense to identify and reduce lexical, syntactic, and syntax ambiguities in requirements is proposed [51]. It uses POS tagging to detect ambiguity in a requirement provided by the analyst. POS uses a multilingual word sense dictionary which includes different kinds of senses related to domain. The ambiguous words are compared with the POS provided tags to classify words as lexical, syntactic, and syntax. Once ambiguous words are identified, these are then replaced by alternative unambiguous words, after discussion with the stakeholders. There is no concrete model or prototype provided in this approach. Table 2 shows related studies on requirements ambiguity.

Table 2: Related studies on Requirements Ambiguity Reduction in RE

Source	Title	Framework/Tool/Technique/Approach	Identified RE Area	Identified Amb.	Addressing Ambiguity				
					Detection	Avoidance	Reduction	Resolution	Removal
2022 [52]	Using NLP Tools to Detect Ambiguities in System Requirements - A Comparison Study	• Four tools used on a set of system reqs. • Shows different recall and precision values	NL Reqs.	Semantic Amb.	✓	-	-	-	-
2022 [53]	Automated handling of anaphoric ambiguity in requirements: a multi-solution study	• Auto handling of anaphoric amb. with NLP and ML tech. • SpanBERT for detection	Reqs. Specification	Anaphoric Amb.	✓	-	-	✓	-
2022 [54]	TAPHSIR: towards Anaphoric ambiguity detection and Resolution in requirements	• TAPSHIR is developed for ambiguity detection and resolution in requirements. • Reviews pronouns and revises the pronouns that create amb.	Reqs. statements	Semantic Amb.	✓	-	-	✓	-
2022 [55]	Identifying Ambiguity Problems in User Stories: A Proposed Framework	• A framework to identify ambiguity in user stories. • Human centered factors are identified. • Framework is evaluated by conducting experiments, to test its effectiveness.	Reqs elicitation: user stories, validation	Semantic Amb.	✓	-	-	-	-
2021 [56]	An Intelligent Analytics Approach to Minimize Complexity in Ambiguous Software Requirements	• A framework for NL to CNL • Proactive approach that uses SBVR. • SBVR provides semantic formulations to make English statements ambiguity free. • Yields higher accuracy as compared to other tools with 0.94 recall and 0.97 precision value.	Reqs. elicitation	Semantic Amb.	✓	-	✓	-	-
2021 [57]	Natural language ambiguity resolution by intelligent semantic annotation of software requirements	• Use of SBVR based CNL to capture reqs. and prepare SRS. • Two sets of reqs., manually validated	Reqs. Document	Semantic Amb.	-	-	-	✓	-
2019 [58]	A framework for software requirement ambiguity avoidance	• SRAAF helps to write unambiguous reqs. by selecting appropriate elicitation techniques • Works with W6H techniques for the evaluation of different attributes	Selection of elicitation technique, SRS document	Semantic Amb.	-	✓			✓
2018 [59]	Pinpointing Ambiguity and Incompleteness in Requirements Engineering via Information Visualization and NLP	• To identify ambiguity in requirements • Investigates human cognitive and analytical abilities with automated reasoning. • Tool pinpoints ambiguities between different viewpoints and missing requirements.	Reqs. Statements	Semantic Amb.	✓	-	-	-	-
2018 [60]	Flexible Ambiguity Resolution and Incompleteness Detection in Requirements Descriptions via an Indicator-Based Configuration of Text Analysis Pipelines	• Developed software to create unamb. reqs. description by combining expert tools. • Developed linguistic indicators	Reqs. description	Refrential Amb., Lexical Amb.	✓	-	-	✓	-
2016 [61]	Ambiguity and tacit knowledge in requirements elicitation interviews	• Designed theoretical framework for categorization of amb. in interviews. • Based on correct and incorrect disambiguation	Reqs. gathered during interviews	Semantic Amb.	✓	-	-	-	-

The existing ambiguity detection and removal techniques in NL are reactive and do not involve users in the activity. There are several studies present in literature that detect requirements ambiguity at different levels [59] [65] [66] [56] [55] [54] [52], and very few studies that are present on ambiguity reduction [48] [49] [50] [51], as given in Table 1.

B. Related Work on Requirements Elicitation Gamified Systems

Gamification provides and interactive platform for requirements elicitation by keeping users engaged throughout the activity [32]. In this way not only, users get engaged but also provide the required information about the system. Researchers have explored gamification and game elements in past studies and reported potential benefits of using gamification during elicitation.

A gamified tool is developed to enhance user participation and stakeholders collaboration [67] using Outsystems Agile Platform. iThink uses the concept of 'six thinking hats' from creative thinking, where each hat represents certain activity related to requirements, along with generation of new ideas. Players are rewarded upon generation of new requirement and refinement of any requirement. iThink involves three user roles acting as players; and game element 'Points' as a reward for performing and achieving the target. The tool is validated on two case studies where the game elements help to motivate and engage the users, but results are dependent on the generation of new ideas. The interface of iThink is unappealing having low amusement for the users involved.

An approach titled as Requirement Elicitation and Verification Integrated in Social Environment (REVISE) [68] is proposed. It enhances knowledge sharing and collaboration among project teams. It uses CARE principles i.e., create, ask for review, review, and extend among user roles involved in the system including creator, reviewer, and customer. The users are given scores upon adding new requirements and performing traceability of requirements. The study also uses other game elements such scores, badges, leaderboard, and profile to improve user involvement. Similarly, to enhance user involvement in the system, Gamified Requirement Engineering Model (GREM) is designed [33]. GREM involves customers during elicitation, to improve system performance. The model describes three variables including gamification, engagement of stakeholders, and performance. Variables, motivation, and emotions are measured by Dichotomous variable, Reiss Profile, and PANAS (Positive and Negative Affect Schedule) respectively. More than one game element is used. User participation is enhanced with points, badges, leaderboard, levels, activity feeds, and challenges. The evaluation of GREM is performed under a controlled experiment. GREM proved to be helpful in improving the quality, creativity, and productivity of the system. Another negative aspect of the gamified system is less collaboration and communication among stakeholders, as reported by the participants.

In another work, DMGame [69] is used for engaging users in RE and requirements prioritization. DMGame is based on Analytic Hierarchy Process (AHP) and genetic algorithms for prioritizing requirements. The system entails three user roles: supervisor, opinion provider, and negotiator, and employs game elements such as time pressure, progress, and points to facilitate collaborative requirements prioritization. Table 3 presents related studies with their game elements along with the RE phase it is used for.

Table 3 also shows commonly used game elements such as points, badges, leaderboard. Other than PBL, levels [73], ranks [70], avatar and quest [83] are also used. Literature shows that game elements in requirements elicitation are points (68%), leaderboard (50%) and badges (37%).

In literature, several approaches for identification and removal of ambiguity in requirements [2] are present. However very few platforms that involve users for reducing ambiguity in requirements [51]. Another problem with ambiguity reduction platforms is their non-interactive and unfriendly interfaces as they do not encourage user participation. This lack of user participation results in misinterpreting user requirements leading to ambiguity in requirements and hence erroneous software models [85]. User involvement in the system helps to reduce ambiguity. The motivation of this work is to propose a pro-active approach to reduce ambiguity in requirements by engaging users during requirements elicitation.

Table 3: Related studies on Gamification in RE

Source	Proposed Work	Validation	Game Elements	Phases of Requirements Engineering				
				Elicitation	Analysis	Specification	Validation	Management
2021 [70]	• A game-based language learning model for elicitation based on socio-cultural theory.	Theoretical design is validated by domain experts	Points, levels, and ranks	✓	-	-	-	-
2021 [71]	• SLR on gamification in elicitation to inquire suitable game elements for RE gamified systems. • The challenges of using gamified systems	-	Points, badges, leaderboard	✓	-	-	-	-
2020 [72]	• Reqs. discussion game RE-PROVO • Evaluate gamifying RE activities through developed game in legacy replacement to encourage innovation and minimizing unnecessary changes	Evaluation done by practitioners	Badges, points, roles, challenges	-	✓	-	-	-
2019 [73]	• A gamification approach GARUSO • Stakeholders from outside of organizations were involved in the RE process	Conducted an empirical study	Points, levels	✓	✓	✓	✓	✓
2019 [74]	• A gamified approach to make RE easier to use and learn. • Offer more games to RE activities	Under evaluation	Feedback	✓	✓	✓	✓	✓
2018 [75]	• Gamified requirement inspection • Ring-i process to allow users and other stakeholders for verification of i* models	Empirical evaluation with students	Rules, goal, and feedback system	-	✓	-	-	-
2017 [76]	• A gamified framework in reqs. analysis and designing engaging systems. • Also guide analyst in acquiring acceptance reqs.	Case study	Badges, paths, leaderboard, points, roles, avatars, rewards, challenges	✓	✓	-	-	-
2017 [69]	• DMGame to engage and motivate stakeholders in requirements prioritization. • for decision making	Scenario based	Progress, time pressure, and pointsification	-	✓	-	-	-
2017 [77]	• Crowd centered RE method to engage stakeholders in the process of requirements engineering. • With the help of REfine tool	Case study	Roles, points, leaderboards, group formation, and exploration	✓	✓	✓	✓	✓
2016 [78]	• A new method of reqs. elicitation and analysis AUCD • For user participation	Validation via projects	Points, rules	✓	✓	-	-	-
2016 [79]	• GREM, a model to engage more stakeholders. • To improve the performance of RE	Controlled experiment	PBL, levels, challenges, and activity feeds	✓	-	-	-	-
2015 [80]	• Web-based gamified platform to involve users	Case study	Leaderboard badges, points, rewards	✓	-	-	✓	-

2015 [81]	• Developed online gamified platform for scenario based RE • Selected user stories with scenarios from behavioral driven development method	Controlled experiments	Points, badges, leaderboard, levels, challenges, avatar, activity feed, progress, quiz, timer, prize etc.	✓	✓	✓	✓	-
2015 [82]	• CCRE in Software Product Organization, • Prototype tool named REfine used to involve stakeholders in RE	Demo application	Leaderboard, points, and roles	✓	✓	✓	✓	✓
2015 [68]	• REVISE, a tool for requirements elicitation and verification	-	Score, badges, and leaderboards	✓	✓	-	-	-
2012 [67]	• iThink, a game-based tool for collaboration, to gather requirements. • Creative thinking technique 'Six Thinking Hats' was also used.	Case study	Points (scoring)	✓	-	-	-	-

III. A PROACTIVE APPROACH TO REDUCE AMBIGUITY IN REQUIREMENTS

We propose a pro-active approach to reduce ambiguity in requirements during elicitation using gamification. A web-based gamified tool is designed to involve and engage users in the system during elicitation. Requirements provided in natural language are checked for ambiguity with the help of POS tags. POS checks ambiguous words in each requirement and the system prompts the user to reduce ambiguity by replacing ambiguous words with alternative unambiguous words. For each action performed during elicitation, the user is awarded with points and badges. These game elements keep the system fun to use thus by involving and engaging the user of the system and performing the desired activity i.e., elicitation and ambiguity reduction, as shown in Fig. 1.

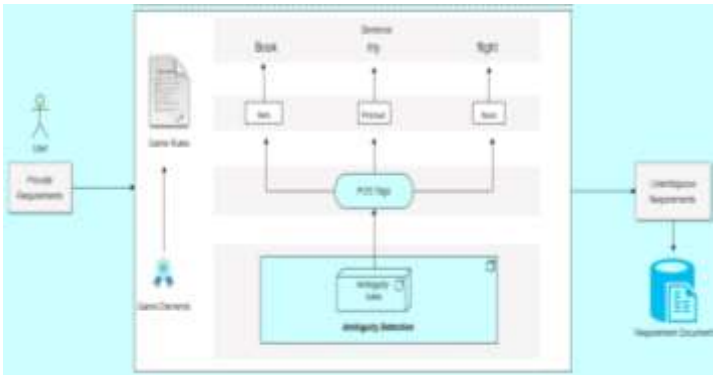

Figure 1: A Proactive Approach for Reducing Ambiguity in Requirements via Gamification

A tool is designed, comprising of four major components: user requirements, game elements and game rules, POS tagging, and ambiguity rules.

A. User Requirements

The user roles of Project Manager (PM), Requirements Engineer (ReqEngr.) and User (User/Customer/Domain Expert (DE) etc.) are involved in system. PM setups the projects and assign roles. ReqEngr. and User/Customer/DE provides the requirements. Both user roles must provide requirements, which are then checked by the system for ambiguity. In case of ambiguous requirement, user is prompted to remove the ambiguity and update the requirement. The system also provides guidelines on writing requirements. In later steps, requirements are updated, verified, and validated by the users.

B. Game Elements and Game Rules

The users involved in the RE perform a certain set of activities. Game elements are given based on the activities and design of the system. PM is managing the activity but not involved in the elicitation of requirements. The game elements assigned to PM are avatar, levels, and progress of the activity. ReqEngr. and User/Customer/DE are involved in elicitation, thus the game elements assigned to them are avatar, points, badges, levels, and leaderboard. These game elements keep the different roles engaged and motivate them to perform requirements elicitation activity.

Similarly, based on game elements and tasks, following game rules [86] for rewards and achievements, presented in Table 4, are designed for the users.

Table 4: Game Rules and Game Elements

Rules	Game Elements
If ReqEngr. and user/customer/DE etc. provide at new requirements in level 1	Points on providing each requirement, level completion badge
If ReqEngr. and user/customer/DE etc. update requirements in level 2	Points on each updated requirement, badge
If user/customer/DE etc. validate requirements in level 2	Points on providing validated requirements
If ReqEngr. verifies requirements in level 2	Points on providing verified requirements
If user/customer/DE etc. provide complete review of requirements	Points on providing complete review, Task

document in level 3	completion trophy
If ReqEngr. provide complete review of requirements document in level 3	Points on providing complete review, Task completion trophy

Table 4 presents six game rules and game elements of a gamified tool.

C. Rule-Based POS Tagging for Ambiguity Identification

The requirements provided by the user are first checked for ambiguity. The system uses POS tagging to identify ambiguity in each requirement. When a user provides NL requirement, POS assign tags to each word given in a sentence. Each word is assigned its respective POS. For example, the sentence 'Book my flight' is entered by the user. Each word is assigned its respective POS, as shown in Fig. 2.

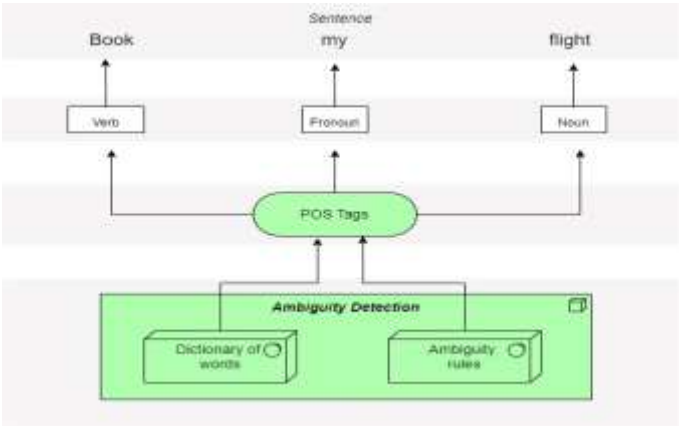

Figure 2: POS Tagging of Words

The POS tags uses collection of words containing nouns, adverbs, conjunctions, etc. and man-made rules. Ambiguity rules and collection of words work together to identify ambiguity in a sentence.

D. Ambiguity Rules

Ambiguity rules for semantic ambiguity [52] [86] are followed to identify and reduce ambiguity in requirements. The rules are designed for referential and coordination ambiguity sub-types.

Table 5: Ambiguity Rules

Ambiguity	Rules
Referential	Use of noun instead of pronoun throughout the specification, specially where two or more nouns precede the pronoun in same sentence
Coordination	Compound sentence containing two or more 'and' should be split into simpler, smaller sentences (without 'and')
	If a specification contains a conjunction ('and' or 'or') and a modifier, and modifier is used for both conjoined words, then explicitly add modifier before each word to disambiguate
	If a specification contains a conjunction ('and') and a modifier, and modifier is used for only one word, then split the sentence after 'and'

Table 5 presents ambiguity rules that combine with the POS dictionary to identify ambiguity in each requirement.

E. Gamified Tool 'Gamify4Req'

Gamify4Req is a web-based tool to elicit requirements from the user. For the development of a tool, PhP7.4 with Apache webserver is used. For identification of ambiguity, POS tagging [87] [88] is used. MySQL database is used at the backend. Gamify4Req has three tier client-server architecture having presentation layer, application layer, and data layer. The users interact with the interface, which is linked to Apache webserver, incorporated with the rules of semantic ambiguity, and connected to the database. The validity of design comprising of UML models, requirements specification guidelines, prototype, game rules and elements, is performed by the domain experts.

IV. VALIDATION OF TOOL

Gamify4Req is evaluated on two confirmatory case studies and requirements elicitation is performed by involving and engaging users. We designed the case studies using six steps, including plan, intervention, train data collectors, data collection, data analysis, and disseminate findings [89]. Firstly, we selected two IT companies. After selection of companies and cases (projects), we identified potential stakeholders i.e., PM, ReqEngr., and User/Customer/DE. Gamified tool is an intervention to collect requirements and engage users for reducing ambiguity in requirements. In the third step, we identified the data collectors in both case studies. Both data collectors are referred by the PM. They are given training on using the gamified tool, game rules, ambiguity rules, and data recording. Data collection is a significant part in which participants are asked to record their activities. Requirements document is also generated at the end of the activity. During data analysis, collected data from case studies, and user feedback is compiled for review. The results of both case studies are analyzed and compared to know the significance of our approach. We also performed statistical analysis of survey data. Lastly, the findings are reported on respective mediums.

A. Case Studies

We selected a small sized IT company consisting of nine team members in a project team for case study I. The team works with Systematic Customer Resolution Unraveling Meeting (SCRUM) methodology where team-client meetings are frequent. Requirements collection is performed during these meetings, where most clients are not clear about the system and unable to provide proper requirements. Due to uncertainty in requirements, challenges of over budget and overscheduling are commonly faced by the team. The existing method of elicitation is interviews. Interviews are conducted manually which continues for 2 months. Project (P1) is selected in case study 1 for our research. P1 is a student direction application (SDA). SDA is an interactive application for administration, students, teachers, parents, and tutors. It is a software to connect teachers and parents/tutors for discussing student's performance. Teachers add various activities and deadlines, along with tests, remarks, and homework.

Similarly, we selected a small sized IT company having seven team members in a project team for case study II. The team works in SCRUM where requirements elicitation takes 2-3 months for collection of requirements. Requirements collection is

performed in initial meetings by the developer. PM reviews the requirements. If any ambiguity occurs in requirements, the client is contacted. Mostly clients are not clear about the system, so unable to provide proper requirements. Uncertainty in requirements creates over budget and overschedule problems for the team along with a bad reputation. Project (P2) is selected in case study II. P2 '*Gotcha'- Depression Helpline'* is a mobile application that defines depression level of the user by taking an assessment quiz which is provided by the expert psychologist. Based on results, it recommends useful exercises, and monitors progress of patient overtime.

B. Results

Results from both case studies are presented in this section. We defined the following metrics given in Table 6 to compare the results of the existing approach with our gamified tool.

Table 6: Metrics for comparison

Sr. No.	Metrics
1	Total number of requirements
2	Time taken to identify each ambiguity
3	Total number of ambiguities
4	Time taken to resolve/reduce an ambiguity

The metrics presented in Table 6 relate to the objective of the study. This gamified tool is based on a pro-active approach. It is significant to measure the total number of requirements collected via tool, and ambiguities identified as compared to the existing method. It is also important to measure the time taken to resolve an ambiguity using a gamified approach.

Results from P1-SDA

In P1 interviews are used to collect requirements from the user. The responses are collected, recorded, and analyzed manually. Ambiguity identification and reduction is performed after reviewing the requirements document during the inspection phase, and discussed with the client. Almost 41 requirements are elicited from the users and product owner. Gamify4Req is used by the same project team for elicitation of requirements in SDA. Three user roles performed the activity involving PM, ReqEngr., and User/Customer/DE. Each user provided at least 15 requirements, excluding PM who, at this stage, performed tasks such as adding projects, and assigning roles. The tool takes requirements from the User/Customer/DE, ReqEngr. verified these requirements, if any requirement needed modification, it is sent back to the User/Customer/DE for the update. User/Customer/DE validated the requirements. In the last step, the requirements document is generated, and activity is closed by the PM. On each task, a reward is given in the form of points and badges, whereas tasks are divided in three different levels. Users can start their profile by creating avatars. Leaderboard is maintained for each user. PM can also view the progress of activity, and close the activity. As shown in Table 7 below.

Table 7: Ambiguity identification and reduction in SDA

No.	Requirements	P1-Existing		P1-Gamify4Req	
		Identification	Categorization	Identification	Categorization
1.	The school administration shall quickly register and login to the system	X	X	✓	✓
2.	The school administration shall enter the details of each user to register him	X	X	✓	✓
3.	Teacher shall be able to enter marks and upload the attendance of students	X	X	✓	✓
4.	Teacher shall be able to add an activity with its deadline	X	X	✓	✓
5.	Teachers shall be able to upload helping material or notes of their course for students	✓	X	✓	✓
6.	Teachers shall be able to view inquiry requests of parents about their children	X	X	✓	✓
7.	Teachers shall be able to reply to complaints, inquiries, and assistance requests through chatbox	X	X	✓	✓
8.	Teacher shall be able to view timetable to know his daily schedule	X	X	✓	✓
9.	Students shall be able to view their daily homework	X	X	✓	✓
10.	Students shall be able to view timetable to know their daily schedule	X	X	✓	✓
11.	Parents shall be able to view their daily homework	X	X	✓	✓
12.	Parents shall be able to make a complaint to teacher about their child performance	X	X	✓	✓
13.	Parents shall be able to make an inquiry request to a teacher about their child performance	X	X	✓	✓

The tool identified ambiguity given in Table 7, and displayed ambiguous words. User/Customer/DE is prompted to remove the ambiguity and enter the correct requirement. Using a gamified tool, the team provided 41 requirements, with 13 identified ambiguities. Each ambiguity took 2 seconds to identify, and 35 seconds to remove. Table 8 shows how Gamify4Req helps to reduce requirements ambiguity in less time and in a more efficient way. Table 8 shows that Gamify4Req produced better results than the existing approach used in P1-SDA.

Table 8: Comparison of results from existing approach and Gamify4Req in SDA

No.	Factors	P1-SDA-Existing Approach	P1-SDA-Gamify4Req
1	Total number of requirements	41	41
2	Total number of ambiguities	1	13
3	Time taken to detect each ambiguity	20 minutes	2 seconds
4	Time taken to resolve/reduce an ambiguity	5 minutes	35 seconds

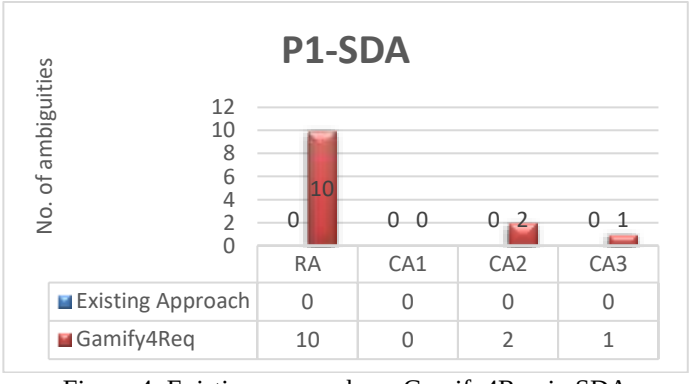

Figure 4: Existing approach vs. Gamify4Req in SDA

More ambiguities are identified and reduced using Gamify4Req. As shown in Fig. 4, in the existing approach ambiguity type is hard to find. Whereas gamified tool identifies the ambiguities and categorizes them.

Results from P2-GOTCHA

In P2, requirements elicitation is performed via interviews. The requirements are recorded manually and later documented on text editor. The customer is directly contacted through telephonic calls to resolve ambiguity. In the existing approach, the team identified ambiguities in 35 requirements. Among these ambiguities, 6 ambiguities are discarded due to their irrelevance to the scope of the tool. However, 6 ambiguities are considered relevant. Each ambiguity took 10 minutes to detect, and 7 minutes to remove, by the team. The project team used Gamify4Req for requirements elicitation of P2. The tool identified ambiguity given in Table 9, and displayed ambiguous words. The team identified 10 ambiguities in 35 requirements, in 2 seconds, and 30 seconds on reducing ambiguity in each requirement. The tool also categorizes each ambiguity according to the rules. As mentioned in table 9 below.

Table 9: Ambiguity identification and reduction in GOTCHA

No.	Requirements	P1-Existing		P1-Gamify4Req	
		Identification	Categorization	Identification	Categorization
1.	In case moderate to severe the system shall recommend exercises and therapy to patient	-	X	✓	✓
2.	The system shall perform authentication and verification of therapist	-	X	✓	✓
3.	The system shall prompt the therapist to tell his available working hours	✓	X	✓	✓
4.	The system shall show the therapist his schedule every time he logs in	✓	X	✓	✓
5.	The system shall show the therapist his schedule of pending therapy sessions that therapist need to conduct	-	-	✓	✓
6.	After the session is completed, the system shall prompt the therapist and patient to give feedback and write complain	-	X	✓	✓
7.	The system shall require the admin to log into the system to perform his duties	✓	X	✓	✓
8.	The system shall give access to admin for registered therapist and patient record	✓	X	✓	✓
9.	The system shall let the admin to view feedback and complains from patients and therapists	✓	X	✓	✓
10.	The system shall let the admin to block registered patient and therapist account after reviewing the complains	✓	X	✓	✓

Our gamified tool Gamify4Req produced better results by identifying more ambiguities as compared to the existing approach. Not only does it take less time to identify ambiguity, but also reduces ambiguity in less time, as shown in Table 10.

Table 10: Comparison of results from existing approach and Gamify4Req in GOTCHA

No.	Factors	P1-SDA-Existing Approach	P1-SDA-Gamify4Req
1	Total number of requirements	35	35
2	Total number of ambiguities	6	10
3	Time taken to	10 minutes	2 seconds

	detect each ambiguity		
4	Time taken to resolve/reduce an ambiguity	7 minutes	30 seconds

The ambiguities identified in the existing approach were out of scope of this work. Out of 11 identified ambiguities, 6 were out of scope. Whereas Gamify4Req identified 11 ambiguities in given requirements. As shown in the comparison graph below in Fig. 5.

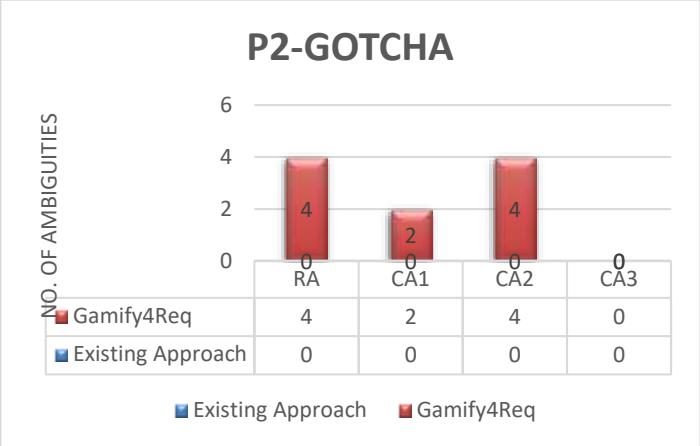

Figure 5: Existing approach vs. Gamify4Req in GOTCHA

The above graph displays ambiguity identification against each rule in existing and gamified approach of both cases. The graph shows our approach outperforms in both projects.

C. Feedback Survey

After analyzing the results from both case studies, we conducted a short survey to get feedback on user involvement in a gamified tool. Firstly, a survey is designed using google forms and communicated to the participants via email. The survey has two sections including game elements and mechanics, ambiguity identification and reduction. In the survey, users are asked to provide their feedback on mostly liked game elements while using Gamify4Req for reducing requirements ambiguity. The respondents selected the most liked game elements while using Gamify4Req. Leaderboard seems to be more significant according to 4 respondents with 66%, followed by points and badges responded by 3 and 3 respondents with 50% and 50% respectively. Levels and project progress are selected as liked element by 2, 2 respondents respectively. Similarly, respondents mentioned avatar as the least liked game element, followed by levels. The respondents also provided feedback on inclusion of game elements for reducing ambiguity. On which 4 respondents strongly agree that inclusion of game elements made system more fun to use, while 2 respondents agree to this. However, 4 respondents agree that the tool helps to identify requirements ambiguity. Next, the respondents are asked if the system takes less time to reduce the ambiguities, upon which 4 respondents strongly agrees, 1 respondent agrees, and 1 respondent remains neutral on this. The participants also highlighted three challenges of using a gamified tool. The challenges are use of the system under controlled environment with fixed and known number of active users, and user training of the system.

D. Statistical Analysis

We also identified the significant difference between groups of data. The groups are composed in a way that each group has two values to find the difference and applied ***Mann Whitney U Test.*** Mann Whitney test is used to compare two sample means belonging to same population. It is also used to check whether two sample means are equal or not. In Mann Whitney U Test, if the p value ranges between 0.01 to 0.05, there is a significant difference [90]. In our survey, groups of roles, experience, and methodology are formed. However, no significant difference using the Mann Whitney U test is found.

V. FINDINGS AND DISCUSSION

Gamify4Req is a pro-active tool which is designed and developed to collect requirements from the users and reduce ambiguity during elicitation by involving and engaging users. One important part of the system is gamification as it uses game elements and game rules to increase user participation in elicitation. For this purpose, we performed in-depth literature review and identified useful game elements being used in different phases of RE. Gamify4Req incorporates useful game elements such as avatar, PBL, levels, and progress to engage users in the activity. The findings show that among 17 game elements listed down from relevant studies in RE, users enjoy using points (68%) the most followed by leaderboard (50%) and badges (37%). The effectiveness of Gamify4Req is evaluated with two confirmatory case studies. The results are compared based on evaluation matrix. In P1 41 requirements are collected using Gamify4Req and 13 ambiguities are identified in 2 seconds on each ambiguity. Whereas, in the existing approach, the team identified only 1 ambiguity in 41 requirements spending 20 minutes on identifying ambiguity. Similarly, in P2, Gamify4Req identified 10 ambiguities from 35 requirements, spending 2 seconds on identifying each ambiguity. The feedback survey from users of Gamify4Req has shown that it not only helps to identify and reduce ambiguity but effectively involves and engages users, resulting in enhanced user participation during elicitation. Through Gamify4Req, requirements are collected, verified, updated, validated, and generated in the form of a requirements document. This suggests that the gamification approach can facilitate improving the quality of the requirements. The study also highlights the potential of gamification in RE and its ability to address the challenge of ambiguity in requirements by engaging users.

VI. CONCLUSION AND FUTURE WORK

The study proposes a novel approach for requirements ambiguity reduction during elicitation using gamification, which has not been previously explored in this context. The work involves analyzing past studies on requirements ambiguity reduction approaches, identifying game elements, designing game rules, designing a gamified tool based on ambiguity rules, and developing the tool. The validation of a tool is another significant contribution of the study, which is performed on industrial case studies. User feedback survey is also conducted, and its effectiveness is evaluated.

The gamified tool Gamify4Req aims to help teams and users to elicit NL requirements, identify and reduce ambiguity in NL requirements by using game elements. These game elements

keep the user interested in the system. According to a user feedback survey, points are the most useful game element in Gamify4Req. Users also find leaderboard and badges useful. The study also highlights the benefit of using gamification in reducing ambiguity in requirements. Not only user identified more ambiguities in given requirements but also reduced ambiguities during elicitation. However, there are certain limitations to the study that need to be acknowledged. Firstly, the novelty of using gamification for reducing ambiguity may require more effort to convince IT companies to invest their time in using the tool. Secondly, the cost factor associated with the tool may pose a challenge for small-sized IT companies. In future studies, gamification can be explored to other types of ambiguities in requirements. The scope of the approach can be expanded to mid- and large-sale projects.

REFERENCES

[1] V. Gervasi and D. Zowghi, "Reasoning about inconsistencies in natural language requirements," *ACM Trans. Softw. Eng. Methodol.*, vol. 14, no. 3, pp. 277–330, Jul. 2005, doi: 10.1145/1072997.1072999.

[2] M. Bano, "Addressing the challenges of requirements ambiguity: A review of empirical literature," in *2015 IEEE Fifth International Workshop on Empirical Requirements Engineering (EmpiRE)*, Ottawa, ON, Canada: IEEE, Aug. 2015, pp. 21–24. doi: 10.1109/EmpiRE.2015.7431303.

[3] A. V. Lamsweerde, "Formal Specification: a Roadmap," *ACM*, 2000.

[4] E. Kamsties, D. M. Berry, and B. Paech, "Detecting Ambiguities in Requirements Documents Using Inspections." 2001.

[5] D. M. Berry, "Ambiguity in Natural Language Requirements Documents," in *Innovations for Requirement Analysis. From Stakeholders' Needs to Formal Designs*, B. Paech and C. Martell, Eds., in Lecture Notes in Computer Science, vol. 5320. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 1–7. doi: 10.1007/978-3-540-89778-1_1.

[6] D. M. Berry and E. Kamsties, "Ambiguity in Requirements Specification," in *Perspectives on Software Requirements*, J. C. S. Prado Leite and J. H. Doorn, Eds., Boston, MA: Springer US, 2004, pp. 7–44. doi: 10.1007/978-1-4615-0465-8_2.

[7] G. Sandhu and S. Sikka, "State-of-art practices to detect Inconsistencies and Ambiguities from Software Requirements," presented at the International Conference on Computing, Communication and Automation, India, India: IEEE, 2015, pp. 812–817.

[8] M. Q. Riaz, W. H. Butt, and S. Rehman, "Automatic Detection of Ambiguous Software Requirements: An Insight," in *2019 5th International Conference on Information Management (ICIM)*, Cambridge, United Kingdom: IEEE, Mar. 2019, pp. 1–6. doi: 10.1109/INFOMAN.2019.8714682.

[9] A. O. J. Sabriye and W. M. N. W. Zainon, "A framework for detecting ambiguity in software requirement specification," in *2017 8th International Conference on Information Technology (ICIT)*, Amman, Jordan: IEEE, May 2017, pp. 209–213. doi: 10.1109/ICITECH.2017.8080002.

[10] A. Ferrari, G. Lipari, S. Gnesi, and G. O. Spagnolo, "Pragmatic ambiguity detection in natural language requirements," in *2014 IEEE 1st International Workshop on Artificial Intelligence for Requirements Engineering (AIRE)*, Karlskrona: IEEE, Aug. 2014, pp. 1–8. doi: 10.1109/AIRE.2014.6894849.

[11] M. C. MacDonald, "Probabilistic constraints and syntactic ambiguity resolution," *Language and Cognitive Processes*, vol. 9, no. 2, pp. 157–201, May 1994, doi: 10.1080/01690969408402115.

[12] H. Kamp and U. Reyle, From discourse to logic: introduction to modeltheoretic semantics of natural language, formal logic and discourse representation theory, Reprinted. in Studies in linguistics and philosophy, no. 42. Dordrecht: Springer-Science+Business Media, B.V, 1993.

[13] A. Ferrari, P. Spoletini, and S. Gnesi, "Ambiguity Cues in Requirements Elicitation Interviews," in *2016 IEEE 24th International Requirements Engineering Conference (RE)*, Beijing, China: IEEE, Sep. 2016, pp. 56–65. doi: 10.1109/RE.2016.25.

[14] U. S. Shah and D. C. Jinwala, "Resolving Ambiguities in Natural Language Software Requirements: A Comprehensive Survey," *SIGSOFT Softw. Eng. Notes*, vol. 40, no. 5, pp. 1–7, Sep. 2015, doi: 10.1145/2815021.2815032.

[15] S. F. Tjong and D. M. Berry, "The Design of SREE — A Prototype Potential Ambiguity Finder for Requirements Specifications and Lessons Learned," in *Requirements Engineering: Foundation for Software Quality*, J. Doerr and A. L. Opdahl, Eds., in Lecture Notes in Computer Science, vol. 7830. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 80–95. doi: 10.1007/978-3-642-37422-7_6.

[16] I. S. Bajwa, M. Lee, and B. Bordbar, "Resolving Syntactic Ambiguities in Natural Language Specification of Constraints," in *Computational Linguistics and Intelligent Text Processing*, A. Gelbukh, Ed., in Lecture Notes in Computer Science, vol. 7181. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 178–187. doi: 10.1007/978-3-642-28604-9_15.

[17] H. Maria and Z. Ali, "Requirement Elicitation Techniques for Open Source Systems: A Review," *ijacsa*, vol. 9, no. 1, 2018, doi: 10.14569/IJACSA.2018.090145.

[18] T. Iqbal, "Requirement Elicitation Technique: - A Review Paper," *IJCMS*, vol. 3, no. 9, Nov. 2014.

[19] T. ur Rehman, M. N. A. Khan, and N. Riaz, "Analysis of Requirement Engineering Processes, Tools/Techniques and Methodologies," *IJITCS*, vol. 5, no. 3, pp. 40–48, Feb. 2013, doi: 10.5815/ijitcs.2013.03.05.

[20] Z. Zhang, "Effective Requirements Development - A Comparison of Requirements Elicitation techniques," in *Software Quality Management XV*, 2007, pp. 225–240.

[21] W. H. B. Wan Hassim, "A Review on Effective Requirement Elicitation Techniques," *International Journal of Advances in Computer Science and Technology*, vol. 6, no. 1, pp. 4–8, 2017.

[22] M. G. Christel and K. C. Kang, "Issues in Requirements Elicitation," Software Engineering Institute, Pittsburgh, Pennsylvania, Technical ESC-TR-92-012, 1992.

[23] B. Metrics, "The Impact of Incomplete or Changing Requirements on IT Project Success," *The Impact of Incomplete or Changing Requirements on IT Project Success*, 2013. https://www.boardroommetrics.com/blog/the-impact-of-incomplete-or-changing-requirements-on-it-project-success-20131222.htm

[24] M. Sadiq and S. K. Jain, "An Insight into Requirements Engineering Processes," in *Advances in Communication, Network, and Computing*, V. V. Das and J. Stephen, Eds., in Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 108. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 313–318. doi: 10.1007/978-3-642-35615-5_48.

[25] T. Iqbal, N. Seyff, and D. M. Fernández, "Generating Requirements Out of Thin Air: Towards Automate Feature Identification for New Apps." arXiv, Sep. 25, 2019. Accessed: Nov. 26, 2022. [Online]. Available: http://arxiv.org/abs/1909.11302

[26] H. S. Dar, "Reducing Ambiguity in Requirements Elicitation via Gamification," in *2020 IEEE 28th International Requirements Engineering Conference (RE)*, Zurich, Switzerland: IEEE, Aug. 2020, pp. 440–444. doi: 10.1109/RE48521.2020.00065.

[27] M. A. Abbasi, J. Jabeen, Y. Hafeez, D.-B. Batool, and N. Fareen, "Assessment of Requirement Elicitation Tools and Techniques by Various Parameters," *Software Engineering*, vol. 3, no. 2, pp. 7–11, 2015, doi: 10.11648/j.se.20150302.11.

[28] S. Tariq, A. Ibrahim, A. Usama, and M. S. Shahbaz, "An Overview of Requirements Elicitation Techniques in Software Engineering with a focus on Agile Development," in *2021 4th International Conference on Computing & Information Sciences (ICCIS)*, Karachi, Pakistan: IEEE, Nov. 2021, pp. 1–6. doi: 10.1109/ICCIS54243.2021.9676192.

[29] B. Mohy-ud-din, M. S. Arshad Malik, M. Awaiz, and A. Shahid, "A Systematic Report on Issue and Challenges during Requirement Elicitation," *IJACSA*, vol. 8, no. 11, pp. 376–379, 2017.

[30] N. Rahim, A. W. Khan, K. Sardaraz, and F. Khan, "A Protocol for Requirements Elicitation Challenges Model," *Science, Technology and Development*, vol. 37, no. 3, pp. 149–157, 2018, doi: 10.3923/std.2018.149.157.

[31] K. Dikert, M. Paasivaara, and C. Lassenius, "Challenges and success factors for large-scale agile transformations: A systematic literature review," *Journal of Systems and Software*, vol. 119, pp. 87–108, Sep. 2016, doi: 10.1016/j.jss.2016.06.013.

[32] J. Passnen, "Enhancing requirements elicitation and validation with gamification. Requirements engineering in combination with gamification using action research," University of Gothenburg, Sweden, 2015.

[33] P. Lombriser, F. Dalpiaz, G. Lucassen, and S. Brinkkemper, "Gamified Requirements Engineering: Model and Experimentation," in *Requirements Engineering: Foundation for Software Quality*, M. Daneva and O. Pastor, Eds., in Lecture Notes in Computer Science, vol. 9619. Cham: Springer International Publishing, 2016, pp. 171–187. doi: 10.1007/978-3-319-30282-9_12.

[34] S. Deterding, M. Sicart, L. Nacke, K. O'Hara, and D. Dixon, "Gamification. using game-design elements in non-gaming contexts," in *CHI '11 Extended Abstracts on Human Factors in Computing Systems*, Vancouver BC Canada: ACM, May 2011, pp. 2425–2428. doi: 10.1145/1979742.1979575.

[35] A. Mora, G. F. Tondello, L. E. Nacke, and J. Arnedo-Moreno, "Effect of personalized gameful design on student engagement," in *2018 IEEE Global Engineering Education Conference (EDUCON)*, Tenerife: IEEE, Apr. 2018, pp. 1925–1933. doi: 10.1109/EDUCON.2018.8363471.

[36] F. Faiella and M. Ricciardi, "Gamification and learning: a review of issues and research," *Journal of e-Learning and Knowledge Society*, vol. Vol 11, p. No 3 (2015): Gamification and Serious Game for Learning, Sep. 2015, doi: 10.20368/1971-8829/1072.

[37] P. Lombriser and R. van der Valk, "Improving the Quality of the Software Development Lifecycle with Gamification," in *COSIT11 Proceedings of the 10th international conference on Spatial information theory*, USA: Springer, 2011, p. 1.

[38] O. Pedreira, F. García, N. Brisaboa, and M. Piattini, "Gamification in software engineering – A systematic mapping," *Information and Software Technology*, vol. 57, pp. 157–168, Jan. 2015, doi: 10.1016/j.infsof.2014.08.007.

[39] R. M. Parizi, "On the gamification of human-centric traceability tasks in software testing and coding," in *2016 IEEE 14th International Conference on Software Engineering Research, Management and Applications (SERA)*, Towson, MD, USA: IEEE, Jun. 2016, pp. 193–200. doi: 10.1109/SERA.2016.7516146.

[40] J. Juul, "The Game, the Player, the World: Looking for a Heart of Gameness," in *Level Up: Digital Games Research Conference Proceedings*, Utrecht: Utrecht University: Utrecht University, 2003, pp. 30–45.

[41] P. Lombriser and F. Dalpiaz, "Engaging stakeholders in scenario-based requirements engineering with gamification," 2015.

[42] N. Unkelos-Shpigel and I. Hadar, "Inviting everyone to play: Gamifying collaborative requirements engineering," in *2015 IEEE Fifth International Workshop on Empirical Requirements Engineering (EmpiRE)*, Ottawa, ON, Canada: IEEE, Aug. 2015, pp. 13–16. doi: 10.1109/EmpiRE.2015.7431301.

[43] R. Cursino, D. Ferreira, M. Lencastre, R. Fagundes, and J. Pimentel, "Gamification in Requirements Engineering: A Systematic Review," in *2018 11th International Conference on the Quality of Information and Communications Technology (QUATIC)*, Coimbra: IEEE, Sep. 2018, pp. 119–125. doi: 10.1109/QUATIC.2018.00025.

[44] A. Umber and I. S. Bajwa, "Minimizing ambiguity in natural language software requirements specification," in *2011 Sixth International Conference on Digital Information Management*, Melbourne, Australia: IEEE, Sep. 2011, pp. 102–107. doi: 10.1109/ICDIM.2011.6093363.

[45] D. Popescu, S. Rugaber, N. Medvidovic, and D. M. Berry, "Reducing Ambiguities in Requirements Specifications Via Automatically Created Object-Oriented Models," in *Innovations for Requirement Analysis. From Stakeholders' Needs to Formal Designs*, B. Paech and C. Martell, Eds., in Lecture Notes in Computer Science, vol. 5320. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 103–124. doi: 10.1007/978-3-540-89778-1_10.

[46] S. B. Preston, "Reducing Ambiguities in Customer Requirements Through Historical Rule-Based Knowledge in a Small Organization." University of Southern Mississippi, 2014. [Online]. Available: https://aquila.usm.edu/dissertations/251

[47] S. Sinha and M. S. Husain, "Proposal for Avoiding Ambiguity in Requirement Engineering using Artificial Intelligence," in *ACEIT Conference Proceeding*, 2016.

[48] T. J. Vimalraj and B. Seema, "Identification of Ambiguity in Requirement Specification using Multilingual Word Sense," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 5, no. 6, Jun. 2016, doi: 10.17148/IJARCCE.2016.5681.

[49] A. Umber and I. S. Bajwa, "A Step Towards Ambiguity Less Natural Language Software Requirements Specifications," *Int. J. Web Appl.*, vol. 4, no. 1, 2012.

[50] Neumann, "Only his only grammarian can only say only what only he only means," *ACM*, vol. 9, no. 1, p. 6, 1986.

[51] H. Yang, A. de Roeck, V. Gervasi, A. Willis, and B. Nuseibeh, "Analysing anaphoric ambiguity in natural language requirements," *Requirements Eng*, vol. 16, no. 3, pp. 163–189, Sep. 2011, doi: 10.1007/s00766-011-0119-y.

[52] S. Chaudhry and S. Imtiaz, "Rule-Based Approach to Reduce Lexical & Semantic Ambiguity in Software Requirements," International Islamic University Islamabad, Islamabad Pakistan, 2020.

[53] S. Osama and M. Aref, "DETECTING AND RESOLVING AMBIGUITY APPROACH IN REQUIREMENT SPECIFICATION: IMPLEMENTATION, RESULTS AND EVALUATION," *International Journal of Intelligent Computing and Information Science*, vol. 18, no. 1, pp. 27–36, 2018.

[54] A. Alzayed and A. Al-Hunaiyyan, "A Bird's Eye View of Natural Language Processing and Requirements Engineering," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 5, 2021.

[55] A. Bajceta, M. Leon, W. Afzal, P. Lindberg, and M. Bohlin, "Using NLP Tools to Detect Ambiguities in System Requirements - A Comparison Study," in *NLP4RE 2022: 5th Workshop on Natural Language Processing for Requirements Engineering*, Birmingham, UK: CEUR Workshop Proceedings, 2022.

[56] S. Ezzini, S. Abualhaija, C. Arora, and M. Sabetzadeh, "Automated handling of anaphoric ambiguity in requirements: a multi-solution study," in *Proceedings of the 44th International Conference on Software Engineering*, Pittsburgh Pennsylvania: ACM, May 2022, pp. 187–199. doi: 10.1145/3510003.3510157.

[57] S. Ezzini, S. Abualhaija, C. Arora, and M. Sabetzadeh, "TAPHSIR: towards AnaPHoric ambiguity detection and ReSolution in requirements," in *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, Singapore Singapore: ACM, Nov. 2022, pp. 1677–1681. doi: 10.1145/3540250.3558928.

[58] A. R. Amna, "Identifying Ambiguity Problems in User Stories : A Proposed Framework," presented at the CEUR Workshop Proceedings, Belgium, Belgium: CEUR Workshop Proceedings, 2022.

[59] F. Ashfaq, I. S. Bajwa, R. Kazmi, A. Khan, and M. Ilyas, "An Intelligent Analytics Approach to Minimize Complexity in Ambiguous Software Requirements," *Scientific Programming*, vol. 2021, pp. 1–20, Mar. 2021, doi: 10.1155/2021/6616564.

[60] F. Ashfaq and I. S. Bajwa, "Natural language ambiguity resolution by intelligent semantic annotation of software requirements," *Autom Softw Eng*, vol. 28, no. 2, p. 13, Nov. 2021, doi: 10.1007/s10515-021-00291-0.

[61] A. K. G. Gupta and A. Deraman, "A framework for software requirement ambiguity avoidance," *IJECE*, vol. 9, no. 6, p. 5436, Dec. 2019, doi: 10.11591/ijece.v9i6.pp5436-5445.

[62] F. Dalpiaz, I. van der Schalk, and G. Lucassen, "Pinpointing Ambiguity and Incompleteness in Requirements Engineering via Information Visualization and NLP," in *Requirements Engineering: Foundation for Software Quality*, E. Kamsties, J. Horkoff, and F. Dalpiaz, Eds., in Lecture Notes in Computer Science, vol. 10753. Cham: Springer International Publishing, 2018, pp. 119–135. doi: 10.1007/978-3-319-77243-1_8.

[63] F. S. B¨aumer and M. Geierhos, "Flexible Ambiguity Resolution and Incompleteness Detection in Requirements Descriptions via an Indicator-Based Configuration of Text Analysis Pipelines," in *Proceedings of the 51st Hawaii International Conference on System Sciences*, University of Hawai', 2018.

[64] A. Ferrari, P. Spoletini, and S. Gnesi, "Ambiguity and tacit knowledge in requirements elicitation interviews," *Requirements Eng*, vol. 21, no. 3, pp. 333–355, Sep. 2016, doi: 10.1007/s00766-016-0249-3.

[65] H. Yang, A. Willis, A. De Roeck, and B. Nuseibeh, "Automatic detection of nocuous coordination ambiguities in natural language requirements," in *Proceedings of the IEEE/ACM international conference on Automated software engineering - ASE '10*, Antwerp, Belgium: ACM Press, 2010, p. 53. doi: 10.1145/1858996.1859007.

[66] R. Beg, Q. Abbas, and A. Joshi, "A Method to Deal with the Type of Lexical Ambiguity in a Software Requirement Specification Document," in *2008 First International Conference on Emerging Trends in Engineering*

and Technology, Nagpur, Maharashtra, India: IEEE, 2008, pp. 1212–1215. doi: 10.1109/ICETET.2008.160.

[67] S. F. Tjong, "Avoiding Ambiguity in Requirements Specifications," University of Nottingham, Malaysia, 2008.

[68] V. Gervasi, A. Ferrari, D. Zowghi, and P. Spoletini, "Ambiguity in requirements engineering: towards a unifying framework.," in *From Software Engineering to Formal Methods and Tools, and Back.*, Springer, Cham, 2019, pp. 191–210.

[69] A. Ferrari and A. Esuli, "An NLP approach for cross-domain ambiguity detection in requirements engineering," *Autom Softw Eng*, vol. 26, no. 3, pp. 559–598, Sep. 2019, doi: 10.1007/s10515-019-00261-7.

[70] J. Fernandes, D. Duarte, C. Ribeiro, C. Farinha, J. M. Pereira, and M. M. da Silva, "iThink: A Game-Based Approach Towards Improving Collaboration and Participation in Requirement Elicitation," *Procedia Computer Science*, vol. 15, pp. 66–77, 2012, doi: 10.1016/j.procs.2012.10.059.

[71] F. Kifetew, D. Munante, A. Perini, A. Susi, A. Siena, and P. Busetta, "DMGame: A Gamified Collaborative Requirements Prioritisation Tool," in *2017 IEEE 25th International Requirements Engineering Conference (RE)*, Lisbon, Portugal: IEEE, Sep. 2017, pp. 468–469. doi: 10.1109/RE.2017.46.

[72] K. Ishaq, F. Rosdi, N. A. M. Zin, and A. Abid, "Requirements Elicitation for Game-based Language Learning Application," in *2021 International Conference on Innovative Computing (ICIC)*, Lahore, Pakistan: IEEE, Nov. 2021, pp. 1–9. doi: 10.1109/ICIC53490.2021.9693004.

[73] S. Gul *et al.*, "Gamification and Gaming Elements for Software Requirements Elicitation: A Systematic Literature Review," *ijeei*, vol. 13, no. 4, pp. 932–950, Dec. 2021, doi: 10.15676/ijeei.2021.13.4.11.

[74] A. Alexandrova and L. Rapanotti, "Requirements analysis gamification in legacy system replacement projects," *Requirements Eng*, vol. 25, no. 2, pp. 131–151, Jun. 2020, doi: 10.1007/s00766-019-00311-2.

[75] M. Z. Kolpondinos and M. Glinz, "GARUSO: a gamification approach for involving stakeholders outside organizational reach in requirements engineering," *Requirements Eng*, vol. 25, no. 2, pp. 185–212, Jun. 2020, doi: 10.1007/s00766-019-00314-z.

[76] M. E. A. TEBIB, "Gamifying Requirements Engineering for Better Practice," in *Joint Proceedings of REFSQ-2019 Workshops, Doctoral Symposium*, Germany: CEUR, 2019. [Online]. Available: https://ceur-ws.org/Vol-2376/DS-paper3.pdf

[77] J. Pimentel, E. Santos, T. Pereira, D. Ferreira, and J. Castro, "A gamified requirements inspection process for goal models," in *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*, Pau France: ACM, Apr. 2018, pp. 1300–1307. doi: 10.1145/3167132.3167272.

[78] L. Piras, E. Paja, P. Giorgini, and J. Mylopoulos, "Goal Models for Acceptance Requirements Analysis and Gamification Design," in *Conceptual Modeling*, H. C. Mayr, G. Guizzardi, H. Ma, and O. Pastor, Eds., in Lecture Notes in Computer Science, vol. 10650. Cham: Springer International Publishing, 2017, pp. 223–230. doi: 10.1007/978-3-319-69904-2_18.

[79] F. Dalpiaz, R. Snijders, S. Brinkkemper, M. Hosseini, A. Shahri, and R. Ali, "Engaging the Crowd of Stakeholders in Requirements Engineering via Gamification," in *Gamification*, S. Stieglitz, C. Lattemann, S. Robra-Bissantz, R. Zarnekow, and T. Brockmann, Eds., in Progress in IS. Cham: Springer International Publishing, 2017, pp. 123–135. doi: 10.1007/978-3-319-45557-0_9.

[80] B. S. Kumar and I. Krishnamurthi, "Improving User Participation in Requirement Elicitation and Analysis by Applying Gamification Using Architect's Use Case Diagram," in *Proceedings of the 3rd International Symposium on Big Data and Cloud Computing Challenges (ISBCC – 16')*, V. Vijayakumar and V. Neelanarayanan, Eds., in Smart Innovation, Systems and Technologies, vol. 49. Cham: Springer International Publishing, 2016, pp. 471–482. doi: 10.1007/978-3-319-30348-2_39.

[81] J. Pasanen, *Enhancing Requirements Elicitation and Validation with Gamification*. University of Gothenburg Chalmers University of Technology, 2016.

[82] R. Snijders, F. Dalpiaz, S. Brinkkemper, M. Hosseini, R. Ali, and A. Ozum, "REfine: A gamified platform for participatory requirements engineering," in *2015 IEEE 1st International Workshop on Crowd-Based Requirements Engineering (CrowdRE)*, Ottawa, ON, Canada: IEEE, Aug. 2015, pp. 1–6. doi: 10.1109/CrowdRE.2015.7367581.

[83] P. Lombriser and F. Dalpiaz, "Engaging Stakeholders in Scenario-Based Requirements Engineering with Gami cation," Utrecht University - Department of Information and Computing Sciences, Netherland, 2015.

[84] R. Snijders, A. Ozum, S. Brinkkemper, and F. Dalpiaz, "Crowd-Centric Requirements Engineering: A method based on crowdsourcing and gamification," Netherland, Technical UU-CS-2015-004, 2015.

[85] N. Masoudi *et al.*, "Elicitation, Computational Representation, and Analysis of Mission and System Requirements," presented at the WCX SAE World Congress Experience, Mar. 2022, pp. 2022-01–0363. doi: 10.4271/2022-01-0363.

[86] H. S. Dar, S. Imtiaz, and M. I. Ullah Lali, "Gamification Tool Design for Reducing Requirements Ambiguity during Elicitation," in *2022 5th International Conference on Computing and Informatics (ICCI)*, New Cairo, Cairo, Egypt: IEEE, Mar. 2022, pp. 080–086. doi: 10.1109/ICCI54321.2022.9756083.

[87] S. G. Kanakaraddi and S. S. Nandyal, "Survey on Parts of Speech Tagger Techniques," in *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT)*, Coimbatore: IEEE, Mar. 2018, pp. 1–6. doi: 10.1109/ICCTCT.2018.8550884.

[88] X. Han, Y. Dang, L. Mei, Y. Wang, S. Li, and X. Zhou, "A Novel Part of Speech Tagging Framework for NLP Based Business Process Management," in *2019 IEEE International Conference on Web Services (ICWS)*, Milan, Italy: IEEE, Jul. 2019, pp. 383–387. doi: 10.1109/ICWS.2019.00068.

[89] P. Neale, S. Thapa, and C. Boyce, "PREPARING A CASE STUDY: A Guide for Designing and Conducting a Case Study for Evaluation Input." Pathfinder International, 2006. [Online]. Available: https://resources.peopleinneed.net/documents/102-m-e-tool-series-case-study.pdf

[90] A. Buthmann, "Making Sense of Mann-Whitney Test for Median Comparison," *I Six Sigma*, 2010. https://www.isixsigma.com/hypothesis-testing/making-sense-mann-whitney-test-median-comparison/#:~:text=Because%20the%20assumptions%20are%20now,is%200.0459%20%E2%80%93%20less%20than%200.05 .

[91] H. Poor, *An Introduction to Signal Detection and Estimation*. New York: Springer-Verlag, 1985, ch. 4.

[92] B. Smith, "An approach to graphs of linear forms (Unpublished work style)," unpublished.

[93] E. H. Miller, "A note on reflector arrays (Periodical style—Accepted for publication)," *IEEE Trans. Antennas Propagat.*, to be published.

[94] J. Wang, "Fundamentals of erbium-doped fiber amplifiers arrays (Periodical style—Submitted for publication)," *IEEE J. Quantum Electron.*, submitted for publication.

AUTHORS

First Author – Hafsa Shareef Dar, PhD Student at International Islamic University Islamabad, Lecturer at University of Gujrat Pakistan.

Second Author - Salma Imtiaz – PhD, Assistant Professor and Head of Department (Software Engineering) at International Islamic University Islamabad, Pakistan

Third Author – Muhammad Ikramullah Lali – PhD, Professor, University of Education Lahore, Pakistan,

Correspondence Author – Hafsa Shareef Dar,