# Reducing Ambiguity in Requirements Elicitation via Gamification


**Ph.D Thesis**

*By*

**HAFSA SHAREEF DAR**

28-FBAS/PHDSE/F16

*Supervised by:*

**Dr. Salma Imtiaz**
**Assistant Professor**
Department of Software Engineering
Faculty of Computing and Information Technology
International Islamic University Islamabad

*Co-Supervised by:*

**Prof. Dr. Muhammad Ikramullah Lali**
**Professor**
University of Education, Lahore

**DEPARTMENT OF SOFTWARE ENGINEERING**
**FACULTY OF COMPUTING AND INFORMATION TECHNOLOGY**
**INTERNATIONAL ISLAMIC UNIVERSITY, ISLAMABAD, PAKISTAN**
**2024**

P.P.L.
ocs. 1
HAR

A Dissertation submitted to the

**Department of Software Engineering**

International Islamic University, Islamabad

as a partial fulfilment of requirements for the award of

the degree of

**Doctor of Philosophy in Software Engineering**

**Department of Software Engineering,**

**International Islamic University, Islamabad**

## Final Approval

It is certified that we have examined the thesis report 'Reducing Ambiguity in Requirements Elicitation via Gamification' submitted by Ms. Hafsa Shareef Dar, Registration No. 28-FBAS/PHDSE/F16, and it is our judgment that this thesis is of sufficient standard to warrant its acceptance by the International Islamic University, Islamabad for the Doctor of Philosophy in Software Engineering.

**EXTERNAL EXAMINERS**

Dr Saad Zafar
Professor,
Riphah International University
Islamabad

Dr. Aamer Nadeem
Professor,
Department of Software Engineering
Capital University of Science and Technology
Islamabad

**INTERNAL EXAMINER**

Dr Muhammad Nasir
Lecturer,
Department of Software Engineering
Faculty of Computing & Information Technology
IIU, Islamabad

**DEAN**

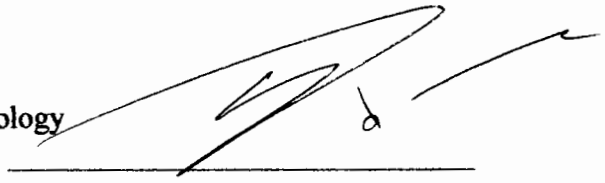
Dr. Mohammad Asmat Ullah Khan
Professor,
Faculty of Computing & Information Technology
IIU, Islamabad

**CHAIRPERSON**

Dr. Salma Imtiaz
Assistant Professor,
Department of Software Engineering
Faculty of Computing & Information Technology
IIU, Islamabad

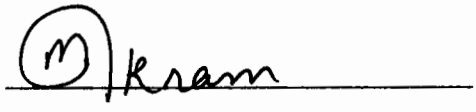
**SUPERVISOR**

Dr. Salma Imtiaz
Assistant Professor,
Department of Software Engineering,
Faculty of Computing & Information Technology
IIU, Islamabad



**CO-SUPERVISOR**

Dr. Muhammad Ikramullah Lali
Professor,
Department of Information Sciences
University of Education, Lahore



# Declaration

I hereby declare that this thesis, neither as a whole nor as a part thereof has been copied out from any source. It is further declared that no portion of the work presented in this report has been submitted in support of any application for any other degree or qualification of this or any other university or institute of learning.

**Hafsa Shareef Dar**
**28-FBAS/PHDSE/F16**

## Dedicated To

*My dear parents, beloved son, and wonderful siblings*

**Hafsa**

# Acknowledgments

I am deeply grateful to Almighty **ALLAH** for his favours and blessings, that encouraged me to write this dissertation. I owe thanks to **ALLAH** for everything.

This thesis would not have been possible without the inspiration and support of many wonderful individuals; deepest thanks and appreciation to all of them for being part of this journey and making this accomplishment possible.

Foremost, I would like to express my sincere gratitude to my supervisor Dr. Salma Imtiaz (Assistant Professor, International Islamic University Islamabad) for her continuous guidance, endless patience, and untiring efforts in refining my work. I would also like to express heartfelt gratitude to my co-supervisor Prof. Dr. Muhammad Ikramullah Lali (Professor, University of Education) for his support since the initial stages of the course till now, and for having faith in me. I would also like to thank other respected faculty members of the Department of Software Engineering for their help and guidance. I also want to acknowledge the support provided by the administrative staff of the Department of Software Engineering.

I am thankful to my friends who have helped me, in any way, during my PhD. The list includes Mr. Luca Piras, Ms. Naomi, Mr. Shahzad, Mr. Zulfiqar, Mr. Ahsan, Mr. Waqas, Mr. Abdul Rahman, Mr. Zeeshan, Ms. Shamaila, Ms. Safia, Ms. Laraib, Dr. Zakia and Dr. Javed.

A special thanks to my mother for believing in me, to my son for being in my life, and to my siblings for standing by my side. I appreciate my extended family especially Prof. Dr. Faisal Saud Dar, and my aunt, for their patience, encouragement, financial, and moral support during this long journey. I owe thanks to all of them who are mentioned here, and to all those who are missed unintentionally.

# ABSTRACT

**Context:** Requirements ambiguity in Natural Language stems from uncertain and unclear requirements that often occur due to lack of user involvement and diverse background among technical teams, users, and domain experts. Though formal methods are also used to mitigate ambiguity, but they demand significant expertise and resources.

**Existing Work and Limitations:** Traditionally, requirements are collected using requirements elicitation techniques such as interviews, surveys, and focus groups etc. The users may also find it difficult to describe the requirements due to unclear system functions and purpose. They may not have a clear idea of their requirements. Moreover, lack of user involvement leads to misinterpretation of requirements during specifications, which also results in ambiguous requirements. The existing approaches to eliminate ambiguity are reactive resulting in rework and increased effort to detect and remove ambiguity. Due to the technical aspects of the process and long meeting sessions, users often feel reluctant to get involved in the activity of reducing ambiguity during elicitation.

**Objective:** The main objective of this study is to reduce requirements ambiguity by involving users in the elicitation activity. Also, to elicit requirements keeping in view the ambiguity rules.

**Methodology:** Firstly, we conducted a Systematic Literature Review (SLR) to empirically validate the game elements identified in the initial literature review. The study designs and develops a gamification tool Gamify4Req that encourages user involvement during requirements elicitation to reduce requirements ambiguity. A tool is developed to reduce requirements ambiguity based on game elements and game rules. A gamified tool (Gamify4Req) incorporates ambiguity rules, game rules, and game elements such as avatar, points, badges, leaderboard, levels, and progress to elicit requirements. The tool uses Rule-Based Parts of Speech (POS) tags that uses ambiguity rules for identification of ambiguity in user-provided requirements. Upon the completion of each task performed by the user, points and badges are given. These awards motivate the user to participate and engage in

elicitation and help to reduce ambiguity in requirements. The validation of the tool is performed on two confirmatory case studies taken from the software industry.

**Results:** The gamified tool produces better results compared to those of existing approaches used in the software industry. We found that the most useful game elements are points (68%) followed by leaderboard (50%) and badges (37%) in requirements elicitation. For coverage of game elements other than the PBL, SLR is conducted. As an outcome, 16 useful game elements including levels (61%), feedback (44%), and progress (38%) are identified. The effectiveness of Gamify4Req is evaluated on two case studies. In Project 1 total 41 requirements are collected. Gamify4Req identified 12 ambiguities, whereas existing approach identified only 1 ambiguity in 41 requirements. Similarly, in Project 2, 35 requirements are collected. Gamify4Req identified 10 ambiguities i.e., 2 seconds on identifying each ambiguity. Previous approach took 20 minutes to identify ambiguity in each requirement. The response from user feedback survey shows that user involvement via Gamify4Req results helps in timely identification and reduction of ambiguity in requirements. The gamified tool reduces ambiguity in requirements by involving users in elicitation and making elicitation more exciting and fun.

**Conclusion:** Ambiguity in requirements is a prominent challenge in NL requirements. It becomes more crucial when user does not participate in elicitation and analyst specify requirements without user involvement. The gamified tool Gamify4Req helps the team to elicit ambiguity free requirements from the user. The game elements keep the interest of user in the system. In this way user not only identifies more ambiguities but also helps to reduce ambiguities during elicitation. Furthermore, this study also provides guidelines to practitioners and researchers on design, development, and use of gamified tool.

**Key Words:** Requirements Elicitation, Requirements Ambiguity, Ambiguity Rules, Ambiguity Identification, Ambiguity Reduction, Gamification, Game Rules, Game Elements, Parts of Speech Tags

# Publications

**Chapter 1:** ***H. S. Dar***, "Reducing Ambiguity in Requirements Elicitation via Gamification," *2020 IEEE 28th International Requirements Engineering Conference (RE)*, Zurich, Switzerland, 2020, pp. 440-444, DOI: 10.1109/RE48521.2020.00065. *[Ranking A]*

**Chapter 2:** ***H. S. Dar***, S. Imtiaz, and M. I. Ullah Lali. "Reducing Requirements Ambiguity via Gamification: Comparison with Traditional Techniques." *Computational Intelligence and Neuroscience* 2022 (2022). *[IF 3.41]*

**Chapter 4:** S. Gul, ***H. S. Dar.,*** et al. "Gamification and gaming elements for software requirements elicitation: a systematic literature review." *International Journal on Electrical Engineering and Informatics* 13.4 (2021): 931-950. *[IF 1.44]*

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

**Chapter 6 & 7:** ***H. S. Dar***, S. Imtiaz, M. I. Ullah Lali, R. Aziz, and J. A. Khan, "Identifying and Reducing Ambiguities in Natural Language Requirements using Gamification: A Gamify4Req Approach", *Journal of Software Evolution and Process.* [Under Review] *[IF 2.55]*

TABLE OF CONTENTS

## LIST OF TABLES

## LIST OF FIGURES

# List of Acronyms

| | |
|---|---|
| AUCD | Architecture Use Case Diagram |
| ARM | Automated Requirements Management |
| AM | Acceptance Model |
| AHP | Analytic Hierarchy Process |
| ATM | Air Traffic Management |
| CCRE | Crowd-Centric Requirements Engineering |
| CNL | Controlled Natural Language |
| CARE | Create, Ask for review, Review, Extend |
| CA | Coordination Ambiguity |
| DMGame | Decision Making Game |
| DE | Domain Expert |
| ELICA | Elicitation Aid |
| FSS | Flow Short Scale |
| GREM | Gamified Requirements Engineering Model |
| GARUSO | Gamification Approach |
| GM | Gamification Model |
| GOTCHA | Depression Helpline |
| HEC | Higher Education Commission |
| HOB | Human and Organizational Behaviour |
| ML | Machine Learning |
| MHLW | Ministry of Health Labor and Welfare |
| MAF | Motivational Antecedents Framework |
| NFR | Non-Functional Requirements |
| NAI | Nocuous Ambiguity Identification |
| NL | Natural Language |
| NLP | Natural Language Processing |
| PACAS | Participatory Architectural Change Management |
| PANAS | Positive and Negative Affect Schedule |
| PLB | Points Levels Badges |
| PBL | Points Badges Leaderboard |

| | |
|---|---|
| PM | Project Manager |
| POS | Parts of Speech |
| QuARS | Quality Analyser for Requirements Specification |
| QoL | Quality of Life |
| QA | Quality Assessment |
| RA | Referential Ambiguity |
| ReqEngr. | Requirement Engineer |
| RETA | Requirements Template Analyser |
| REFINE | Requirements Elicitation and Refinement |
| REVISE | Requirements Elicitation and Verification Integrated in Social Environment |
| RE | Requirements Engineering |
| SARAF | Systematic Acceptance Requirements Analysis Framework |
| SCRUM | Systematic Customer Resolution Unravelling Meeting |
| SDA | Student Direction Application |
| SLR | Systematic Literature Review |
| SUPERSEDE | Supporting Personalized Software by Exploiting Data and End user Feedback |
| SBVR | Semantic Business Vocabulary and Rules |
| SRAAF | Software Requirements Ambiguity Avoidance Framework |
| SPO | Software Product Organization |
| SDLC | Software Development Life Cycle |
| SE | Software Engineering |
| SRS | Software Requirements Specification |
| TM | Tactical Model |
| UI | User Involvement |
| UML | Unified Modelling Language |
| UCM | User Context Model |

# CHAPTER 1

## *INTRODUCTION*

# Chapter 1 Introduction

## 1.1 Requirements Elicitation

Requirements elicitation is the foremost activity in Requirements Engineering (RE). During elicitation, requirements are collected from the stakeholders, users, and customers to build the system. Requirements elicitation means discovery of requirements, which is a critical activity [1]. With the help of elicitation, the real needs of the customer and system are identified. Elicitation holds a crucial place in the software development process. The analyst keeps the focus on understanding the requirements, scope, vision, and constraint of the system under development [2]. Similarly, requirements are defined in a broader context in elicitation phase. Requirements elicitation tends to focus on i) knowledge area where system is applied, ii) specific customer problem where system will be applied, iii) system interaction with environment, iv) investigation of user needs and v) constraints of system development [3].

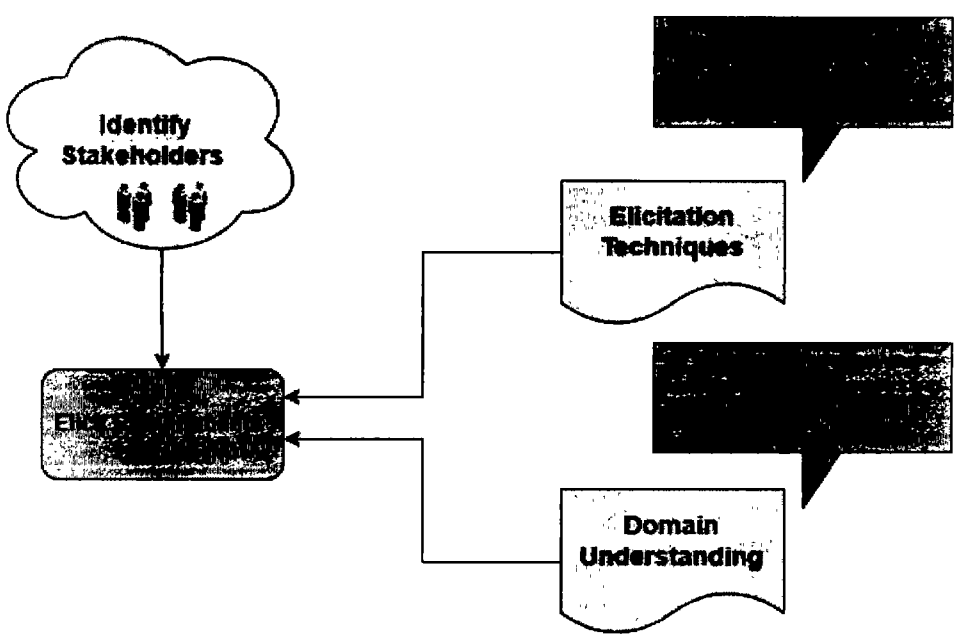



*Figure 1.1 Requirements Elicitation*

As shown in the figure 1.1, requirements elicitation is performed for the discovery of requirements. Stakeholders are identified and selected for requirements elicitation. Requirements elicitation is performed by using number of elicitation techniques such as traditional techniques, cognitive techniques, contextual techniques, and others [4]. These techniques help the RE team to collect correct and valid requirements, such that it does not affect the rest of the software development life cycle (SDLC) phases. The selected elicitation techniques are based on different parameters specific to project such as technical factors and project environment. After selection of related techniques, requirements are collected from the users. The conventional textual elicitation techniques acquire requirements either by asking questions, observing relevant documents or usage scenarios.

Requirements elicitation involves a variety of elicitation methods such as classic or traditional [5] [6], observational, contextual, or social analysis, [7], and cognitive or analytical to gather the requirements. These methods are further divided into a number of techniques used by the RE team according to the nature of the project and convenience of use [8]. The elicitation techniques are given in detail below.

### 1.2.1 Traditional Elicitation Techniques

Traditional requirements elicitation techniques encourage verbal communication between stakeholders and the team. It is a natural way of expressing needs by answering questions [5]. Because of the nature of these methods, it is effective in understanding problems and eliciting product requirements. The most common techniques used under conversational methods are interviews, questionnaires/surveys, brainstorming sessions, and focus groups. An analyst or experienced facilitator from within the organization or someone hired externally conducts the brainstorming and focus group sessions. The techniques under traditional methods focus on defining product features and creating innovative ideas, but there are certain limitations to them.

### 1.2.2 Contextual Elicitation Techniques

This method provides a deep understanding of the system by closely observing human behaviour [9]. Contextual techniques allow to elicit the requirements at a user's workplace,

which means the requirements are collected with a closed context of end-user. Ethnography, social analysis, and observation [10] are some common techniques of this method. The observer plays an important role in the execution of these techniques.

### 1.2.3 Cognitive Elicitation Techniques

Cognitive techniques extract requirements directly from the user's behaviour. Besides studying human behaviour, it also includes extraction of requirements from existing documents or acquiring the requirements from series of inferences. Cognitive method includes different elicitation techniques [5] such as card sorting [8], laddering, and repository grids etc.

Each technique differs, based on its suitability to the scenario, the number of users, mode of user availability, contextual informational, domain knowledge and others. Any problem with the collected requirement may negatively impact the system, thus, it is important to select an effective elicitation technique [11]. Traditional elicitation techniques comprise of different problems including problems of scope, understanding, and volatility [12]. In traditional techniques, it is difficult to ensure the presence of stakeholders [6] because either they are reluctant to give information as they are busy, or consider this activity a burden or boring, or lack of system knowledge. The lack of user involvement in providing requirements may lead to erroneous, ambiguous and misunderstood requirements [13] [14]. Some common challenges identified from literature are shown in the table 1.1.

*Table 1.1 Challenges of Traditional Elicitation Techniques*

| No. | Techniques | Methods | Challenges | Reference |
|---|---|---|---|---|
| 1. | Traditional | Interviews | Lack of user involvement,<br>Time consuming,<br>Conversation between stakeholder and interviewer is often quite difficult,<br>Setting up meeting environment and scheduling meeting | [2] [6] [15] [16] [17]<br>[2]<br>[18]<br>[19] |
| | | Questionnaire | Lacks user clarification on the topic,<br>No flexibility in stakeholder's language or idea,<br>Understanding issues due to dissimilar environments | [5]<br>[19]<br>[18] |
| | | Brainstorming | Lack of user involvement,<br>Time consuming in case not well- | [6]<br>[19] |

| | | | | |
|---|---|---|---|---|
| | | | organized | |
| 2. | Contextual | Ethnography | Limitation of the responsiveness and sensitivity of observation to real environment,<br>Requires skilled ethnographers,<br>Time consuming | [10]<br>[9]<br>[6] [9] |
| | | Social Analysis | Time consuming | [2] |
| | | Observation | Hard to specify and analyse observer perception | [6] |
| 3. | Cognitive | Card Sorting | Participants must be included,<br>Lack of user involvement,<br>Based on expert's knowledge | [5]<br>[6] [20]<br>[5] |
| | | Laddering | Domain information of stakeholder is important,<br>Lack of user involvement,<br>Does not work well with new system development,<br>Adding or deleting user requirement in hierarchy is difficult | [5]<br>[6]<br>[19]<br>[19] |
| | | Repository Grids | Information gathering is done from other sources,<br>Lack of user involvement,<br>Time consuming | [6]<br>[6]<br>[19] |

One of the challenges in requirements elicitation is lack of user involvement [15] [16] [17] which occurs due to reasons like reluctance in providing information as they are busy, or consider elicitation activity boring. Some other challenges include selection of appropriate elicitation technique(s) [18], effectiveness of elicitation technique(s) [11], lack of understanding the system and conceptual domain, volatility [12] [21], conflicting and ambiguous requirements [22] [23], lack of proper analysis [12] [18], and issues in sources of requirements [21]. However, lack of user involvement is a one of the common challenges in requirements elicitation [2] [6] [15].

## 1.2 Requirements Ambiguity

In software engineering, acquiring unambiguous natural language (NL) requirements is critical for the success of software systems. The complex structure of NL results is an inherent challenge of ambiguity. The term 'ambiguity' within RE, refers to the situation whereby a reader can determine multiple interpretations despite his familiarity with the RE context [24]. Previous studies claim that ambiguity is more intractable among other requirements defects like incompleteness and misunderstandings [22]. According to the

research [25], around 87.7% documents are specified in NL, and 5.3% of the documents are specified in formal language. Thus NL is the most widely used requirement specification language due to its comprehension and acceptability [26].

Formal languages use mathematical inductions to rigorously define both syntax and semantics. This helps in verification of the equivalence between the speciation phase and implementation, with unambiguous representation. Although the languages remove ambiguity, they are quite complex to learn and implement [25], and require expertise, time and effort for their use. They also lack the expressive power of NL due to their structured nature [27].

NL is prone to ambiguity, which can lead to incomplete, inconsistent, and misunderstood requirements specifications [28]. This misunderstanding occurs because NL expressions are based on human judgments of real-world scenarios and are later transmitted to other stages of software development [29]. Despite being a complex task [23], a more informed decision regarding ambiguity can be made by understanding its nature and source.

Much research is conducted in requirements ambiguity. Ambiguity in requirements is classified by the researchers, among which classification by Barry Boehm is quite notable [30] [31]. Ambiguity is classified as lexical ambiguity, syntactic ambiguity, semantic ambiguity, pragmatic ambiguity, and language errors. A detailed description of each ambiguity type is provided below.

### 1.2.1 Lexical Ambiguity

Lexical ambiguity occurs when a single word has multiple meanings [32] [33] [34]. Lexical ambiguity can be divided into two parts: homonymy and polysemy. Homonymy refers to words that have different meanings and etymologies but have same spellings [35], such as 'world bank', which can refer to a financial institution or loping land. Polysemous words, on the other hand, have several related meanings that may be based on context or etymology [34], such as the word 'green', which refers to the colour or to unripe fruit [36]. Despite the fact that lexical ambiguity is a crucial issue in RE, it has not received much attention from

the research community [37]. However, some work is done on lexical expressions in order to address other forms of requirements ambiguity [38] [39].

### 1.2.2 Syntactic Ambiguity

Syntactic or structural ambiguity occurs due to the parsing of a sentence in multiple ways, leading to different meanings [33] [35] [39]. If a sentence contains one or more parse in text, then it refers to as syntactic ambiguity [32]. Parse means to break down a sentence into parts so that the meaning of the sentence can be understood. For instance, '*I saw a girl with the binoculars*' can be parsed into two different meanings: 1. The girl had a binocular or 2. I used a binocular to see the girl. Syntactic ambiguity is further divided into attachment ambiguity and scope ambiguity [34]. Attachment ambiguity occurs when there is a doubt of attaching a part or clause of sentence to the other part of the sentence [40]. Scope ambiguity refers to a situation where the meaning or interpretation of a word or phrase is unclear due to its scope or range of application within a given context [33]. Moreover, the studies present in literature have also discussed the misplaced use of 'also' [41], 'only' [42], and 'cues' as syntactic ambiguity.

### 1.2.3 Semantic Ambiguity

Semantic ambiguity occurs when predicate logic leads to multiple interpretations of the sentence without any lexical or structural ambiguity [35] [43]. In semantic ambiguity, there is more than one logical expression in which a sentence can be translated [44]. The use of quantifiers along with incorrect combination leads to semantic ambiguity [37]. Anaphoric or referential ambiguity, and coordination ambiguity are two types of semantic ambiguity [34]. Words like 'many', 'some', 'each' etc. come under scope ambiguity because these words alter the scope of a sentence [45]. Anaphoric ambiguity occurs where there are multiple possible references to a previously mentioned word within a sentence [46]. The presence of quantifiers and, in particular, the word 'all' are cues for semantic ambiguity [47]. Other cues are 'all' and 'plurals' [48]. Coordination ambiguity occurs when there is more than one conjunction with a modifier in a sentence. In RE, researchers have worked on coordination ambiguity [49] such as conjunctions 'and', 'or', and differentiation between

nocuous and innocuous ambiguities. Nocuous cases lead to misunderstanding, whereas, in innocuous cases one reading is preferred [37]. Previous studies have identified that semantic ambiguity is not considered in linguistic classification [41], which includes the cues of syntactic ambiguity.

### 1.2.4 Pragmatic Ambiguity

Pragmatic ambiguity occurs due to the gap between uncertainty of common-sense knowledge and contextual knowledge of humans [32] [33]. It focuses on the meaning and context of a sentence [34] [35]. Pragmatic ambiguity depends on the context of requirements, and on the reader's knowledge. Due to the differences in readers' backgrounds, requirements are subject to varying interpretations. Furthermore, pragmatic ambiguity is considered as a sub-class of referential ambiguity in RE, which occurs when a reader disagrees on how a pronoun should be interpreted. Referential ambiguity [50] is considered in semantic ambiguity class when the potential reference of pronoun is in the same sentence of the pronoun [37].

### 1.2.5 Language Errors

Language errors occur when sentence grammar is poor, wrong, and difficult to understand [33] [37]. It is defined as a separate ambiguity class. Furthermore, figure 2.1 highlights the types and sub-types of natural language ambiguity as discussed above.

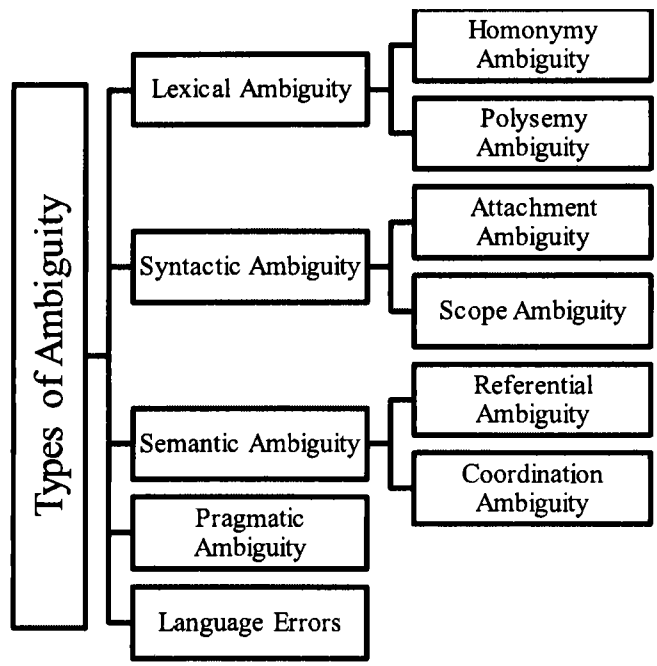



*Figure 2.1 Types of natural language ambiguities*

Ambiguity is also classified as: intentional and non-intentional [27], nocuous and innocuous [37], acknowledged and unacknowledged ambiguity [51]. RE is more focused towards ambiguity avoidance and detection. However, not all ambiguities are considered harmful; preserving intentional ambiguity is considered helpful in providing flexibility in other software development phases such as design and implementation [23].

### 1.2.6 Addressing Ambiguity

In previous studies, researchers have utilized Natural Language Processing (NLP) techniques, tools, and methods as a remedy for addressing ambiguity in software requirements. This includes identification, extraction, removal and management of ambiguity [24]. Typically, ambiguity is not addressed during requirements elicitation. Most approaches address requirements ambiguity in inspection phase where the user is not engaged in the activity, and ambiguity is identified and detected [32] from Software Requirements Specifications (SRS) written in NL. Requirements written in NL tend to be ambiguous, thus pre-processing of SRS document is required to detect and resolve ambiguity [52].

Researchers have worked on requirements ambiguity detection, avoidance, resolution, and removal. Most of the work is conducted manually or through semi-automation during requirements inspection phase. Also, manual resolution of requirements ambiguity is time consuming, error prone, and a costly process [2] [53]. Researchers have come up with the introduction of more interactive interfaces to involve stakeholders in requirements elicitation.

## 1.3 Gamification

The term 'Gamification' was first introduced in early 2000 but the concept was documented in 2010 [54]. The goal of using gamification is to engage users, consumers, employees, and partners etc. to collaborate and interact especially with the activities that are not enjoyable for them. Researchers have defined gamification in various gaming contexts among which the popular context is using gaming elements in non-gaming or serious context, as shown in the following figure.

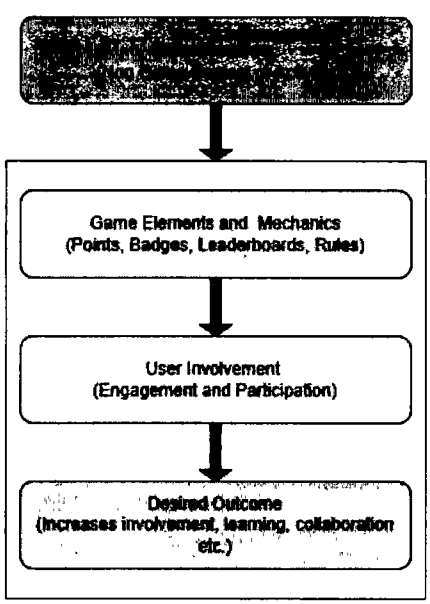



*Figure 3.1 Gamification*

As shown in Figure 3.1, gamification uses game elements and mechanics to involve users in a non-gaming context. The user involvement helps in achieving the desired goal. However,

a study [55] conducted in 2019 states that gamification is a state-of-the-art concept and has gained much popularity since then. The essential elements of a gamification framework are setting goals, rules, rewards, leaderboards, scoreboards, feedback, measurement etc. [56]. The common context of gamification is its widespread use in human resources, data collection, e-learning, healthcare [57] [58], and e-commerce [59]. Moreover, it has become a popular strategy for sustainable businesses [60].

According to the report conducted in 2012 [61], gamification can suffer from poor designs, unrealistic goals, improper rules, and gaming mechanisms [62]. Some studies have stated problems associated with the use of gamification [63] such as increase of stress and tension which can affect productivity in the long run. Implementation requires analysis, awareness to participate, lack of standard plugins; and difficulty to find active users [64]. These issues need to be addressed for an effective gamification solution. However, it is used in research and development because it incorporates mechanics and emotional appeal for encouraging motivation, and sense of flow.

Gamification supports RE by enhancing user involvement and engagement in the process by enjoyable ways [64]. By incorporating game elements and mechanics, gamification fosters collaboration, motivation, and knowledge sharing among RE team and users. It transforms the tedious and complex requirements elicitation activity, into an interactive and enjoyable experience. Furthermore, gamification promotes feedback that is productive for continuous improvements and refinement of the requirements. Gamification brings a fun-based perspective to RE, enhancing efficiency, creativity, and overall project success.

### 1.3.1 Game Elements

Gamification, with the help of game elements, focuses on user involvement. Game elements ensure user engagement in gamified systems. There are more than a hundred game elements present in literature, and some popular ones include PBL i.e. points, badges [65], and leaderboards [66] [67] [68], along with score, feedback, and display of the progress, and avatars [69].

### 1.3.2 Gamification in Requirements Elicitation

Gamification of some aspects of RE is already done by the researchers. Gamification in elicitation is used in recent years [70] [71], and is focused on involvement of users and gathering requirements. Gamification involves human-centric activities to involve and engage the users that make the process of elicitation easier. It also helps to have close interaction with the users for getting clear and understandable requirements for system development. Gamification maintains the fun part by using game elements in serious scenarios to gain user's interest during elicitation [72]. Moreover, ambiguity occurs due to the lack of information or no information at all. Thus, by motivating a user to involve in the elicitation using gamification may help in getting unambiguous requirements from the user, as shown in the figure 4.1 below.

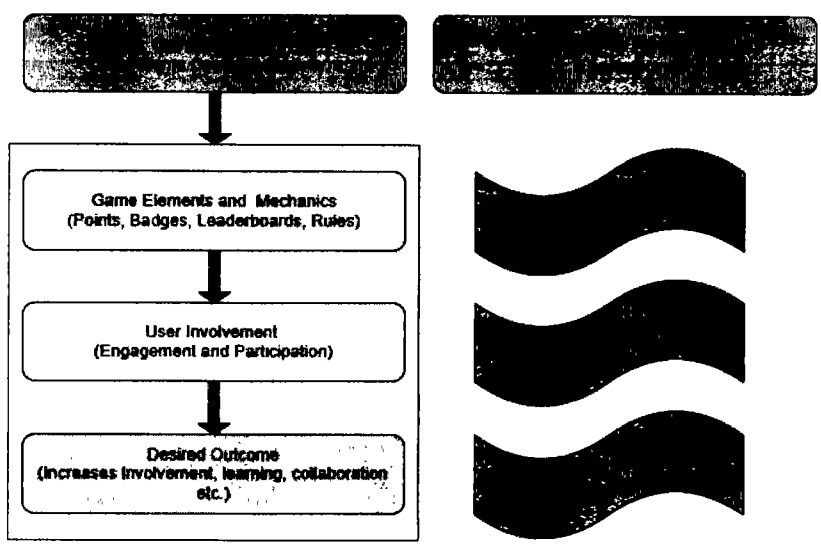



*Figure 4.1 Gamification in Requirements Elicitation*

Gamification supports requirements elicitation activity [73]. Stakeholders' involvement is increased using the gamified tool named iThink [74]. iThink uses the concept of six thinking hats to generate new requirements. Players (users) are awarded points/scores on providing and updating each requirement. The researchers suggest that due to limitation of the study, it is difficult to generalize the outcome [75]. Secondly, some aspects such as the design and usability of the platform are missing as well [76] [77].

Researchers have explored gamification in requirements elicitation with the help of different techniques and tools. Requirements visualization is considered to attract more stakeholders as participants, in eliciting requirements. For this purpose, Architect Use Case Diagram (AUCD) is introduced [78]. AUCD uses images as communication media and requirements are represented visually. The game works by setting goals, selecting players, explaining the game rules, starting the game, consolidation, and analysing the results. AUCD is developed to work between stakeholders and the elicitation process. Survey is conducted from the players and the RE team. The tool is designed to work with functional requirements, performance, and scalability requirements. AUCD does not support NL requirements and works with use case diagram.

The gamified platform GREM is used for requirements elicitation based on user stories [79]. The purpose of this platform is to engage the user and enhance the performance of RE process. Badges and points are awarded when users submit their story. It is used in a dedicated and controlled environment where participants are informed about the purpose of using this platform. A total of 12 stakeholders are involved, and split into two teams in the process, and constrained to communicate with their team members only. Despite this, the investigators stated that GREM is not a comprehensive model for requirements elicitation.

Requirements elicitation is enhanced using gamification to involve stakeholders in requirements engineering [71]. For this purpose, a web based gamified platform is introduced to address the problems of availability, and to increase the motivation of the stakeholders. The requirements are elicited in the form of user stories. The researchers conclude that the experiments may negatively affect collaboration between people by minimizing the productivity level. One of the biggest challenges is the participation of stakeholders during the experiment, because none of the participants thought it to be a fun task, yet everyone agreed that it was an important one.

Ambiguity in requirements is a common challenge because not every stakeholder participates in requirements elicitation activity. Even if they get involved in the activity, they do not understand the system, hence, cannot tell the requirements. This results in ambiguous requirements. Due to the limitations of current elicitation techniques, the problems are still

there. While addressing the level of requirements ambiguity, three approaches are used i.e., ambiguity detection, ambiguity removal, and ambiguity reduction. Ambiguity detection and removal are reactive approaches and require SRS or requirements document to detect and remove ambiguity, whereas ambiguity reduction is a proactive approach. Pro-active approach means to identify and remove ambiguity at the time when requirements are being provided/written [80]. Ambiguity in requirements is detected and removed during requirements inspection. Due to lack of user involvement in the inspection process, SRS document is the source of requirements inspection. Moreover, the ambiguity identified at this stage requires much rework and effort for its removal. The possibility of misinterpretation of requirements remains high. However, the use of gamification enhances user involvement in activity by employing fun elements or game elements. The use of gamification in requirements elicitation helps in mitigating requirements ambiguity. Traditional elicitation methods often encounter challenges in collecting comprehensive and clear user requirements due to lack of user involvement. Whereas gamification offers more structured and engaging approach that turns elicitation process into a game-like experience. By employing fun elements such as points, badges, rewards, and competition, it encourages active participation, enhances user involvement, and fosters a more dynamic interactive environment for collaboration. As a result, this not only makes the elicitation process enjoyable but also helps the users to express their needs more explicitly, leading to reduced number of ambiguities in requirements. It also facilitates in extraction of more precise requirements for software development projects.

Due to complex structure of NL, it is hard to design a comprehensive solution to address all types of ambiguities. For this reason, the scope of this study is limited to semantic ambiguity only. Semantic ambiguity deals with the misinterpretation of a word/phrase which is common in SRS during inspection. This study aims to design and develop a solution that uses gamification to encourage user participation during elicitation for the purpose of reducing ambiguity in NL requirements. Unambiguous requirements can be ensured while ensuring the participation of users in requirements elicitation.

## 1.4 Research Motivation

In previous studies, researchers have proposed several approaches for the identification and removal of ambiguity in requirements [25] but there are very few platforms that involve users during elicitation [81] [82] [83] [84] [85] for reducing ambiguity in requirements [86]. Another problem with requirements ambiguity reduction platforms is their non-interactive and non-friendly interfaces, as they do not encourage users in the activity. If users are not involved, then the requirements team misinterprets user requirements, which results in requirements ambiguity and erroneous software models [82]. User involvement in the system helps to reduce ambiguity. The motivation of this work is to reduce ambiguity in requirements by i). Involving users during elicitation, and 2). Using ambiguity rules. In this way, not only user involvement is enhanced but the user becomes helpful in reducing ambiguity at the time of elicitation. We have used gamification for the purpose of user involvement during elicitation to keep them interested and engaged.

## 1.5 Problem Statement

Researchers have proposed several manual, automated, and semi-automated techniques, methods, and tools etc. for the identification and removal of ambiguity in requirements. There is limited work in previous literature on reducing ambiguity in NL requirements because ambiguity is a complex problem and difficult to address comprehensively. The proposed approaches are reactive and do not involve users in requirements elicitation. Even if users are involved in the activity, they lack interest due to long meetings, boredom during activity, and unclarity of the system. Existing approaches address ambiguity at the time of requirements inspection, and consider SRS or requirements documents as source of requirements, thus minimizing the need of users at the time of elicitation and maximizing the chances of ambiguity during requirements elicitation. Furthermore, these systems are not user friendly, interactive, and have limitations regarding responsiveness.

## 1.6 Aims and Objectives

Ambiguous requirements have a negative impact on software development processes, and results in system failure. This study aims to reduce requirements ambiguity by involving users during requirements elicitation. For this purpose, the study has the following research objectives (RO).

**RO 1:** To identify the game elements for requirements elicitation.
**RO 2:** To reduce ambiguity during elicitation via user involvement.
**RO 3:** To validate the tool with the help of multiple case studies.

### 1.6.1 Research Questions

This study comprises of following research questions (RQ), formulated to address the objectives of this study.

**RQ 1:** *What are the useful game elements for elicitation of requirements?*

**RQ 2:** *What are the game elements empirically validated for requirements elicitation?*

**RQ 3**: *How effective is the developed gamification tool in reducing ambiguity during elicitation?*

## 1.7 Research Methodology

Research methodology helps in systematically conducting the investigation to answer the study questions. The research methodology of this study is as follows:

### 1.7.1 Literature Survey

A preliminary literature survey is conducted for problem identification and investigation. The goal of the literature survey is to investigate the concept of requirements ambiguity and gamification in requirements elicitation. It mainly covers related studies on requirements ambiguity and identification of game elements that are useful in requirements elicitation.

Computing databases such as IEEE, ACM, Springer, Science Direct etc. and search repositories are used to access relevant material. The literature survey focuses on those game elements that are useful in reducing ambiguity in requirements during elicitation.

### 1.7.2 Systematic Literature Review

A literature survey is performed at an initial level but for in-depth coverage and identification of game elements, Systematic Literature Review (SLR) is performed as part of research methodology. SLR is conducted for empirical validation of game elements for requirements elicitation. Furthermore, it is performed to systematically identify, evaluate, and interpret all evidence. Computing databases are selected to search relevant studies. These databases include Google Scholar, IEEE, ACM, Springer, Science Direct etc. The SLR identifies game elements that are useful in reducing requirements ambiguity during elicitation.



### 1.7.3 Tool Development

Gamification is an interactive platform through which users are involved in the system and provide unambiguous requirements. The gamified tool is designed and developed for reducing ambiguity in requirements during elicitation. The gamified tool is developed based on game elements, game rules, and ambiguity rules. The web-based gamified tool is built in Java Virtual Machine (JVM) using Apache webserver. It is deployed using MySQL database at the back-end. The gamified tool supports RE teams during requirements elicitation for collecting ambiguity-free requirements.

### 1.7.4 Tool Validation

Validation is a significant part of the study. It is performed by conducting two case studies from the software market. The case studies are confirmatory in nature. Requirements elicitation is performed with the help of the gamified tool. The tool is not an existing tool. It is a gamified tool designed, developed, and validated as part of this research study.

Objective 1 is mapped on two RQs of the study. The method used to answer RQ 1 is literature review, while SLR is used to answer RQ 1.1. The outcome of this phase are the identified game elements that are useful during elicitation. Similarly, Objective 2 is mapped on RQ 2.

A tool Gamify4Req is developed to identify and reduce ambiguity in requirements during elicitation. Finally, objective 3 is mapped on RQ 2, in which case study and feedback survey is used as a research method. The outcome of this phase is a validated tool and feedback on user engagement in the activity. Figure 5.1 presents the research summary of research objectives (RO), RQ, and research methods (RM) on research outcomes of this study.

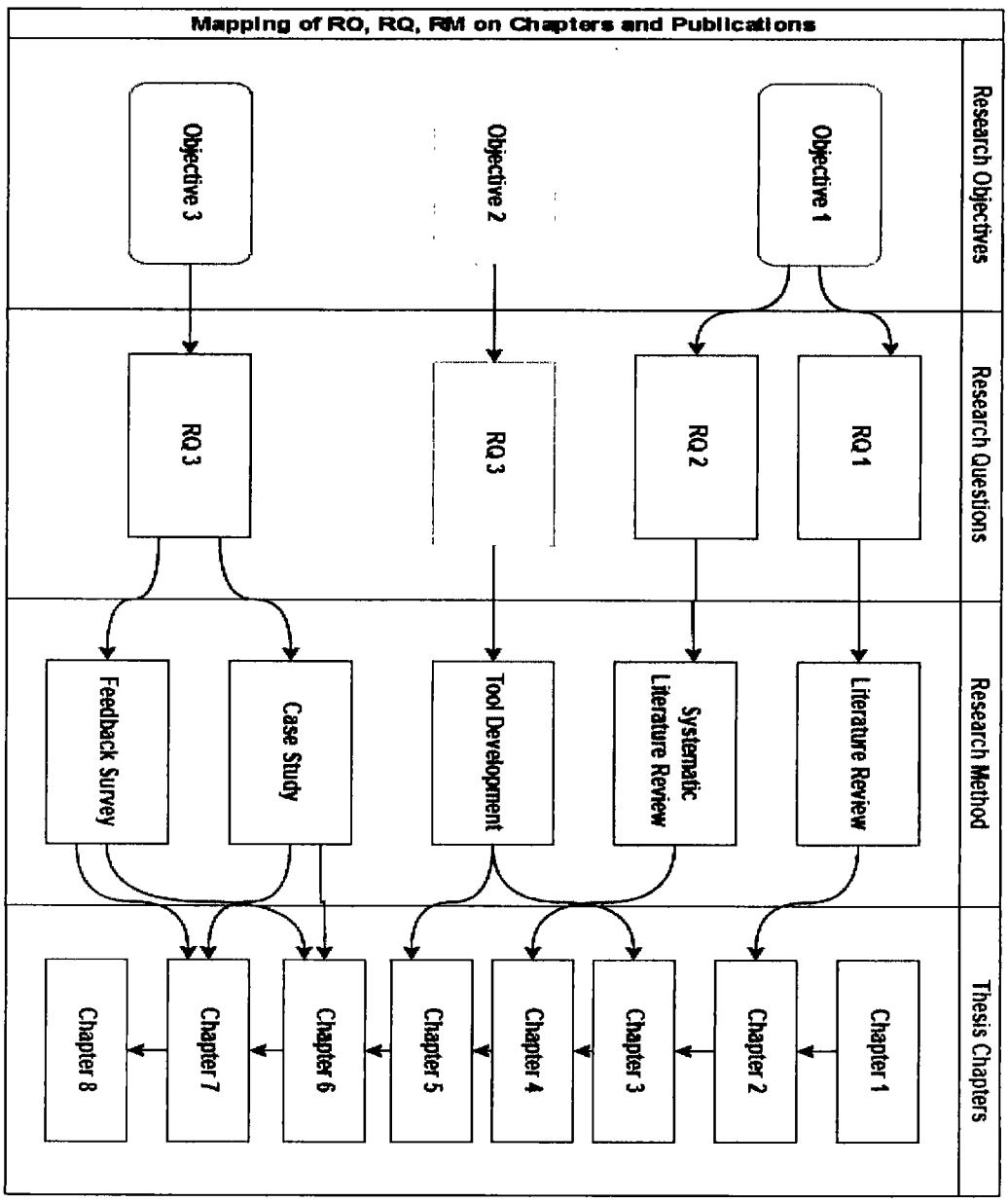



*Figure 5.1 Research Summary: Mapping of RO, RQ and Research Methods*

Figure 5.1 shows the mapping of RO, RQ, and research method with respect to thesis dissertation chapters of this research. We have also published all major sections of this research in well-reputed journals and conferences. Chapter 2 and 4 present literature review and SLR respectively, which are published in peer reviewed journals. Tool development is discussed in chapter 5 and mapped on a conference publication related to tool design and development. Lastly, the results and findings, mapped in chapter 6 and 7, are published in a peer reviewed journal.

## 1.8 Thesis Contribution

The study offers contribution in this field by:

- Identifying useful game elements for reducing requirements ambiguity during elicitation.
- Identification of useful game elements other than the PBL for reducing requirements ambiguity during elicitation
- Defining game rules for gamified tool to reduce requirements ambiguity during elicitation.
- Developing a gamified tool for reducing requirements ambiguity during elicitation. The tool incorporates game elements, game rules and ambiguity rules to identify ambiguity and reduce ambiguity with user involvement during elicitation.
- Performing validation of the tool in real world settings with two confirmatory case studies selected from the software industry.
- Conducting a survey to get the feedback on user involvement in a gamified tool for reducing requirements ambiguity during elicitation.

## 1.9 Dissertation Structure

To perform the research study in this dissertation, we have divided each part in a logical manner. The thesis dissertation structure of the study is given below in Figure 7.1.

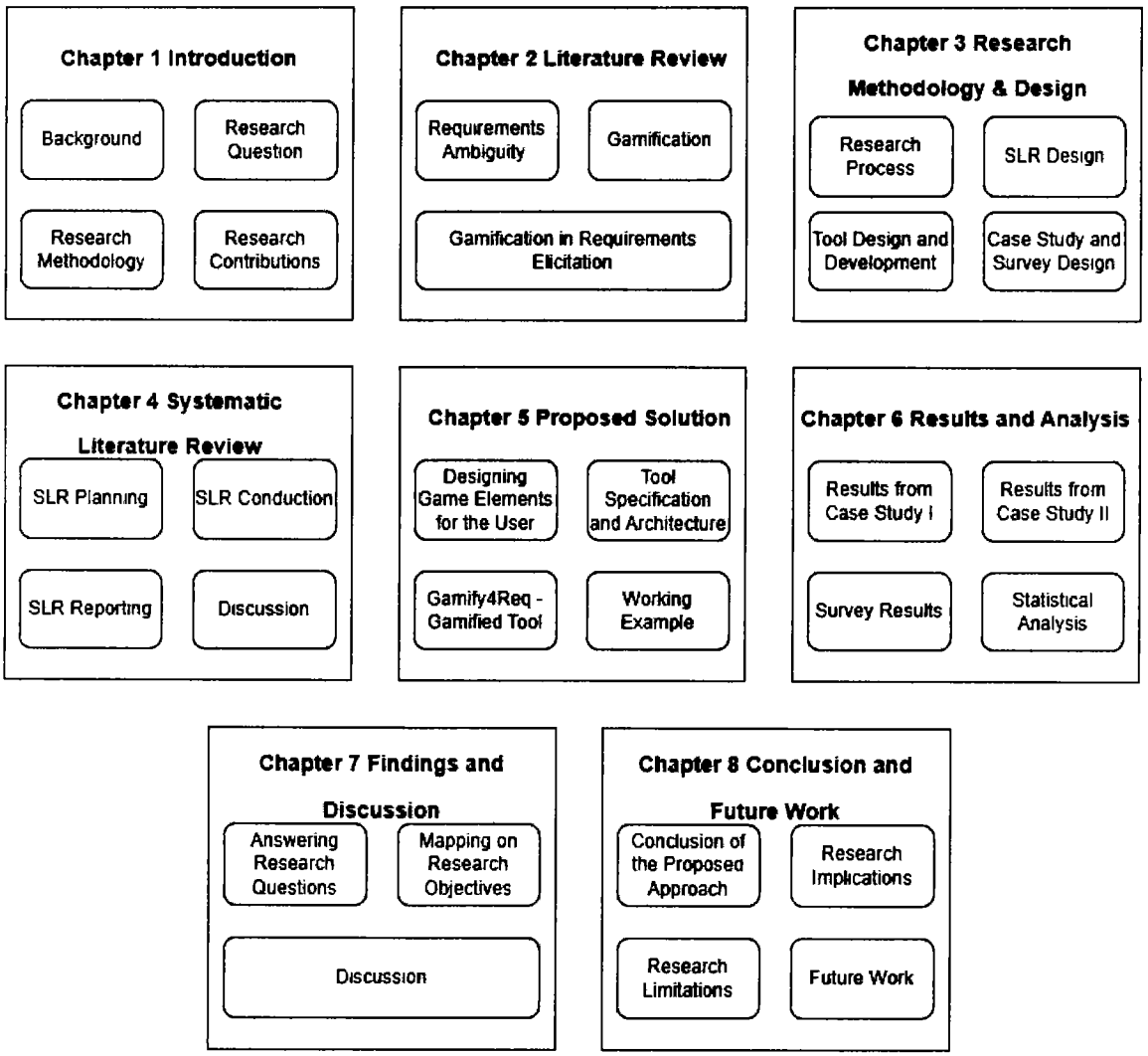



*Figure 6.1 Logical Structure and Relationship between Chapters*

- Chapter 2 presents detailed literature review on requirements ambiguity, requirements ambiguity in elicitation, gamification and game elements, and gamified tools. The literature is extracted from computing databases, consisting of reputed journals and conferences.
- Chapter 3 covers important sections of research methodology of this study. For this research, an initial review is extended and carried out in detail to identify gaps in reducing requirements ambiguity. Game elements are also identified and used as recommended by the literature.

- Chapter 4 presents SLR on identification, analysis, and interpretation of relevant studies on game elements, other than the game elements identified in chapter 2. In this chapter, SLR protocol is presented along with the outcomes of SLR.
- Chapter 5 provides a discussion on gamification tools and parameters used in this tool. Technical aspects of the tool development are discussed in this chapter. The chapter covers the proposed solution. Chapter 5 also presents tool validation on multiple case studies. Case studies are discussed along with the selected cases and projects. Also, a feedback survey questionnaire is discussed in detail.
- Chapter 6 describes the results and analysis of requirements ambiguity gamification tool. This chapter covers detailed results of both cases and projects. First, results with previous approach are presented, which are later presented using the proposed tool. Statistical analysis of the survey is presented in this chapter.
- Chapter 7 covers the findings and discussion section. Each objective is explained, and research questions are answered in this chapter.
- Chapter 8 concludes the research with directions for future study. It is useful for researchers and practitioners. There is an attempt to present the limitations of the approach and relate them to future directions. Implications on research for industry are also presented along with threats to validity.
- References presents all referencing material used in this research.
- Appendices cover appendix sections covering details on results, checklist, survey questionnaire, and statistical tests

# CHAPTER 2

## *LITERATURE REVIEW*

# Chapter 2 Literature Review

## 2. Literature Review

In this chapter, literature review is presented. Relevant studies on requirements ambiguity in elicitation, gamification in elicitation, and game elements in RE are covered.

## 2.1 Background

Requirements elicitation helps to get user requirements, but it also suffers from the biggest challenges in software development such as user understanding, involvement, and expectation from the system [87]. Therefore, it is necessary to specify the requirements with proper care to avoid confusion and ambiguity during software development. However, resolving requirements ambiguity is a time taking, error prone and costly process [52] [88]. Ambiguity in requirements arise from the discrepancy between how customers articulate units of information and how analysts interpret them [37]. These units of information contain two types of data: one pertaining to system needs and the other to domain aspects. The term 'articulation of unit of information' refers to any spoken sentence fragment or set of words. While NL is commonly used to specify requirements [89], requirements written in English may contain problems such as ambiguity [90] and inconsistency. This ambiguity is present due to the inherent ambiguity in English language.

### 2.1.1 Related Work on Ambiguity in Natural Language Requirements

A system is developed to document requirements in natural language in which written requirements are checked for validity and ambiguities, specifically lexical ambiguity including 'homonyms', 'homographs', and 'heteronyms' [25]. The system works in two phases: check requirements validity, and check ambiguity in requirements. For this purpose, an algorithm is designed and implemented in eight steps. However, the authors claimed that it is possible that any kind of ambiguity might be present in the examined sentence, even the lexical one, because natural language is quite a vast field. Despite the conventional

approaches to recognize the functionality of any system, Object-Oriented approach based on natural language processing (NLP), has gained much popularity [91]. This is the reason requirements are widely written in natural language. An approach named OpenNLP is used to process SRS by producing parts of speech (POS) tag from NL statements. This approach is designed to support developers and software engineers to extract object-oriented elements from requirements. Ambiguity is addressed once requirements are collected from the user. A theoretical framework is developed for categorizing ambiguity in interviews, during requirements elicitation phase [92]. This framework is based on four sub-phenomena including unclarity, multiple understanding, correct disambiguation, and incorrect disambiguation. This work also shows the relationship between ambiguity and the knowledge of the customer which is not passed to the analyst during interviews. Generally, ambiguities that arise during interviews are due to misunderstandings. A set of 34 analyst and customer interviews are conducted to study the phenomenon. It is concluded that analysts should know what ambiguity means because it helps throughout the process of requirements engineering.

Similarly, an approach is presented to enhance the concept of ambiguity with two categories i.e., intentional, and unintentional [93]. The 'intentional' category deals with the concepts presented in literature, whereas 'unintentional' focuses on misunderstanding in language during communication. Reference material from literature is studied that presents semantic ambiguity in words and sentences. It is concluded that ambiguity affects the ability of a receiver, while in written and spoken language, to understand or communicate. However, ambiguity reduction is significant if individuals have a better understanding of language and know the implementation of the context.

A system in RE is implemented to show the need of deep semantics for understanding the meaning of natural language requirements [94]. SenseGraph implements a deep semantic by feeding NL requirements documents and generates a deep semantic version showing casual and temporal links, which could be queried by NLP too. SenseGraph presents information as objects and events: objects are the concepts derived subjects, and event generally represents meaning of the sentence. This is helpful in supporting human activities by

automating RE activities, thus overcoming the challenge of linguistic analysis that is based on shallow semantics. The system is preliminary validated for analyzing national projects on terrorism. Ambiguity resolution does not address the association of various contexts with a given word relating to each other.

To identify defects in requirements such as ambiguity and incompleteness, an approach is proposed [95] that investigates human cognitive and analytical abilities with automated reasoning to find out if this will affect quality of a requirement. A proposed tool-based approach pinpoints ambiguities between different viewpoints and missing requirements. For this purpose, NLP is blended with visualization techniques to interpret the type of defects. To validate the tool, a quasi-experiment is conducted on a small group of students, which yields significant results in recall, but no significant difference is found in precision. It is concluded that visual presentation of requirements in 2D space could be interesting and opens door for future work in requirements.

A theoretical framework is proposed for the occurrence of ambiguity in requirements [96]. The proposed framework addresses semantic ambiguity in requirements elicitation and analysis. The roles of the reader and author are interacting with the requirements, in which author is committing a requirement whereas the reader is a participant in the development process. Both roles may or may not recognize ambiguity present in requirements. A framework to avoid ambiguity, known as SRAAF – Software Requirement Ambiguity Avoidance Framework [97], is proposed to support requirement engineers for writing unambiguous requirements by selecting an appropriate elicitation technique. The framework was based on the evaluation of different attributes, conducted with the W6H technique. The selection of elicitation technique was based on situational attributes such as project features, attributes of requirements engineer and stakeholders, and W6H technique was then applied to handle ambiguity. The focus of this study was to avoid ambiguities before writing any statements to SRS document. The framework was not fully developed and implemented and does not involve advanced NLP technology to apply W6H techniques, because it is currently not available.

Different stakeholders with diverse backgrounds need effective communication to reach a common understanding during RE. A similar solution in NLP is proposed for identification of ambiguous terms between different domains [98]. The approach is used to construct domain specific models for 'one for each stakeholders' domain. The word embeddings of each model are compared to measure the differences of term usages. The approach is evaluated on five domains and seven scenarios of elicitation. During evaluation, automatically detected ambiguity is ranked with the manually detected ambiguity. Although the automatic ranking led to a maximum of Kendall Tau's of 88%, yet its performance is unsuccessful in several scenarios.

A framework for NL to Controlled Natural Language (CNL) is designed and prototype is developed [99]. This proactive approach is designed to generate clear and understandable requirements with the help of SBVR. SBVR provides semantic formulations to make English statements controllable and distinct. It also eliminates the scope of ambiguity and referential ambiguity. Data is collected from open-source SRS documents. The evaluation of the framework is conducted via performance evaluation and document verification. Furthermore, a rule-based parser algorithm is used to identify the tags. The approach yields higher accuracy as compared to other NL-based tools with 0.94 recall and 0.97 precision value.

Ambiguity is seen as an intrinsic problem in NL-based requirements, according to some studies, whereas a study [100] indicates ambiguity in user stories as a linguistic and cognitive problem. A framework is proposed to identify ambiguities in user stories. Firstly, human-centred factors are identified that influence a reader's interpretation of viewing user story. Secondly, a framework is developed for detecting problems with the user stories containing ambiguity. The framework is evaluated by conducting experiments, to test its effectiveness.

Ambiguity detection tools help to save time during RE. Four tools are compared with the dataset of 180 system requirements provided by Alstom [101] namely: Automated Requirements Management (ARM), Quality Analyzer for Requirements Specifications (QuARS), Requirements Template Analyzer (RETA), and Requirements Complexity Measurement (RCM). These tools are used to analyse requirements and detect ambiguities.

Results show that out of 180 requirements 74 requirements are ambiguous with the highest precision obtained at 0.68 by RCM. RETA achieved the highest recall of 0.98. It is hard to achieve high precision in ambiguity detection using pattern-based detection. High recall is also possible, but it will also result in false-positive results.

Table 2.2 discusses the contribution of previous works presented in literature, identified area of RE they worked on, and proposed approach.

*Table 2.2 Requirements ambiguity in different phases of RE*

| Source | Proposed Approach | Identified RE Area | Identified Amb. | Addressing Ambiguity | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | Detection | Avoidance | Reduction | Resolution | Removal |
| 2023 [102] | • SLR on ambiguity detection in RE<br>• 60 papers between 2000-2023 | Reqs. elicitation | - | ✓ | - | - | - | - |
| 2022 [101] | • Four tools used with dataset of 180 system reqs.<br>• Shows different recall and precision values | NL Reqs. | Semantic Amb. | ✓ | - | - | - | - |
| 2022 [103] | • Auto handling of anaphoric amb.with NLP and ML tech.<br>• Dataset of 1350 reqs.<br>• SpanBERT for detection | Reqs. Specification | Anaphoric Amb. | ✓ | - | - | ✓ | - |
| 2022 [104] | • TAPSHIR is developed for ambiguity detection and resolution in requirements.<br>• Reviews pronouns and revises the pronouns that create amb. | Reqs. statements | Semantic Amb. | ✓ | - | - | ✓ | - |
| 2022 [100] | • A framework to identify ambiguity in user stories.<br>• Human centered factors are identified.<br>• Framework is evaluated by conducting experiments, to test its effectiveness. | Reqs elicitation: user stories, validation | Semantic Amb. | ✓ | - | - | - | - |
| 2021 [99] | • A framework for NL to CNL<br>• Proactive approach that uses SBVR.<br>• SBVR provides semantic formulations to make English statements ambiguity free.<br>• Yields higher accuracy as compared to other tools with 0.94 recall and 0.97 precision | Reqs. elicitation | Semantic Amb. | ✓ | - | ✓ | - | - |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | value. | | | | | | | |
| 2021 [105] | • Use of SBVR based CNL to capture reqs. and prepare SRS.<br>• Two sets of reqs., manually validated | Reqs. Document | Semantic Amb. | - | - | - | ✓ | - |
| 2019 [97] | • SRAAF helps to write unambiguous reqs. by selecting appropriate elicitation techniques<br>• Works with W6H techniques for the evaluation of different attributes | Selection of elicitation technique,<br><br>SRS document | Semantic Amb. | - | ✓ | - | - | ✓ |
| 2018 [95] | • To identify ambiguity in requirements<br>• Investigates human cognitive and analytical abilities with automated reasoning.<br>• Tool pinpoints ambiguities between different viewpoints and missing requirements. | Reqs. Statements | Semantic Amb. | ✓ | - | - | - | - |
| 2018 [106] | • Developed software to create unamb. reqs. description by combining expert tools.<br>• Developed linguistic indicators | Reqs. description | Refrential Amb., Lexical Amb. | ✓ | - | - | ✓ | - |
| 2016 [92] | • Designed theoretical framework for categorization of amb. in interviews.<br>• Based on correct and incorrect disambiguation | Reqs. gathered during interviews | Semantic Amb. | ✓ | - | - | - | - |
| 2010 [107] | • A ML algorithm<br>• A prototype tool named Nocuous Ambiguity Identification (NAI) used for evaluation of given approach | Reqs. document written in NL | Coordinatio n Amb. | ✓ | - | - | - | - |
| 2008 [25] | • A system to document requirements in NL.<br>• Checks for ambiguity validity.<br>• Developed an algorithm | Considers entire NL reqs. | Semantic Amb. | ✓ | - | - | - | - |
| 2008 [108] | • Designed an experimental tool SREE to detect and avoid amb. In requirements | SRS | Words 'all, any, or, and, but, unless, if, only, also, it, they' | ✓ | ✓ | - | - | - |

It can be seen from the Table 2.2 that existing approaches address ambiguity in detection, avoidance, reduction, and resolution. There are certain limitations of existing approaches. For instance frameworks and models are developed but there is no performance evaluation and implementation of the frameworks [100] [98] [97]. Furthermore, practical usefulness of the tool is not determined [104]; even tools fail to recognize ambiguity in requirements [96]; they are insufficient and not suitable for different types of ambiguity [25], and no significance difference is found in precision of results [95].

#### 2.1.1.1 Requirements Ambiguity Identification and Reduction Using Machine Learning-Based Approaches

To detect ambiguity, a machine learning (ML) algorithm is used on requirements document written in natural language [107]. The specific ambiguities present in sentences are first extracted automatically. Based on a set of heuristics collected, drawn on human judgments, a machine learning algorithm determines whether the ambiguity is nocuous or innocuous. Nocuous Ambiguity Identification (NAI) a prototype tool is implemented to evaluate the approach. However, to explore further aspects of ambiguity, more heuristics are required to enhance the validity of the tool. This tool cannot be used for all kinds of ambiguity and focuses only on coordination ambiguity. An approach based on bidirectional encoder representations from transformers (BERT) was used for identification of ambiguity in requirements [109]. For all the contexts, in which a term or word is used in the requirements document, this approach displays a list of similar words. Along with the word, it also shows example sentences from the corpus of words signifying context-specific interpretation of the word. The work was further validated on textual data taken from eight different application domain. The proposed approach proved to be effective in identifying ambiguities in inter-domains. Another tool named TAPSHIR [104] is developed for ambiguity detection and resolution in requirements. The tool reviews pronouns in requirements specification and revises the pronouns that create misunderstanding during the development process. TAPSHIR works two-ways: firstly, it uses ML to detect anaphoric ambiguity in requirements, and secondly, it automatically interprets anaphora occurrences. The results

show that TAPSHIR has detected all ambiguous requirements with a recall value of 100%. Requirements Engineers can easily review and validate the outcome produced by TAPSHIR.

In a similar work [103] an automated approach was designed for addressing anaphoric ambiguity in requirements. Both NLP and ML technologies were utilized to pursue six alternative solutions for empirically assessing each solution with 1350 requirements collected from the industry. Side-by-side empirical investigation of each solution was helpful in examining the usefulness of different NLP and ML technologies for addressing RE challenges. Supervised ML for large-scale language model outperformed in ambiguity detection, whereas for anaphora interpretation SpanBERT – a variant of BERT, yielded accurate solution.

These approaches are a way to identify and reduce ambiguity in requirements but are not related to scope of our study.

### 2.1.2 Visual Techniques in Requirements Elicitation

The requirements elicitation process is supported by an interactive visual analytic tool named ELICA – ELICitation Aid [110]. Its purpose is to record intentions of the speaker with the help of interactive visualization, combined with emotions and an analytical tone to get requirements from a repository of existing documents. Hence, user participation is missing in this ELICA which makes it limited in its use.

This limitation can be managed in collaborative requirements elicitation by using visualization tools. These visualization tools are used to increase awareness and understanding of the system, among users [111]. Examples of these tools are Borland's Caliber Analyst, GatherSpace, and Requirements Composer with features of dashboards, 3D tags, bubble charts, and tree maps for supporting visualization in requirements elicitation.

Another important aspect is to send frequent reminders to the users for using elicitation platforms regularly. One such visualization supporting tool is AirT [112] that uses storyboards to elicit users' feedback. Few of its limitations include infeasibility with the project resources and timeline, due to which some requirements were postponed to later versions.

It is observed that more than one elicitation technique is used during requirements elicitation. However, researchers have worked on the development of more interactive interfaces for increasing users' involvement and interest during elicitation, to extract required information from them. In recent years [71] [73], 'gamification' has been used in requirements elicitation with a purpose to involve more users to gather requirements.

## 2.2 Gamification

### 2.2.1 Background

Originated in the digital media industry in 2008, the term 'gamification' was adopted at a global level in 2010 [54]. The main reasons behind its widespread popularity were conferences and industry players. Since then, gamification has gained much popularity [68] [113] because it includes game elements just like video games, that tend to entertain the users by creating more fun. Gamification is an informal term, that refers to enhancing user involvement and experience, using game elements in non-gaming systems [56]. According to the Cambridge Advanced Learner's Dictionary and Thesaurus [114], it is *"the practice of making activities more like games in order to make them more interesting or enjoyable"*. Similarly, Oxford Learner's Dictionary [115], defines it as *"the use of elements of game-playing in another activity, usually in order to make that activity more interesting"*. According to its more formal definitions, gamification makes non-gaming activities more interesting [116]. However, the effectiveness of gameful systems depend on the way they help the user to achieve their goals [117] [118] [119]. There are similar concepts associated with gamification such as 'game layers', 'behavioural games', 'applied gaming', 'ludic qualities', 'applied gaming', 'gamenics theory', 'taskification', 'productive games', 'gameful design', 'storyfication' and 'includification'.

Gamification is a significant approach used in non-gaming settings [120], and is recommended in those applications where boredom, repetition and passiveness are dominant to encourage a desired behaviour. Not only this, but it also describes interactive features to engage and motivate [55] [121] [122] end-users to participate [123] with the help of game

mechanics and elements [113]. In other words, it makes boring tasks interesting and enjoyable [76] with the help of gaming elements involved in it, whereas personalized gamification is more effective than non-personalized gamification [124] possibly due to an increased level of user motivation.

There is a lot of difference between gamification and digital games [125]. A digital game is taken as a formal system based on rules with a quantifiable outcome, where player(s) puts effort to shape the outcome [126]. On the other hand, gamification applies the gaming elements to enhance user experience and engagement in non-gaming contexts. Table 3.2 shows the difference in a more comprehensive manner.

*Table 3.2 Digital games vs. Gamification*

| Digital Games | Gamification |
|---|---|
| Rules and objectives derive games | Reward and points are deriving forces |
| Outcome is in the form of winning or losing | To motivate users, points are given. Losing is not possible |
| Complex and costly | Not complex and costly |
| Based on some story, content-based | Addition of gaming features without content |

A digital game is based on rules and objectives, where the result is always either winning or losing a game. Games are also considered to be expensive mediums with complex structures. On the other hand, gamification motivates the players by giving rewards and points, and is not considered as expensive and complex a medium.

### 2.2.2 Game Elements and Game Rules

Gamification doesn't create games but rather uses gaming elements to solve business problems [66]. Hence, it is important to know which game elements are included in gamification. Generally, the elements described as 'ten ingredients of great games', include avatars' self-representation, 3D environments, narrative contexts, levels and ranks, feedback, reputation, economies and marketplace, rule-based competition, teams, parallel configurable communication systems, and time pressure [127]. The way these elements are perceived can also be affected by the role i.e., designer or a user and must be taken as important features or building blocks shared by the games.

Previous studies have discussed some commonly used gaming elements such as awards/rewards, points, badges [65] [128], leaderboards (PBL) [66] [128], ranking, levels, quests, bets, avatars [69] [65], and stories [125]. Table 4.2 shows the description of some game elements used in gamification.

*Table 4.2 Game elements*

| Game Elements | Description |
|---|---|
| Points | A reward given in the form of points when a certain task is completed |
| Badges | A representation of achievements |
| Leaderboards | Leaderboards allow users to compare themselves in ranking |
| Ranks | A rank increases competence level with the top players, such as votes |
| Levels | A level is increased when users achieve certain number of points |
| Avatars | A virtual character of users having attributes that can be personalized based on interests |
| Quests | A story-based task that users must complete, presented as a quest |
| Bets | Like estimation where users bet on certain event |
| Awards/Rewards | Given to the player on completion of some task or a behaviour |
| Stories | A background narrative and dramatic sequence to arouse users' emotions |

Despite having a variety of game elements, previous studies reveal that points, badges, and voting systems are the most used game elements [65]. Some researchers believed that with a single gaming element, a learning experience can be gamified [120]. However, these elements are classified based on their structural characteristics and motivational significance [117], but these classifications do not help the designer to decide which pattern is well suited to solve user needs. As a result, they often choose a combination of these elements such as PBL [129], because of their easy implementation.

To design gamified systems, three main factors are attached to the games: rules, goal, and feedback system [125]. To play a game, some clear and easy rules should be assigned for fair play. Game rules are specifically defined for this gamified tool. They give directions to the users for correctly performing the desired tasks and explain how rewards are given upon completion of a task or activity.

## 2.3 Related Work on Gamification in Requirements Elicitation

Requirements are an integral part of the system [130] and require a lot more attention and interest of users, because they describe the behaviour, functionalities, and attributes of the system [67]. In recent years, gamification has grabbed attention in many applications due to making low-interest and unattractive tasks look more fun. It has also been used in RE to allow the users to express their goals and needs [77]. Because of the narrative context used in gamification, users can relate to the real-life examples rather than abstract concepts.

Gamification has been applied to the RE process that consists of elicitation, negotiation, prioritization, validation, and specification. However, specification and validation are the least explored areas in gamification. Gamification is used among virtual teams during the elicitation process [131]. In requirements elicitation, few aspects are gamified, yet there are many to be explored. Currently, the issues in practices do exist such as lack of technological support, reliability of results of gamified platforms, and evaluation problems.

A platform developed in 2012, named iThink [74], is a gamified tool used to increase stakeholder's participation and collaboration. iThink was based on the concept of *six thinking hats,* which is a method of creative thinking behind gamification. The players are rewarded if they generate new requirements, also if they refine any requirements. There are different roles involved in iThink including player/user and project manager. A project manager sets up the project, but this role is not considered as a player, thus it doesn't get any rewards. The game was implemented with the help of an agile platform named Outsystems Agile Platform. For the evaluation of iThink, two case studies were conducted: one took place at a childcare centre for revamping its information system, while the second case study was conducted in a classroom where students were asked to use the prototype information system for some course. iThink was engaging and motivating but according to the participants' feedback, it has drawbacks such as depending on how ideas are generated. Furthermore, the generalization of results was difficult because test samples were quite limited.

A similar work, to increase the collaboration and knowledge sharing among the RE team, was proposed with the name REVISE [130]. Requirement Elicitation and Verification

Integrated in Social Environment (REVISE) was based on cognitive theories and gamification elements. REVISE was developed using a set of CARE principles i.e., create, ask for review, review, and extend. The creator has the role of system analyst, the reviewer reviews the requirements, and the customer selects requirements and discusses them using collaborative environment. All stakeholders can add and trace new requirements. It was proposed that the scoring would be individual, and team based, where reviewers would be rewarded with additional scores. Similarly, gaming elements like leaderboards, badges and profiles would be used. As the proposed game would be based on distribution cognition and boundary objects, complete requirements document is mandatory as an input to the system. In this work, only theoretical framework was given.

The insufficient engagement of stakeholders leads to low quality requirements and poor system performance. The GREM – Gamified Requirements Engineering Model [71] was proposed to use gamification platform for requirements elicitation. The aim of GREM was to engage stakeholders in elicitation for improving performance of system. Therefore, it used variables including one independent variable i.e., gamification, and two dependent variables i.e., stakeholder engagement and performance. Dichotomous variables were used to measure these variables, and motivation was measured with Reiss Profile, and PANAS (Positive and Negative Affect Schedule) was used for emotions. User stories were used for requirements elicitation, whereas acceptance test was performed with the help of gaming platform. The platform was developed in WordPress and Captain UP API was used to embed different gaming elements such as PBL, levels, activity feeds, and challenges. For evaluation of GREM, a controlled experiment was conducted in a company involving 12 employees categorized into two groups based on gender and expertise, and motivation. The positive aspects of GREM were recorded as per participants' feedback and it was concluded that gamification helped in raising quality, productivity, and creativity in the system, yet the communication and collaboration among stakeholders was reduced, and might have negative effects. Furthermore, the choice of using different gaming elements has also an overall impact on stakeholder engagement and performance of the system. To represent requirements, more experiments using GREM were required because it was not evaluated by keeping in mind the engagement of stakeholders.

Similarly, to engage stakeholders in elicitation process, a digital platform for scenario based RE is developed using gamification [77]. The platform is based on three concepts i.e., engagement, performance, and gamification, and developed using WordPress and API named Captain UP. Three variables were used with two controlled variables of stakeholder expertise and motivation. To engage the user, three sub-dimensions were defined including emotions, cognition, and behaviour whereas performance was further categorized into creativity and quality. The platform was developed with 17 gaming elements and to measure different elements like emotions etc., Reiss profile test, PANAS, and Flow Shor Scale (FSS) were used. Points and badges were rewarded to the user after submission of user stories. For evaluation of platform, an IT firm was selected for conducting experiments. It was observed that behaviour created more statistical differences than emotions. Stakeholders who were exposed to gaming elements were more active in requirements production, thus user engagement was high during elicitation. Also, gamification indirectly had an impact on user performance, which derived a change in the behavioural dimension. Despite having active participation, it was concluded that keeping track of user stories was difficult because of having too many people involved, the sample size was small, and only interested employees took part in the experiment. It was also observed that extrinsic rewards are effective but in short term use, however long-term effects of using gaming elements are unknown.

For prioritizing requirements and stakeholder engagement in RE, a tool supported collaborative requirements prioritization process is presented [132]. The tool named DMGame used gaming elements to involve stakeholders in effective decision making. The tool was developed using genetic algorithms and Analytic Hierarchy Process (AHP). For prioritization of requirements, a set of requirements was considered for inclusion in the next iteration. The people involved in prioritization were stakeholders having different roles (supervisor, opinion-provider, and negotiator) and significance. Progress, time-pressure and pointsification were used as game elements. Moreover, Java was used on the server side, relying on Spring Framework. The tool has been validated on three industrial case studies taken from SUPERSEDE project.

A generic framework for acceptance requirements was designed using gamification [127]. The purpose of the framework was to model, analyse and fulfil acceptance requirements with the help of gamification. The Agon framework was composed of four models namely acceptance model (AM), tactical model (TM), gamification model (GM), and user context level (UCM) integrated at three different levels. AM, TM, and GM were designed by extending Non-Functional Requirements Framework (NFR), whereas UCM was based on Context Dimension Trees. With 270 goals and 376 relationships among refinements, operations, and contributions, it was validated on a Meeting Scheduler example. Different gaming elements were used, including points, leaderboards, and ranking. As it was performed on an already developed example, therefore no thorough evaluation was performed. It was observed that Agon was a large model and could become larger. More complex and real case studies were required to confirm the generality, utility, and versatility of Agon. Later, Agon and Motivational Antecedents Framework (MAF) were compared and performed on the same case study of Meeting Scheduler [134]. Both frameworks were compared based on concepts, tools, and gamification techniques. MAF was designed on the principles of Human and Organizational Behaviour (HOB), while Agon design was drawn from Software Engineering. Agon advocates user perspective, software, cognitive and psychological aspects for user engagement. On the other hand, MAF is more inclined towards social behaviour such as work environment etc. and is conceptual. Agon in RE framework provides gamification solution for acceptance requirements. Similarly, a meta-model for Agon was described [135] using a Systematic Acceptance Requirements Analysis Framework based on gamification, to support analysis and design of software systems. A Participatory Architectural Change Management in ATM Systems (PACAS) was introduced for Air Traffic Management (ATM) system. The meta-models of all four models used previously [133] were defined and composed of psychological strategies and gaming solutions. It was concluded based on preliminary evidence that Agon was a useful framework, as evaluated by both non-experts such as students, and experts of RE and gamification (from PACAS).

While designing gamified solutions, acceptance and motivational aspects are needed. For this purpose, stakeholders are required to participate in the process. A work [136] presented

in 2019, discussed set of key requirements to design a gamification solution. The two major components of the system were design thinking and Agon framework. To identify key requirements, different models and existing case studies including SUPERSEDE and DMGame [132] were analysed.

A systematic literature review is conducted to explore gamification in requirements elicitation [137]. The purpose is to know suitable game elements for requirements elicitation. Initially 525 studies were collected out of which 48 studies were selected as primary studies. In SLR, other than PBL, 18 suitable game elements were studied. Interaction with other users and privacy risk are some of the challenges of gamified systems.

Requirements elicitation using gamification strategies is discussed and deployed [138]. The system gives incentives to the users based on their performance. Two surveys are conducted: one with a gamification system, and one without it, to study the impact of gamification on requirements elicitation. Game elements such as PBL, activity feed, avatar, and quest are used while conducting surveys. It is concluded that although gamified strategy helps to engage users yet communication between participants should be improved. After a change management is performed, its impact assessment is not performed.

Table 5.2 presents some of the gamified platforms, models, and tools in requirements engineering, with the contribution they have made, and game elements used.

*Table 5.2 Previous studies on gamification in different phases of RE*

| Source | Proposed Work | Game Elements | Phases of Requirements Engineering | | | | |
|---|---|---|---|---|---|---|---|
| | | | Elicitation | Analysis | Specification | Validation | Management |
| 2023 [139] | • Gamified approach to improve the understanding of reqs. gathering by students<br>• Use of multiple methods including survey, empirical evaluation, observation, and brainstorming | Time, challenges | ✓ | - | - | - | - |
| 2020 [140] | • Reqs. discussion game RE-PROVO<br>• Evaluate gamifying RE activities through developed game in legacy replacement to | Badges, points, roles, challenges | - | ✓ | - | - | - |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | encourage innovation and minimizing unnecessary changes | | | | | | |
| 2019 [141] | • A gamification approach GARUSO<br>• Stakeholders from outside of organizations were involved in the RE process | Points, levels | ✓ | ✓ | ✓ | ✓ | ✓ |
| 2019 [79] | • A gamified approach to make RE easier to use and learn.<br>• Offer more games to RE activities.<br>• Presented idea is still under evaluation | Feedback | ✓ | ✓ | ✓ | ✓ | ✓ |
| 2018 [142] | • Gamified requirement inspection<br>• Ring-i process to allow users and other stakeholders for verification of i* models | Rules, goal, and feedback system | - | ✓ | - | - | - |
| 2017 [135] | • A gamified framework SARAF in reqs. analysis and designing engaging systems.<br>• Also guide analyst in acquiring acceptance reqs. | Badges, paths, leaderboard, points, roles, avatars, rewards, challenges | ✓ | ✓ | - | - | - |
| 2017 [132] | • DMGame to engage and motivate stakeholders in requirements prioritization.<br>• for decision making | Progress, time pressure, and pointsificatio n | - | ✓ | - | - | - |
| 2017 [143] | • Crowd centred RE method to engage stakeholders in the process of requirements engineering.<br>• With the help of REfine tool | Roles, points, leaderboard, group formation, and exploration | ✓ | ✓ | ✓ | ✓ | ✓ |
| 2016 [78] | • A new method of reqs. elicitation and analysis AUCD<br>• For user participation | Points, rules | ✓ | ✓ | - | - | - |
| 2016] [73] | • GREM, a model to engage more stakeholders.<br>• To improve the performance of RE | PBL, levels, challenges, and activity feeds | ✓ | - | - | - | - |
| 2015 [73] | • Web-based gamified platform to involve users | Leaderboard , badges, points, rewards | ✓ | - | - | ✓ | - |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 2015 [77] | • Developed online gamified platform for scenario based RE<br>• Selected user stories with scenarios from behavioural driven development method<br>• Performed controlled experiment | Points, badges, leaderboard, levels, challenges, avatar, activity feed, progress, quiz, time, prize etc. | ✓ | ✓ | ✓ | ✓ | - |
| 2015 [144] | • CCRE in Software Product Organization,<br>• Prototype tool named REfine used to involve stakeholders in RE | Leaderboard , points, and roles | ✓ | ✓ | ✓ | ✓ | ✓ |
| 2015 [130] | • REVISE, a tool for requirements elicitation and verification | Score, badges, and leaderboards | ✓ | ✓ | - | - | - |
| 2012 [74] | • iThink, a game-based tool for collaboration, to gather requirements.<br>• Creative thinking technique 'Six Thinking Hats' was also used. | Points (scoring) | ✓ | - | - | - | - |

Table 5.2 shows that RE has been benefited using gamification, yet it requires awareness of participation. Existing work has some limitations such as less attractive platforms [144], ineffective results [106], absence of conceptual foundations [130], invalidity of developed solutions [145], theoretical framework without implementation and evaluation [130], and results biases [141].

### 2.3.1 Game Elements for Elicitation

Game elements are the driving force for gamified solutions in requirements elicitation. Literature is evident on the use of game elements for the purpose of engaging user to the system, as mentioned in above sections. However, in table 6.2 the following game elements have been identified according to their application in RE.

*Table 6.2 Identified game elements in different phases of RE*

| Studies | Game Elements | Requirements Engineering | | | | |
|---|---|---|---|---|---|---|
| | | **Elicitation** | **Analysis** | **Specification** | **Validation** | **Management** |
| | Points | 11 | 5 | 2 | 3 | 2 |
| | Badges | 6 | 3 | 0 | 1 | 0 |

| | | | | | |
|---|---|---|---|---|---|
| Leaderboard | 8 | 3 | 1 | 2 | 1 |
| Levels | 4 | 1 | 1 | 1 | 1 |
| Roles | 4 | 3 | 1 | 1 | 1 |
| Feedback | 1 | 2 | 1 | 1 | 1 |
| Challenge | 4 | 2 | 0 | 0 | 0 |
| Reward | 2 | 1 | 0 | 1 | 0 |
| Avatar | 2 | 1 | 0 | 0 | 0 |
| Paths | 1 | 1 | 0 | 0 | 0 |
| Group Formation | 1 | 1 | 1 | 1 | 1 |
| Explore | 1 | 1 | 1 | 1 | 1 |
| Rules | 1 | 2 | 0 | 0 | 0 |
| Progress | 1 | 1 | 0 | 0 | 0 |
| Score | 1 | 1 | 0 | 0 | 0 |
| Goals | 1 | 1 | 0 | 0 | 0 |
| Feed | 1 | 0 | 0 | 0 | 0 |
| Time/Time Pressure | 2 | 1 | 0 | 0 | 0 |

Table 6.2 presents 18 game elements in RE based approaches, tools, and systems. These game elements include points, badges, leaderboard, levels, roles, feedback, challenge, reward, avatar, paths, group formation, explore, rules, progress, score, goals, and feed.

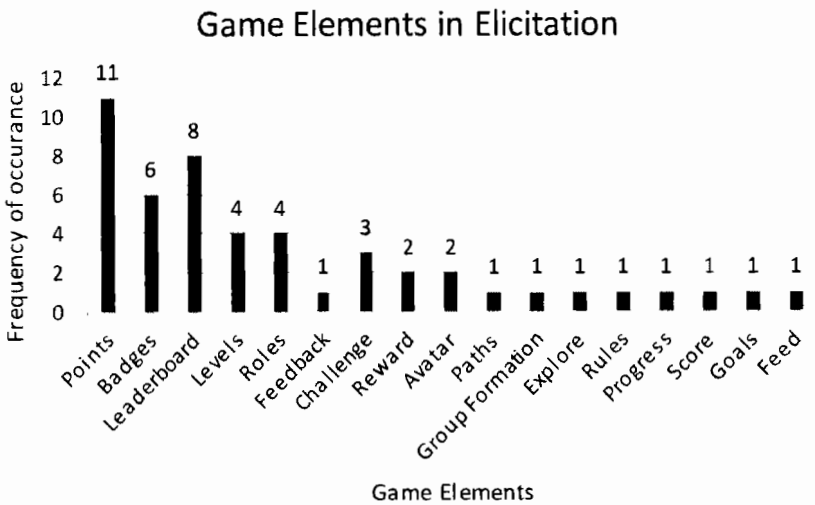



*Figure 7.2 Game elements in requirements elicitation*

In previous work on requirements elicitation, the frequency of occurrence of points is more than any other game element, that is, 68%, followed by leaderboard 50% and badges 37% as shown in figure 4.2.

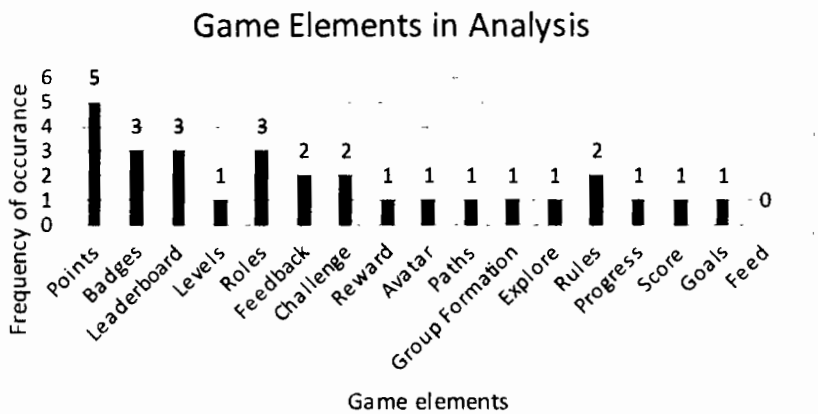



*Figure 8.2 Game elements in requirements analysis*

Figure 5.2 shows game elements used in requirements analysis. In requirements analysis, points are the most used game element with 31% of frequency of occurrence, whereas badges, leaderboard, and roles come with 18% each.

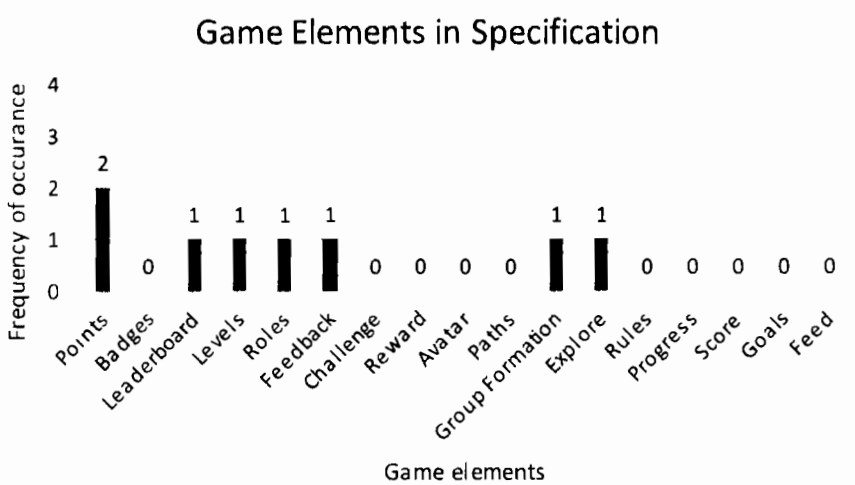



*Figure 9.2 Game elements in requirements specification*

Figure 6.2 shows game elements used in requirements specification. In requirements specification, points are the most used game element with 12% of frequency of occurrence.

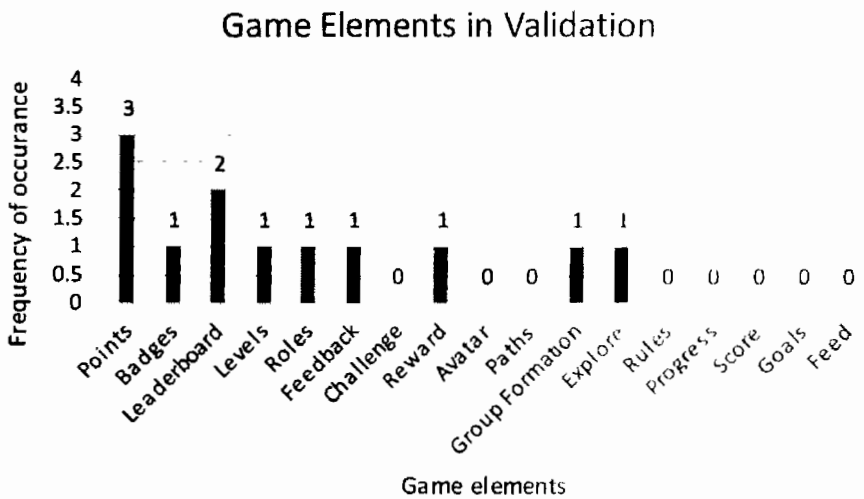



*Figure 10.2 Game elements in requirements validation*

Figure 7.2 shows game elements used in requirements validation. In requirements validation, points are the most used game element with 18% frequency of occurrence, followed by leaderboard.

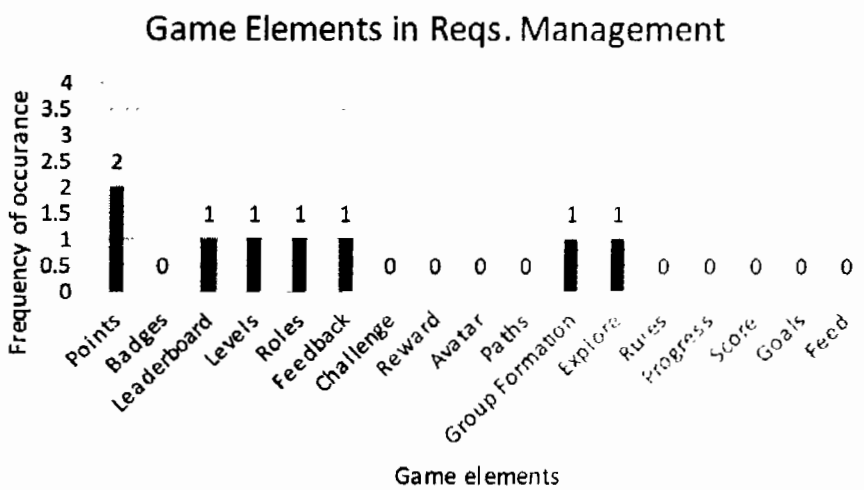



*Figure 11.2 Game elements in requirements management*

Figure 8.2 shows game elements used in requirements management. In requirements management, points are the most used game element with 12% frequency of occurrence. Literature shows that gamification is used in almost every activity of RE. The use of gamification in requirements elicitation is still highest in number. We have covered 16

studies of gamification in RE and identified 18 game elements including the popular ones such as PBL, levels, ranks, roles, rules, challenges etc. According to figure 1.2, points are the widely used game elements in requirements elicitation with 68% of occurrence, however, not the only game element used. Leaderboards are the second most used game element with 50% occurrence, and badges with 37% occurrence in requirements elicitation.

# CHAPTER 3

## *RESEARCH METHODOLOGY AND DESIGN*

# Chapter 3 Research Methodology and Design

## 3.1 Research Methodology

This chapter describes the research process and research methods adopted at each step of the process, along with the rationale for their selection. This study uses a confirmatory research approach [146] [147] to perform field research. Confirmatory research has a clear research question to test existing theories [148]. This is further used to derive a proposition about the intention of the study, selection of cases, and collection of data. In confirmatory studies, evidence is built when case studies also support the related theory. Confirmatory case studies, in this research, provide evidence that the gamified tool effectively reduces ambiguity by actively engaging users during elicitation, which is supported by empirical data from the industry. Moreover, the study involves the conducting of a literature survey to get an insight into the related work presented in literature. Secondly, gamification tool is designed. Gamification tool is developed and validated on two case studies from the industry. The feedback of users on involvement in a gamified tool is taken by conducting a survey. Finally, the results are analysed and concluded doing statistical analysis.

Following figure 12.3 shows the research methodology of study.

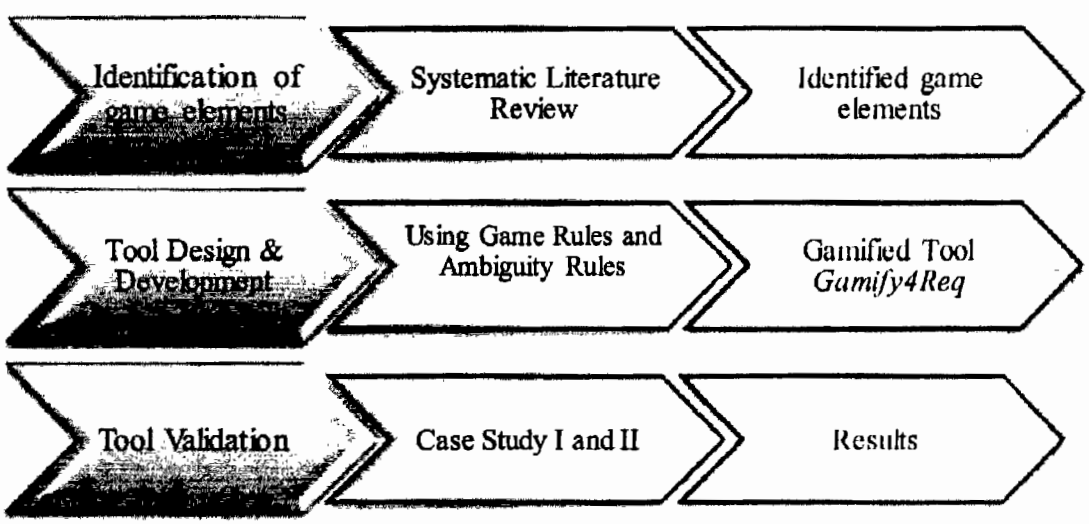



*Figure 12.3 Research Methodology*

## 3.2 Research Process

The research process comprises of different steps including SLR, tool design and development, validation of tool on multiple case studies, and feedback survey from the users. The SLR helps to identify game elements useful for gamified systems in reducing ambiguity in requirements. Another important step is to design a gamified tool for reducing requirements ambiguity by involving users during elicitation. The tool incorporates game elements and game rules. Based on the design, a web-based tool 'Gamify4Req' is developed to reduce ambiguity in requirements during elicitation. The tool identifies ambiguity in each requirement, by involving and engaging users, that helps to reduce ambiguity. On performing each task, the user is awarded with points and badges. Whereas other game elements are also used to engage users thorough the activity. The gamified tool performs three tasks. It: a) identifies ambiguity in requirements, b). involves users in the activity, and c). reduces ambiguity in requirements during elicitation. The tool validation is performed by conducting two case studies [149]. The RE teams and users provide the requirements using gamified tools, and results are then recorded. A feedback survey on user involvement in the tool is also conducted. After recording all results, they were analyzed and concluded.

Following figure 13.3 shows the research process of study.

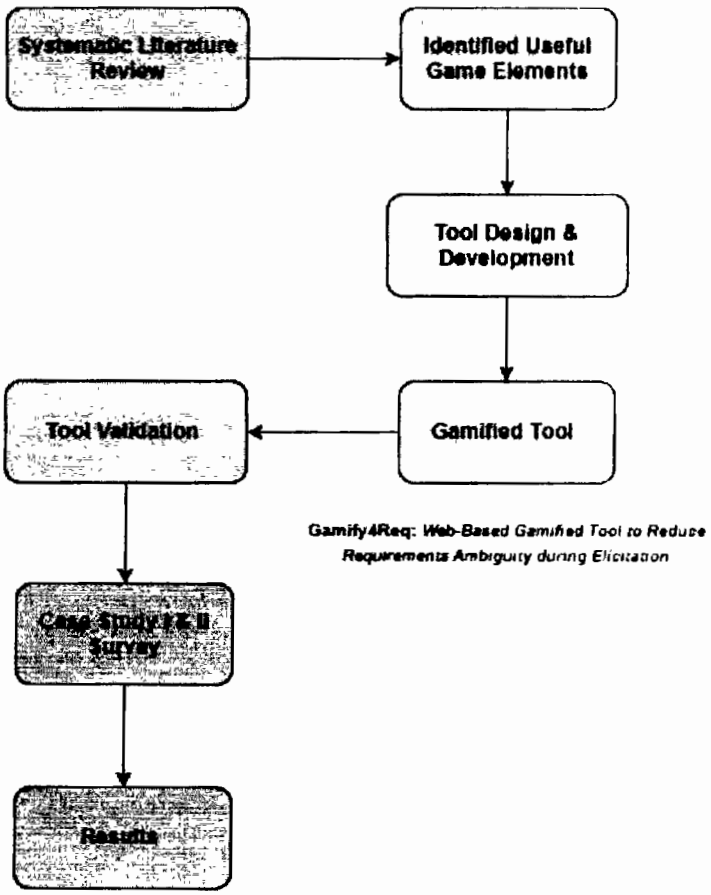



*Figure 13.3 Research Process*

In the following sections, each step of the research process is presented in detail.

### 3.2.1 Literature Review

The first step of the research process is a detailed literature review. Literature review is also a significant part of this study because the focus of the review is to study relevant work on requirements ambiguity, gamification, and identification of useful game elements that help to reduce requirements ambiguity during elicitation. All the articles and relevant material are accessed using computing databases such as IEEE, ACM, Springer, and Science Direct. These databases are selected because of their relevance and vast coverage of articles. The data is accessed in an organized manner by applying the following search queries to collect relevant studies. Therefore, the search strings used are:

a. (Requirements ambiguity AND requirements elicitation OR requirements gathering)

b. (Gamification OR game design OR gamified system AND requirements elicitation OR requirements gathering)
c. (Requirements ambiguity AND gamification AND requirements elicitation)

In the above-mentioned strings AND is used to connect different keywords, and OR is used for merging of two synonyms giving hundreds of results in Google Scholar. The period for the search in which papers are published is not restricted, therefore papers published up to 2023 are included. The papers which are not written in English language are excluded. Tutorials, abstracts, and posters are also excluded in the study.

### 3.2.2 Systematic Literature Review

The purpose of conducting SLR is to systematically identify, evaluate and analyse all relevant information pertaining to a specific research topic. SLR is an effective way for gathering comprehensive insights on existing studies, as suggested by Kitchenham [150]. SLR has gained popularity in the field of software engineering as researchers increasingly publish their findings using SLR.

There are numerous reasons for conducting an SLR. It involves a systematic method for reviewing literature, helps in identification and comprehensive analysis of relevant factors and data from previous studies. Researchers can ensure the possible coverage of related articles that have been previously published. This approach also emphasizes the retrieval of the most pertinent data, and on the selection of high-quality papers. SLR focuses on establishing a protocol that outlines the research questions to be addressed and provides an overall strategy for conducting a review. This strategy contains details such as maximum number of papers to be included, databases to be searched, and specific keywords and strings to be utilized. Additionally, inclusion and exclusion criteria are developed to select the articles from the SLR study.

In this study, one of the purposes of conducting SLR is to identify the empirically validated useful game elements other than commonly used game elements i.e., points, badges, and leaderboard or PBL, for requirements elicitation. Before conducting an SLR, we reviewed

the literature and identified game elements that are used in different phases of RE, as mentioned in section 3.2.1.

### 3.2.2.1 Process of SLR

The process of conducting an SLR involves nine main steps, which are further categorized into three phases. The first phase refers to 'planning' an SLR which involves formalizing the research objective, developing, and evaluating the review protocol. The second phase is 'conducting' an SLR, where studies are identified and selected articles undergo quality assessment; also, data is extracted and synthesized. The third phase is 'reporting' the findings of an SLR where the results obtained from the second phase are reported and discussed. Following figure 14.3 shows the nine steps under three phases of an SLR.

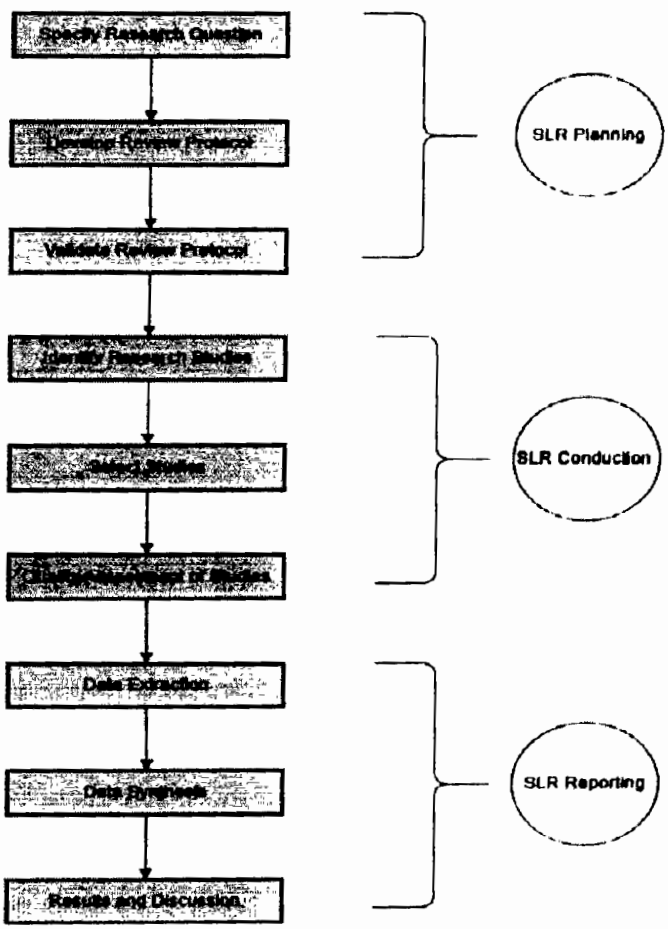



*Figure 14.3 Process of SLR*

Figure 11.3 illustrates the entire process of SLR, highlighting the nine steps. The process begins with the identification of the research question to be addressed in literature. This is followed by the development and validation of the protocol, which outlines the overall strategy, list of databases, keywords, and inclusion/exclusion criteria for papers. The subsequent steps involve the identification and selection of research studies followed by a thorough scrutiny of the studies based on quality assessment criteria. Finally, data is extracted and synthesized, leading to the final step of reporting the results. SLR is further presented in chapter 4.

### 3.2.3 Tool Design

A pro-active approach is used to reduce ambiguity in requirements using gamification. For this purpose, a gamified tool is designed for reducing requirements ambiguity using gamification during elicitation. It has four major parts: provision of user requirements, game elements and game rules, identification of ambiguity, and ambiguity rules.

Figure 15.3 presents a complete picture of the tool design.

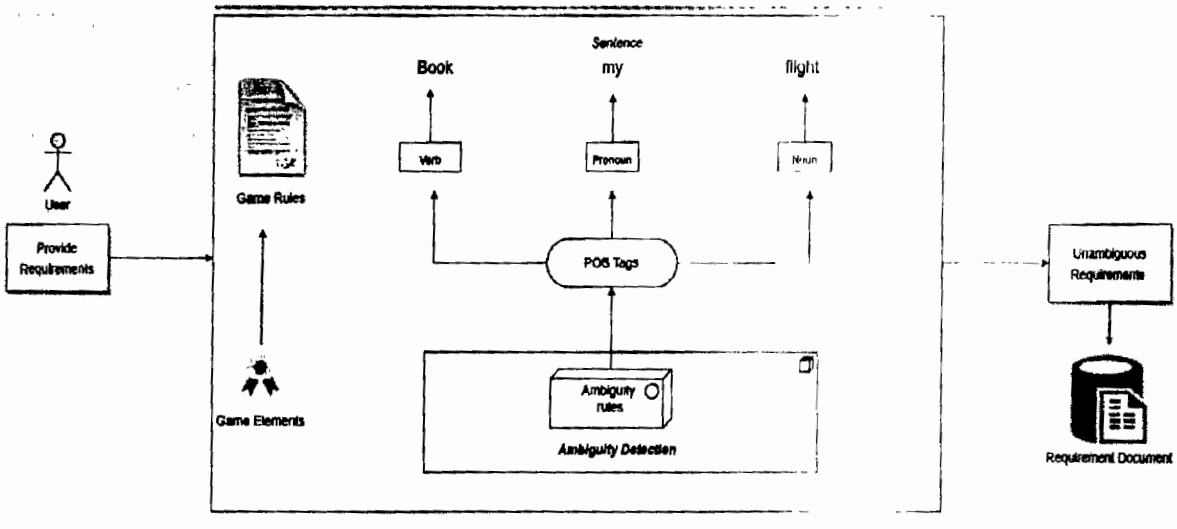



*Figure 15.3 Gamified tool design*

In figure 15.3 tool is designed comprising of four main parts.

1. Provision of user requirements
2. Game elements and game rules

3. Identification of ambiguity
4. Ambiguity rules

### Provision of User Requirements

Three user roles are involved in the tool. Project Manager (PM) setups the project, assigns roles to the project, and closes the activity. User (User/Customer/DE) and Requirements Engineer (ReqEngr.) provide requirements. Later these requirements are updated, verified, and validated by the users.

### Game Elements and Game Rules

Each user performs a certain set of tasks in the tool. For the involvement and engagement of users in the tool, game elements are used. These game elements are identified from literature. Game elements are used based on the activities and design of the tool. For instance, PM is managing the activity but not involved in providing requirements, thus the game elements assigned to PM are avatar, levels, and progress of the activity. Similarly, User/Customer/DE and ReqEngr. are involved in providing, verifying, and validating requirements, thus the game elements assigned to them are avatar, levels, points, badges, and leaderboard. These game elements keep users involved and engaged in the tool throughout the activity. The following table 7.3 shows user roles, set of tasks and game elements assigned to each role.

*Table 7.3 Game elements assigned to users*

| User Roles | Game Elements | Tasks |
|---|---|---|
| User/Customer/DE | Avatar, Points, Badges, Leaderboard, Levels | Profile completion, providing new requirements, updating requirements, validating requirements, reviewing requirements document |
| ReqEngr. | | Profile completion, providing new requirements, updating requirements, verifying requirements, reviewing requirements document |
| PM | Avatar, Levels, Progress | Profile creation, adding project, assigning roles, checking progress, review requirements document for ambiguity, close activity |

The gamified tool incorporates game rules to give directions to the users. Game rules specify how certain objectives are met and rewards are given.

*Table 8.3 Game rules*

| No. | Rules | Rewards |
|---|---|---|
| 1. | If ReqEngr. and User/Customer/DE etc. provide new requirements | Points on providing each requirement. Level completion badge. |
| 2. | If ReqEngr. and User/Customer/DE etc. update requirements | Points on each updated requirement . |
| 3. | If User/Customer/DE etc. validate requirements | Points on providing validated requirements. |
| 4. | If ReqEngr. verifies requirements in level 2 | Points on providing verified requirements. |
| 5. | If user/customer/DE etc. provides review of requirements document | Points on providing review. Task completion trophy. |
| 6. | If ReqEngr. provides review of requirements document | Points on providing review. Task completion trophy. |

Table 8.3 displays the game rules defined in gamified tool. Points and badges are given on performing tasks like provide requirements, update requirements, verify requirements, validate requirements, and review requirements document. Similarly, the leaderboard of each user is maintained alongside.

### Identification of Ambiguity

Another important part of the tool is to identify ambiguity in user provided requirements. When user provides requirements, the tool identifies ambiguity in the given requirement. Parts of Speech (POS) tagging [151] is used to identify ambiguous words. For instance, a sentence 'Book my flight' is provided by the user. The words 'Book' 'my' and 'flight' are assigned parts of speech 'verb' 'pronoun' and 'noun', as shown in figure 16.3 below.

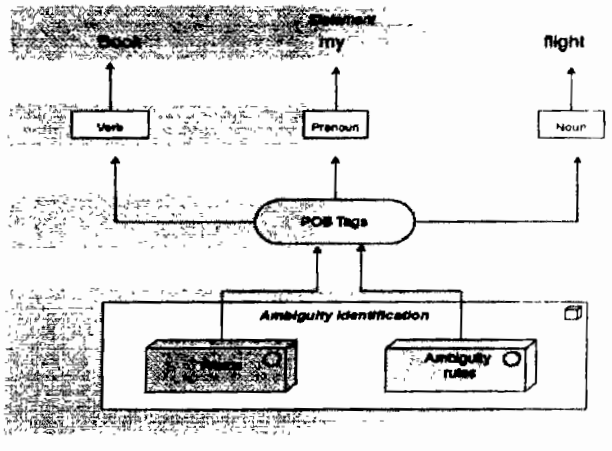



*Figure 16.3 Ambiguity identification*

In figure 16.3 POS tags uses dictionary of words containing nouns, pronouns, adverbs, conjunctions, and other man-made rules. However, for identification of ambiguity in given requirements, rule-based POS tags are used.

***Rule-Based POS Tags***

Rule-based POS taggers in NLP assign grammatical tags to words in a sentence based on a set of predefined rules. The rule-based approach relies on linguistic rules and patterns to determine the appropriate POS tags for words, allowing accurate analysis and understanding of requirements [151].

The rule-based POS tagger, based on defined rules, assign the POS tags, as shown in the given example of NL requirement statement.

Requirement: The system shall assign the course to the teacher

Tags: "The" (Determiner), "system" (Noun), "shall" (Verb), "assign" (Verb), "the" (Determiner), "course" (Noun), "to" (Preposition), "the" (Determiner), "teacher" (Noun).

In this example, the rule-based POS tagger successfully identifies and tags in the words based on linguistic rules. The implementation of rule-based POS tagger involves creating and finetuning rules based on grammatical patterns. These rules enable the tagger to accurately identify POS categories such as nouns, adjectives, adverbs, pronouns, prepositions, conjunctions, etc. in the text document, providing valuable information about the action or state expressed by the words. The specific rules used in the tagger depends on the linguistic requirements and the targeted POS category for analysis.

In our tool, when the user provides NL requirements, rule-based POS tagger assign respective parts of speech to each word in the sentence. Ambiguity rules and dictionary of words work together to identify ambiguity in requirement.

**Ambiguity Rules**

In gamified tool, POS tags identify each word and check whether the word is ambiguous or not. Semantic ambiguity in NL requirements occur when a word has more than one

interpretations in a given context [30] [35] for instance 'Aly stood on the bank' where *bank* is an ambiguous word in this context leading to multiple meanings [42]. For this purpose, it combines a dictionary of words and ambiguity rules [80] [152] to identify any ambiguity in the requirement. The rules are given in table 9.3.

*Table 9.3 Ambiguity rules*

| Ambiguity | | Rules |
|---|---|---|
| Referential | RA | Use of noun instead of pronoun throughout the specification, specially where two or more nouns precede the pronoun in same sentence. |
| Coordination | CA1 | Compound sentence containing two or more 'and' should be split into simpler, smaller sentences (without 'and'). |
| | CA2 | 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. |
| | CA3 | If a specification contains a conjunction ('and') and a modifier, and modifier is used for only one word, then split the sentence after 'and'. |

Ambiguity rules for semantic ambiguity are followed to identify and reduce ambiguity in requirements. Referential ambiguity in a requirement occurs when a word has more than one references in a sentence, for example 'The *truck* shall treat the *roads* before *they* close' [46]. In this example the word *they* has two references *truck* and *roads* which makes the sentence ambiguous. Similarly, coordination ambiguity occurs when more than one conjunction *and/or* is present in a sentence or when one conjunction is used with a modifier [30] thus making it a compound ambiguous sentence, for instance 'I saw Aly and Hina and Hira saw me' or 'Young boy and girl'. The rules as shown in Table 9.3 are designed for referential and coordination ambiguity sub-types.

### 3.2.4 Tool Development

The purpose of web-based gamified tool, Gamify4Req, is to facilitate the user, RE and RE teams to perform requirements elicitation from the User/Customer/DE and ReqEngr. Therefore, PM is the administrator of this system. The gamified tool has the following functionalities, that is, to:

- Provide requirements: *The requirements given by the user/customer/DE (Domain Expert) and requirements engineer.*

- Identify ambiguity: *Identification of ambiguity in given requirement by the gamified tool.*
- Verify requirements: *Verification of the given requirement by the requirements engineer.*
- Validate requirements: *Validation of the given requirement by the User/Customer/DE*
- Review requirements document: *Review of requirements document by the User/Customer/DE, ReqEngr. and PM, generated at the end of the activity.*

The gamified tool is web-based and multi-player involving User/Customer/DE, ReqEngr., and PM. Different game elements are used to involve and engage users in the system and motivate them to provide requirements. The game elements support each user according to their role in the system. Game elements for each role are used, for instance when a user provides the requirements, it is awarded with points; upon completion of major task, a badge is given; a leaderboard is maintained to display the user's points and achievements.

Similarly, when User/Customer/DE and ReqEngr. provide requirements, the system checks the requirements for identification of ambiguity using ambiguity rules, with the help of POS tags. At this stage, requirements are not considered from any other source other than the users. The User/Customer/DE provides, updates, validates, and reviews the requirements. The ReqEngr. provide, update, verify and review the requirements. PM registers all users, adds projects, assigns roles to the projects, track progress, and review the final requirements document.

### 3.2.5 Tool Validation

The validation of a tool is an important step in this study. The gamified tool is validated by two case studies. The purpose of selecting two case studies is to confirm that the gamified tool helps and supports requirements elicitation activity in ambiguity identification and reduction by involving users during elicitation.

Furthermore, confirmatory research is conducted to find out if the gamification theory is supported by the facts [153]. The goal of confirmatory research is to confirm the pre-specified relationship between reducing requirements ambiguity, by involving the users via gamification. It provides statistically significant results with definitive answers to the research hypothesis.

The validation of gamified tool 'Gamify4Req' is done by conducting multiple case studies in the software market of Pakistan. The type of case studies selected for validation are confirmatory in nature. In our research, confirmatory case studies helped to find out the extent to which gamification is supported by the industry.

### 3.2.5.1 Case Study Design

A case study represents the story of something interesting or unique stories about people, organizations, processes, events etc. [154]. In this work, a case study is selected as we want to study the phenomenon of reducing ambiguity in requirements during elicitation using gamification in real settings. Within the case study, cases are selected because of their effectiveness and relevance. Similarly, we have selected two case studies from small-sized software houses, and the case or unit of analysis is projects. There are six steps involved in designing and conducting the case study, as shown in table 10.3 below.

*Table 10.3 Case study design*

<table>
<tr><th>No.</th><th>Stages</th><th>Steps</th><th>Activities</th></tr>
<tr><td rowspan="4">1</td><td rowspan="4">Plan</td><td>Identification of stakeholders</td><td>Users/Customers/DE, requirements engineer, project manager</td></tr>
<tr><td>Finalization of case study topic, cases</td><td>Confirmatory case study, software projects</td></tr>
<tr><td>Identification of sources of information and from whom</td><td><table><tr><th>Sources</th><th>Information</th></tr><tr><td>User/Customer/ DE</td><td>Requirements elicitation + Validation</td></tr><tr><td>ReqEngr.</td><td>Requirements elicitation + Verification</td></tr><tr><td>PM</td><td>Initiation + Role assignment</td></tr></table></td></tr>
<tr><td>Following ethical standards<br>Required documents</td><td>• Local<br>• Requirements documents/SRS</td></tr>
<tr><td>2</td><td>Intervention</td><td>Software tool</td><td>• Use of tool to reduce ambiguity in requirements during elicitation.<br>• User involvement and engagement</td></tr>
<tr><td>3</td><td>Train Data Collectors</td><td>Identification and training of data collectors</td><td>• Training of the tool to perform elicitation.<br>• Data collection using gamified tool</td></tr>
<tr><td>4</td><td>Collect Data</td><td>Gathering of relevant data</td><td>• Each user uses the system.</td></tr>
</table>

| | | | |
|---|---|---|---|
| | | | • Data is recorded using the tool.<br>  |
| 5 | Analyse Data | Analysis of the results | • Analysis and review of collected data and results |
| 6 | Disseminate Findings | Report Dissemination | • Thesis dissertation write-up<br>• Publication of results |

In table 10.3, case study design is presented in six steps. Specific set of activities are performed with relevance to our study design. The following are the details of six steps of a case study as given in the table 10.3.

***Plan***

The first step involves planning a case study that includes identification of stakeholders, finalization of cases, and identification of sources of information. In selected case studies, PM and ReqEngr. are involved from RE teams, and User/Customer/DE are involved as clients to give requirements. Another important source of information is the requirements document. While conducting case studies, it is important to follow ethical standards which are applicable particular to the case. In both case studies, local ethical standards are followed.

***Intervention***

Next step in conducting case study is intervention. In this step, a gamified tool is used as an intervention. Firstly, Gamify4Req, a gamified tool, is developed to reduce ambiguity in requirements by involving users to the system. The tool is used by the participants involved in both case studies.

***Train Data Collectors***

In both cases, it is important to train the data collectors i.e., PM, ReqEngr., and User/Customer/DE. They are already identified during planning. One member of each team is trained to use the system, and they further train their team as well as the users involved in the system. Training is crucial and a time-consuming part of the study.

***Collect Data***

Collection of data is a significant step in both cases. Each RE team is requested to provide the details on requirements elicitation, ambiguity identification and reduction, and any

documents available. In both cases, a manual approach of collecting requirements is used. After collecting data using the existing approach, both teams are given the tool to elicit requirements from the user. Gamify4Req identify and reduce ambiguity in requirements and store all requirements in given template, which is downloadable in MS Word file.

***Analyze Data***

Once the data is collected, it is reviewed and analysed using graphs and statistical tests. The results are checked against the hypothesis of the study.

***Disseminate Findings***

After analysis of data, the findings are reported in the thesis. The results are also published as a part of the PhD degree requirement.

A proactive approach is used to reduce the ambiguity in requirements, provided by the users during elicitation. Moreover, the existing techniques are reactive because they first elicit requirements, document these requirements, manually detect the ambiguity, and manually remove it. However, our technique cannot be compared with the existing techniques present in literature.

### 3.2.6 Case Study I

Company Name: ABC

Details: The first case belongs to Pakistan's software market. ABC is a small level software house comprising of team with nine team members including project manager, product owner, software developers, technical report writers, designers, and others. The team works with Systematic Customer Resolution Unraveling Meeting (SCRUM) methodology, and team to client meetings are quite frequent. Requirements are elicited during these meetings in which the client is mostly not clear about the system, hence could not provide proper requirements. Due to uncertainty in requirements, the projects go over schedule and over the budget with compromised system at the end. Generally, requirements elicitation, documentation and ambiguity identification are manually performed by the product owner

in almost 2 months. It also includes requirements specification, verification, and validation. However, ABC has its own template for recording requirements. A project manager is involved in the activity throughout the project.

We have considered Project (P1) as a case study I for our research.
Project: Student Direction App (SDA)

Details: SDA is a mobile application for students to maximize their performance by reducing lengthy manual work. This application creates a real time interaction among administration, teachers, students, and parents/tutors with the chat box feature. Parents and teachers can stay well connected and updated about student's current performance. By using the digital diary, teachers can upload daily homework. A teacher can add an activity and a deadline for that activity as well as upload homework remarks, test marks, activity feedback of an individual student. It provides easy access to notes and helping materials uploaded by teachers and is time-saving for the students.

### 3.2.7 Case Study II

Company Name: XYZ

Details: This case also belongs to Pakistani software market. XYZ is a small level software house with seven team members. Included roles are project manager, senior developer, junior software developers, technical writers, and others. XYZ also works in agile methodology SCRUM, where the requirements elicitation process takes 2-3 months to get requirements. These requirements are collected in initial meetings. Senior developer and project manager review these requirements and contact the client if any ambiguity is found. Most of the time the client is not clear about the system, hence couldn't provide proper requirements. Due to uncertainty in requirements, the project is over scheduled and goes over the budget with a compromised system at the end. Once the senior developer finalizes the requirements, it is handed over to the technical writer for requirements specification. Like company ABC, XYZ has its defined template for requirements, in which requirements are specified as features. A project manager is involved in the activity throughout the project.

We have considered Project (P2) as a case study II for our research.

Project: GOTCHA-Depression Helpline (GOTCHA)

Details: *'Gotcha'- Depression Helpline'* is a mobile application that determines the depression level of the user via an assessment quiz (provided by an expert psychologist). It also recommends appropriate exercises according to the depression level of the user and monitors a patient's progress over time. It also features an app-to-app calling system which will allow the users to talk to the therapists via voice call. The therapist can provide talk therapy sessions which can play a vital role in the treatment of depression and anxiety in people.

Case studies are important part of the validation yet not the only part of it. After getting results from both cases, another important factor is measured i.e., user involvement in the tool. For this purpose, users engaged during requirements elicitation are asked to provide feedback by filling a survey questionnaire.

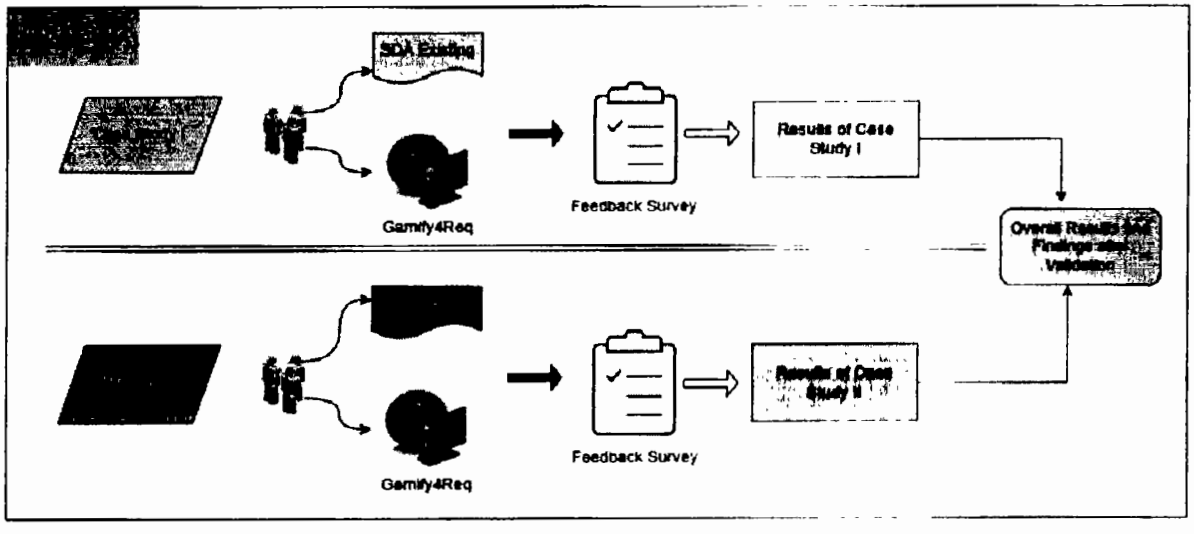



*Figure 17.3 Tool validation*

Figure 17.3 shows how both case studies are examined during validation. After that the feedback survey is conducted by the participants to know users' involvement in the activity. Finally, overall results are analysed and reported.

### 3.2.8 Feedback Survey

Measuring user involvement during requirements elicitation is an important part of this research. It is measured by conducting a feedback survey from the users involved in P1 and P2. The questionnaire instrument is designed by following the guidelines of Mark Kasunic [155], and divided into three sections including demographic information, game elements and game mechanic, and ambiguity identification and reduction. There are seven stages in the survey suggested by Kasunic, as shown in the figure 18.3.

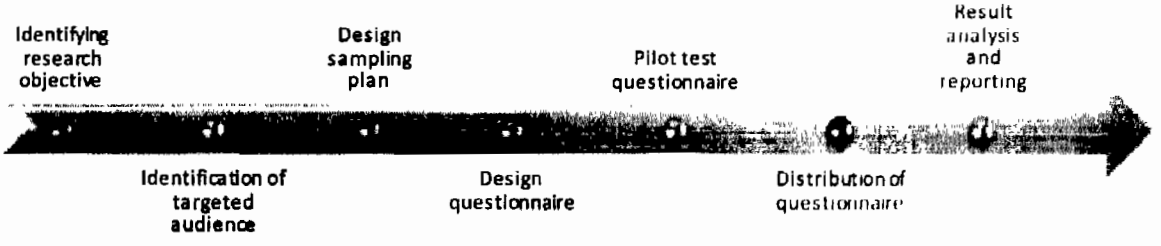



*Figure 18.3 Survey design*

The purpose of conducting this survey is to get user feedback on their involvement in the system and to know if a gamified tool reduced ambiguity during elicitation. Each section comprises of a different number of relevant questions. The questionnaire is self-administered, designed using Google Forms, and the link is shared with the users via email.

Figure 19.3 shows the mapping of survey steps of this study on the survey steps by Kasunic.

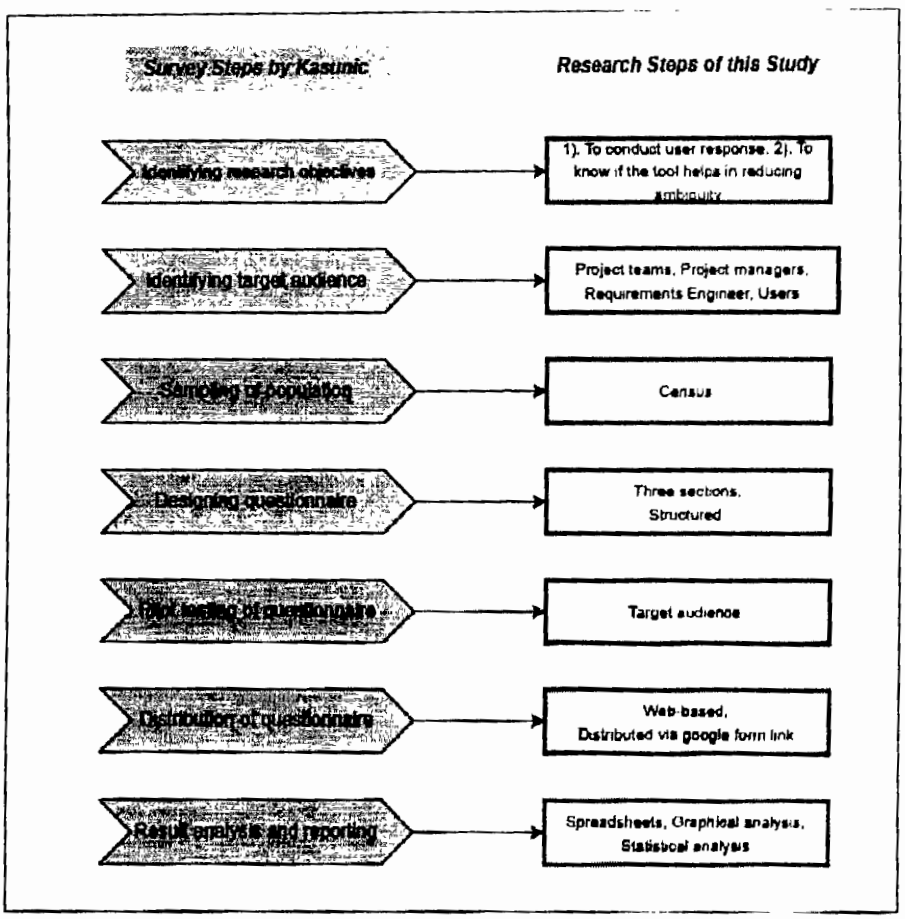



*Figure 19.3 Survey steps of the study*

### 3.2.7.1 Identification of Research Objectives

At stage one of conducting a survey, identification of the research objective is done. The objectives of this study have already been identified earlier in Chapter 1. Same objectives are considered while designing the questionnaire. However, the objectives of this questionnaire, which are in line with the objectives of this study, are defined. The objectives of survey are defined and are as follows:
Objective 1: To conduct user response on using the gamified tool.

Objective 2: To know if the gamified tool helped in reducing ambiguity in requirements during elicitation.

### 3.2.7.2 Identification of Target Audience

In stage two, the target audience is identified. In our study, RE teams and users are included in the special groups which are under study. It is also significant to know who is more suitable for providing the required information. In RE teams, project managers, and requirements engineers are identified, whereas to provide clients' feedback User/Customer/DE is identified.

### 3.2.7.3 Sampling of Population

In stage three, sampling of population is done. Generally, sampling is done to select who can be approached for filling the survey. In this study all individuals from the population must provide their feedback. We have selected census as our sampling strategy, which engages the whole population as a sample. Census helps to increase the precision and lessen sampling errors. Gamification is an emerging field which is not widely adapted yet; therefore, the population size is small and coverable.

### 3.2.7.4 Designing Questionnaire

In stage four, the questionnaire is designed and aligned with the research objectives. Firstly, questions are determined, the type and format of each question is specified, sequence of questions and overall layout is designed, and finally supplementary documents are developed and attached. The questionnaire is divided into three sections according to the objectives. The first section is about demographic information of the participants, in which different questions related to their personal information are asked. In section two, the use of game elements is focused upon, and they are asked about their experience of using the system. The last section three focuses on ambiguity identification and reduction questions. All questions are closed ended and Likert scale is used to get response on each question.

### 3.2.7.5 Pilot Testing of Questionnaire

A pilot testing of questionnaire is conducted to verify the questions from experts before distributing it among the participants. In stage five, the pilot test of questionnaire is done by

the target audience including project managers, requirements engineers and users. The questionnaire is modified after getting their suggestions and feedback.

### 3.2.7.6 Distribution of Questionnaire

In stage six, questionnaire distribution among respondents is done. Questionnaire is designed using Google Forms and its web link is shared through email. Web based surveys are easy to disseminate and reach maximum target audience in time. Keeping in mind the time and availability of participants, we limit the total duration of the questionnaire to 15-20 minutes.

### 3.2.7.7 Result Analysis and Reporting

In stage seven, data from all respondents is collected in an MS Excel sheet. Questions from each section are answered individually. However, data in raw form is processed into meaningful information in the form of graphs and charts. The Likert scale helped to cover more perspectives on the answer and provided complete coverage of each question. Data is then analysed and reported. User involvement is an important part of our research study therefore, it is significant to report the results in thesis dissertation.

# CHAPTER 4

## *SYSTEMATIC LITERATURE REVIEW*

# Chapter 4 Systematic Literature Review

## 4.1 SLR Planning

The initial stage of SLR involves the process of planning, during which all the necessary steps for conducting SLR are carefully outlined. The following section provides a comprehensive overview of our SLR planning, highlighting its key aspects.

### 4.1.1 Specify Research Question

The main objective for conducing SLR is to identify useful game elements, that are empirically validated for requirements elicitation. These useful game elements are identified from the literature. For this purpose, the formulated research question of this study is:

**RQ 2:** ***What are the game elements empirically validated for requirements elicitation?***

As discussed earlier in chapter 2, PBL are the most useful game elements in requirements elicitation. Besides PBL there are some other game elements that can be used in requirements elicitation.

### 4.1.2 Develop Review Protocol

The review protocol involves a comprehensive strategy of the steps involved in the SLR. It keeps the record of all the steps performed during SLR and helps in identifying relevant studies. Review protocol consists of the following steps:

#### 4.1.2.1 Keywords and Strings

The selection of appropriate keywords is quite significant in any research. The effectiveness of data retrieval relies heavily on the specific keywords identified during the research. After keen observation of the research question and objective of SLR, the most appropriate keywords of our research were determined.

The keywords of this study are 'game elements' and 'requirements elicitation'. Next step is to assign synonyms to the keywords to create a search string. The creation of search strings involves the process of combining keywords and synonyms. The objective of the research questions and SLR is considered while developing the string. Boolean terms OR and AND are utilized to create a string by effectively combining keywords and synonyms. This search string is helpful in extracting relevant data for our study, as shown in table 11.4.

*Table 11.4 SLR keywords, synonyms, and string*

| No. | Keywords | Synonyms | Search Strings |
|---|---|---|---|
| 1. | Game elements | Game components, Game features | (*Game components* OR *game features*) AND (*Requirements gathering method* OR *requirements discovery practice* OR *requirements collection*) |
| 2. | Requirements elicitation | Requirements gathering method, Requirements discovery practice, Requirements collection | |

Table 11.4 presents two keywords and related synonyms. We have created one string which is related to the research question and objective of this SLR.

### 4.1.2.2 Data Sources

The search string is implemented on scientific databases and digital libraries to get relevant studies. We have selected some renowned databases for conducting SLR, including IEEE, ACM, etc. Table 12.4 provides the list of digital databases and libraries used in this study.

*Table 12.4 SLR Data sources*

| Sr. No. | Digital Databases | Weblink |
|---|---|---|
| 1. | IEEE | https://ieeexplore.ieee.org/Xplore/home.jsp |
| 2. | ACM | https://dl.acm.org/ |
| 3. | Springer | https://link.springer.com/ |
| 4. | Science Direct | https://www.sciencedirect.com/ |
| 5. | Google Scholar | https://scholar.google.com/ |

Furthermore, study selection criteria are employed to address any potential redundancies in the retrieved data.

#### 4.1.2.3 Inclusion and Exclusion Criteria

The resources encompass a significant amount of data that may not be relevant to the research. Also, it is not practical to include all the papers obtained as a result of running the string on digital databases. Furthermore, it is not feasible to examine each paper in detail to determine its relevance to our research. In the review protocol of the SLR, we established a specific inclusion and exclusion criteria to filter out the papers that should be included in the study. The inclusion and exclusion criteria are given below.

- We selected those articles that are related to the domain of software engineering, but also focused on requirements engineering and gamification.
- Research articles and book chapters are included.
- To cover recent advancements and trends within our domain, we selected publications from the year 2015 and onwards.
- To conduct this SLR, we specifically chose the articles available in the HEC (Higher Education Commission) repository, freely accessible and openly available.
- The articles where an abstract is available but full text is unavailable are excluded.
- Articles written in English are included, whereas papers written in other languages are excluded.
- Position papers, posters and letters are excluded.

### 4.1.3 Validate Review Protocol

Review protocol is validated to ensure fairness during the collection of research articles. The validation criteria proposed by Kitchenham is used to validate our protocol. Additionally, expert researchers in the field have also verified the validity of the research protocol.

## 4.2 SLR Conduction

After SLR planning, next phase is the conduction of SLR. In this phase, the focus is on identifying the articles that need to be studied. Initially, a significant number of articles may appear from various databases based on the keywords and string. This is considered the primary data. To refine and narrow down this primary data, a selection procedure is applied

to identify the most relevant articles. A quality criterion is also applied to assess the selected studies, ensuring that only high-quality articles are included in the final analysis. The data extraction and synthesis process specifically target the papers selected after undergoing the quality assessment.

### 4.2.1 Identify Research Studies

Systematic search strategies are implemented to collect relevant research articles for this study, aiming to ensure an unbiased selection of articles. This method of a systematic search strategy differs SLR from a traditional literature review. All primary studies relevant to addressing the research question are identified from a diverse data source. The query string is executed across the selected digital databases, generating a list of articles that are subsequently subjected to selection criteria for inclusion.

Following the review protocol, query is executed on different digital libraries as specified earlier. The outcome of query is recorded and presented in a table format. These represent the candidate articles collected from each source in response to the string. Table 13.4 presents the count of articles retrieved from each digital library against the string.

*Table 13.4 SLR initial screening*

| Sr. No. | Data Sources | Initial Screening |
|---|---|---|
| 1. | IEEE | 50 |
| 2. | ACM | 37 |
| 3. | Springer | 38 |
| 4. | Science Direct | 30 |
| 5. | Google Scholar | 45 |
| **Total** | | **200** |

After running the query string on selected databases, 200 articles are selected during initial screening, as shown in table 13.4. During initial screening, 50 articles from IEEE, 37 articles from ACM, 38 articles from Springer, 30 articles from Science Direct, and 45 articles from Google Scholar are selected. This step is followed by the selection of studies for conducting SLR.

### 4.2.2 Select Studies

After the initial screening of research articles have been completed, the articles undergo a filtration process for the selection in the study. Multiple filtration levels of study selection are implemented. The first level involves assessing the total number of results retrieved from the query string, according to the inclusion and exclusion criteria. The duplicated articles are excluded from the list. The selected articles are further filtered through quality assessment. The same selection criteria are applied on the research question. The filtration levels of selecting studies are outlined as follows:

1st filter: In 1st filter, abstract and keywords are selected based on the relevance to research question.

2nd Filter: In 2nd filter, duplicate articles are removed from the selection.

Final filter: In final filter, the articles from 1st and 2nd filters are included only.

Table 14.4 presents multiple levels of filtration and selection of studies.

*Table 14.4 SLR filtration and selection of studies*

| Sr. No. | Data Sources | 1st Filter | 2nd Filter | Final Filter |
|---|---|---|---|---|
| | | | RQ 1.1 | |
| 1. | IEEE | 19 | 12 | 8 |
| 2. | ACM | 5 | 2 | 0 |
| 3. | Springer | 8 | 6 | 3 |
| 4. | Science Direct | 4 | 2 | 1 |
| 5. | Google Scholar | 15 | 7 | 6 |
| **Total** | - | **51** | **29** | **18** |

During 1st filter, 51 articles are selected among which 19 articles from IEEE, 5 articles from ACM, 8 articles from Springer, 4 articles from Science Direct, and 15 articles from Google Scholar are gathered. In 2nd filter, 29 articles are selected among which 12 articles are collected from IEEE, 2 articles from ACM, 6 articles from Springer, 2 articles from Science Direct, and 7 articles from Google Scholar are collected. Similarly, the final filter extracted 18 articles for the research question. The final filter results in collecting 8 articles from IEEE, 3 articles from Springer, 1 article from Science Direct, and 6 articles from Google Scholar.

The schematic filtration process extracted 18 research articles for research question.

### 4.2.3 Quality Assessment of Studies

The selected research articles are further assessed for quality. For this purpose, a quality assessment (QA) checklist by Kitchenham is followed. This checklist helps to check whether articles should be included in SLR or not. The checklist is used to check if the aims, findings, techniques and methodology, results, and conclusions are clearly mentioned in the article. The QA checklist consists of questionnaire with predefined answers. The answer to any question the options of 'Yes' and 'No' are given, where 'Yes' is further assigned the value '1' and 'No' is assigned a value of '0'.

## 4.3 SLR Reporting

Reporting is the final phase of SLR in which the consolidated information is used to answer the research question. We have focused on the contribution of the SLR to investigate the domain and new findings related to our study.

### 4.3.1 Data Extraction

Once the selection of studies is completed, all the collected studies are read according to the predefined criteria. The studies that provided relevant information to answer the research question are then selected. The data extraction template is used to extract data from each study.

We aim to find the useful game elements that are empirically validated for requirements elicitation, other than the PBL. Therefore, the information extracted from each primary study is shown in the table 15.4.

*Table 15.4 SLR data extraction*

| ID | Reference | Author | Year |
|---|---|---|---|
| 1 | [156] | Garcia et. al. | 2020 |
| 2 | [157] | Carolina et. al. | 2018 |
| 3 | [158] | Gafni et. al. | 2018 |
| 4 | [159] | Shih et. al. | 2017 |
| 5 | [160] | Brull et. al. | 2016 |

| | | | |
|---|---|---|---|
| 6 | [161] | Hishamuddin et. al. | 2018 |
| 7 | [162] | Nasirzadeh et. al. | 2020 |
| 8 | [163] | Sailer et. al. | 2017 |
| 9 | [57] | Darejeh et. al. | 2016 |
| 10 | [164] | Toda et. al. | 2019 |
| 11 | [165] | Wee et. al. | 2019 |
| 12 | [71] | Lombriser et. al. | 2016 |
| 13 | [166] | Robson et. al. | 2015 |
| 14 | [167] | Ghaban et. al. | 2019 |
| 15 | [168] | Ferrer Conill | 2016 |
| 16 | [169] | Bertholdo et. al. | 2018 |
| 17 | [170] | Schobel et. al. | 2020 |
| 18 | [171] | Mese et. al. | 2019 |

The 18 primary studies are categorized to answer the research question, as shown in the figure 20.4.

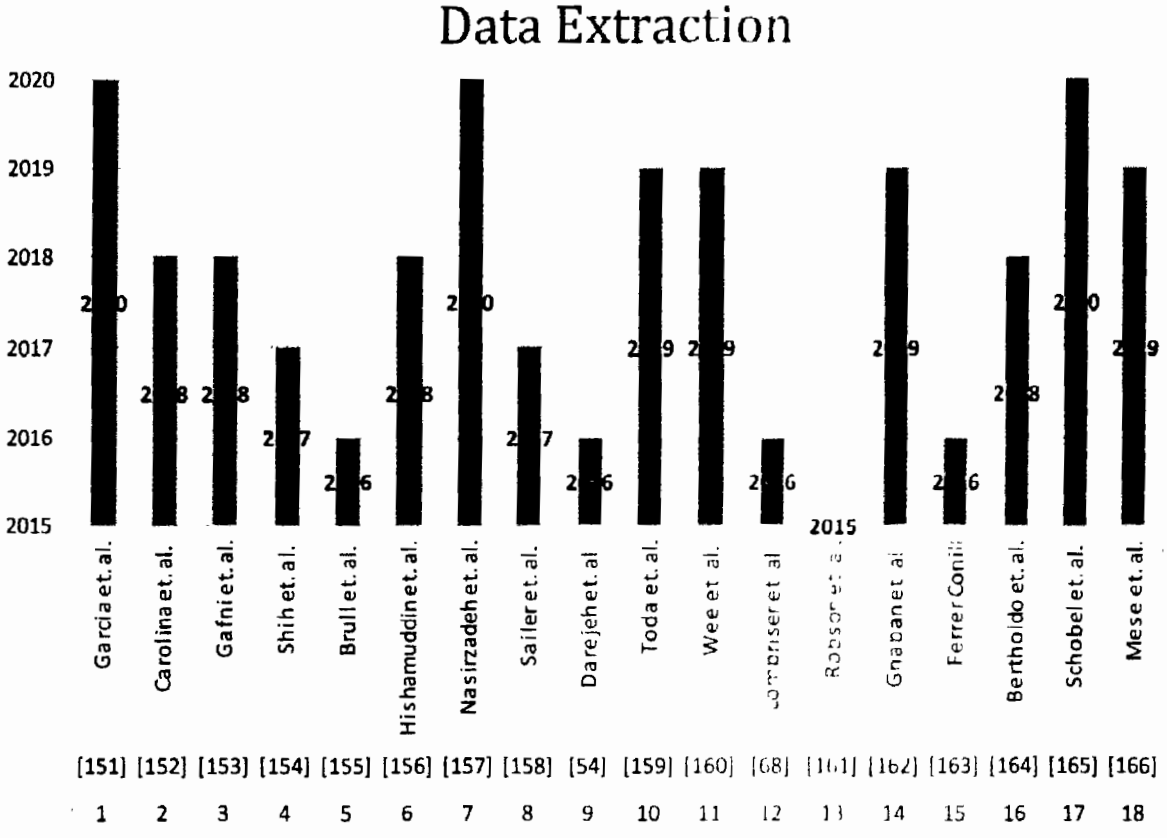



*Figure 20.4 SLR data extraction year-wise*

The year-wise data extraction of primary studies is shown in the figure 20.4. After data extraction, data synthesis is performed.

### 4.3.2 Data Synthesis

In data synthesis, the useful game elements other than the PBL are described. After reading 18 articles, 16 game elements other than the PBL are identified. These game elements can be seen in the table below.

*Table 16.4 SLR data synthesis*

| Sr. No. | Game Element | Reference | N=18 | Percentage |
|---|---|---|---|---|
| 1. | Trophy | [158] [166] [167] | 3 | 16.6% |
| 2. | Level | [156] [157] [158] [160] [162] [57] [164] [71] [166] [170] [171] | 11 | 61.1% |
| 3. | Medals and Scoreboards | [158] [167] [168] [170] | 4 | 22.2% |
| 4. | Virtual Goods | [158] [159] [162] [170] | 4 | 22.2% |
| 5. | User Profile and Storytelling | [157] [161] [163] [165] [71] | 5 | 27.7% |
| 6. | Activity Feed | [171] | 1 | 5.5% |
| 7. | Feedback | [156] [159] [161] [162] [163] [71] [170] [171] | 8 | 44.4% |
| 8. | Time Limit | [156] [170] | 2 | 11.1% |
| 9. | Achievement and Gift | [157] [166] | 2 | 11.1% |
| 10. | Quest | [157] [159] | 2 | 11.1% |
| 11. | Status | [157] [158] [71] | 3 | 16.6% |
| 12. | Penalty and count down | [162] | 1 | 5.5% |
| 13. | Performance | [163] | 1 | 5.5% |
| 14. | Progress | [162] [164] [71] [166] [169] [170] [171] | 7 | 38.8% |
| 15. | Mission | [156] | 1 | 5.5% |

| | | | | |
|---|---|---|---|---|
| 16. | Blocked Content | [156] | 1 | 5.5% |

In table 16.4, game elements collected from the primary studies. The frequency of game elements occurrence is mentioned in the field where total number of studies N=18. The game elements other than the PBL include trophy, level, medal and scoreboard, virtual goods etc.

### 4.3.3 Results and Discussion

In this section, the results after conducting SLR are presented and discussed. The research question is also answered to fulfil the aim of this study.

#### 4.3.3.1 Answering the Research Question

After data extraction and synthesis, 16 game elements are identified. According to table 6.4, the most useful game elements that are empirically validated for requirements elicitation are level (61.1%), feedback (44.4%), and progress (38.8%). However, other game elements have not frequently occurred during investigation. These game elements, other than the PBL, are useful in requirements elicitation. Furthermore, in Chapter 2 we have identified the game elements that are useful in requirements elicitation. According to our investigation, points (68%) are the widely used game elements, followed by the badges (50%) and leaderboards (50%).

Game elements play a significant role in gamification and considered as a key concept of gamification. The use of game elements is significant in gamified systems, for collecting user requirements. The involvement of users during requirements elicitation has a direct influence on the quality of requirements. Thus, using game elements is advantageous in enhancing user involvement within the system.

# CHAPTER 5

## *PROPOSED SOLUTION*

# Chapter 5 Proposed Solution

## 5.1 Tool Development

In this chapter, we describe how requirements ambiguity is reduced using gamification during requirements elicitation. We outline the tool i.e., Gamify4Req, based on the design presented in Chapter 3. In the next sections, game elements for all users are defined, also tool development is presented in detail.

### 5.1.1 Game Elements for the Users

Each user role has its set of responsibilities in the system; hence the motivation is different for different types of users. Similarly, in this system we have used game elements according to user roles and activities. The roles of User/Customer/DE and ReqEngr. are assigned the game elements of 'avatar', 'points', 'badges', 'levels', and 'leaderboard'. The PM is assigned 'avatar', 'levels', and 'progress'.

Table 17.5 shows the game elements used for user/customer/DE etc. respectively.

*Table 17.5 Tasks and game elements for Users/Customers/DE etc.*

| Tasks | Game Elements | |
|---|---|---|
| Registration and profile completion | Avatar | |
| | *Level 1* | |
| Providing new requirements | Points | |
| After providing at least 15 requirements | Points, Task completion badge | Leaderboard |
| | *Level 2* | |
| On updating and validating requirements | Points | |
| After providing 5 updated and valid requirements | Points, badge | Leaderboard |
| | *Level 3* | |
| Review requirements document | Points, Activity completion trophy | Leaderboard |

Table 17.5 shows the tasks of User/Customer/DE and game elements against each task and activity. User/Customer/DE completes profile and represents its presence by selecting an avatar. Level 1 unlocks as soon as User/Customer/DE completes avatar. In level 1, User/Customer/DE provides requirements. Each requirement carries 10 points. In level 1, at

least 15 requirements must be provided. The number of requirements is fixed at 15 for the purpose of validation only to help maintain the game balance. Too few requirements may make the activity short-lived or too many requirements may make the activity tedious. Moreover, each level of the game introduces different tasks related to requirements. Fixing requirements gives clear sense of progression to the users in an enjoyable way. After providing 15 requirements, User/Customer/DE is awarded with a badge. Leaderboard of User/Customer/DE is maintained alongside. In level 2, requirements can be updated, and validated by the User/Customer/DE if required. This number is fixed at 5 requirements to increase the complexity of Level 2 to make it more fun. Each validation carries points and on completion of level 2, a task completion badge is awarded. Similarly in level 3, User/Customer/DE reviews the requirements document generated by the tool. Points are awarded along with activity completion trophy.

*Table 18.5 Tasks and game elements for ReqEngr.*

| Tasks | Game Elements | |
|---|---|---|
| Registration and profile completion | Avatar | |
| | *Level 1* | |
| Providing new requirements | Points | |
| After providing at least 15 requirements | Points, Task completion badge | Leaderboard |
| | *Level 2* | |
| On updating verifying requirements | Points, badge | Leaderboard |
| | *Level 3* | |
| Review requirements document | Points, Activity completion trophy | Leaderboard |

Table 18.5 shows game elements of ReqEngr. ReqEngr. completes the profile and represents its presence by selecting an avatar. Level 1 unlocks as soon as ReqEngr. Completes an avatar. In level 1, ReqEngr. provides requirements. Each requirement carries 10 points. In level 1, at least 15 requirements must be provided. After providing 15 requirements, ReqEngr. is awarded with a badge. Leaderboard of ReqEngr. is maintained alongside. In level 2, requirements coming from the User/Customer/DE are verified and checked. Verification of each requirement carries points and on completion of level 2, a task completion badge is awarded. Similarly in level 3, ReqEngr. reviews the requirements document generated by the tool. Points are awarded along with activity completion trophy. The leaderboard is updated after each reward and all achievements are displayed on the

leaderboard. As the system is multi-player, thus the leaderboard not only displays user individual score but also the score of other users of the system.

In table 19.5 game elements of PM including avatar, levels, and progress of the activity are shown. The system has initially only one PM.

*Table 19.5 Tasks and game elements for PM*

| Tasks | Game Elements | |
|---|---|---|
| Registration and profile completion | Avatar | |
| | *Level 1* | |
| Add project titles and description | | |
| Assign roles | | Progress |
| | *Level 2* | |
| Review requirements document | Activity completion trophy | Progress |

PM tasks are different during requirements elicitation, as compared to the tasks of ReqEngr. and User/Customer/DE. However, major tasks include adding projects, assigning roles, managing users, and reviewing the requirements document once the whole process is completed.

### 5.1.2 Requirements Specification Guidelines

It is important to mention the guidelines of specifying requirements in NL to the users so they may provide accurately specified requirements. These guidelines aim at specifying requirements of gamified tool. These guidelines are inspired from IEEE guide to software requirements specifications [172]. Therefore, the following guidelines are designed to specify requirements.

*Table 20.5 Requirements Specification Guidelines*

| No. | Requirements Specification Guidelines |
|---|---|
| 1 | For functional requirements, use the word 'shall' |
| 2 | For statements of fact, use the word 'will' |
| 3 | For goals, use the word 'should' |
| 4 | Avoid the words 'are', 'is', 'was', 'must', 'support', 'etc.', 'and/or', 'not limited to' |

Users provide requirements according to the guidelines provided in table 20.5. For the input of functional requirements into the gamified tool, each requirement must use the word 'shall'

for example 'The customer shall register to the system'. The requirements presenting statements of fact must use the word 'will' such as 'The system will track user's daily steps to provide feedback on their activity levels'. Similarly, the requirements showing goals of the system must use the word 'should' such as 'The system should motivate users to practice on a regular basis'. However, to avoid ambiguity at first place the words like 'is', 'are', 'was' etc. must be avoided in requirements.

### 5.1.3 Tool Specifications and Architecture

The architecture of web-based Gamify4Req-Tool is presented in this section. Gamify4Req is a web-based graphical tool to elicit requirements from the user. For this purpose, PhP7.4 with Apache web server is used, as shown in the figure 21.5.

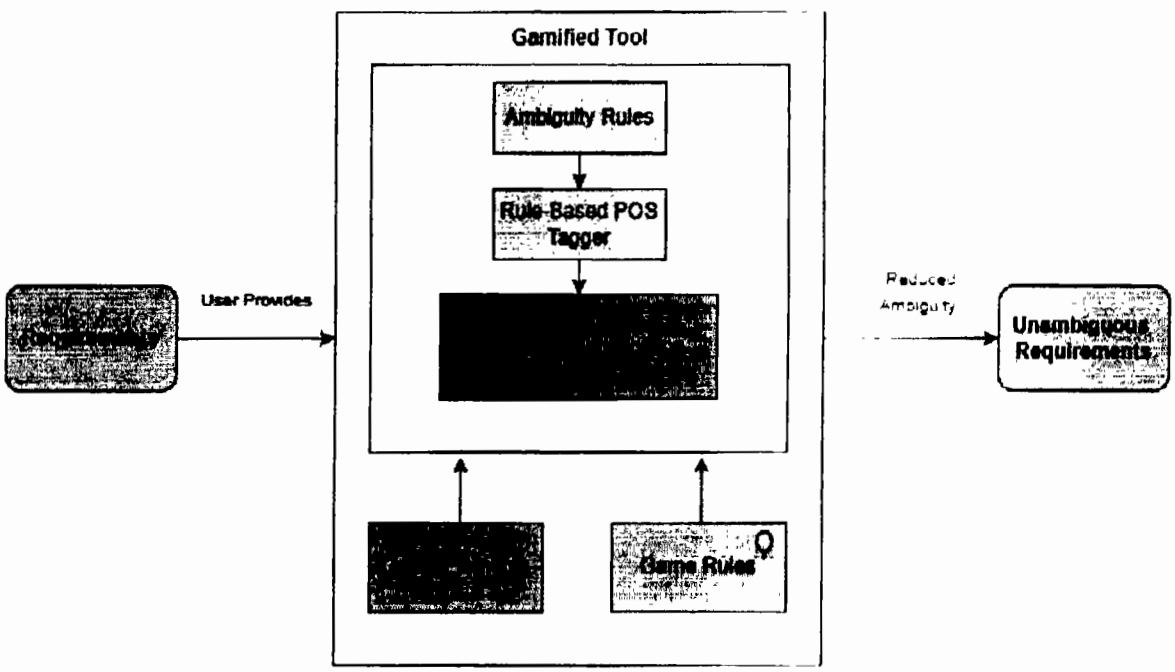



*Figure 21.5 Design of Gamify4Req*

For identification of ambiguity, rule-based POS tagging [151] [173] is used. MySQL database is used at the back-end. The details of tool specifications are shown in figure 22.5.

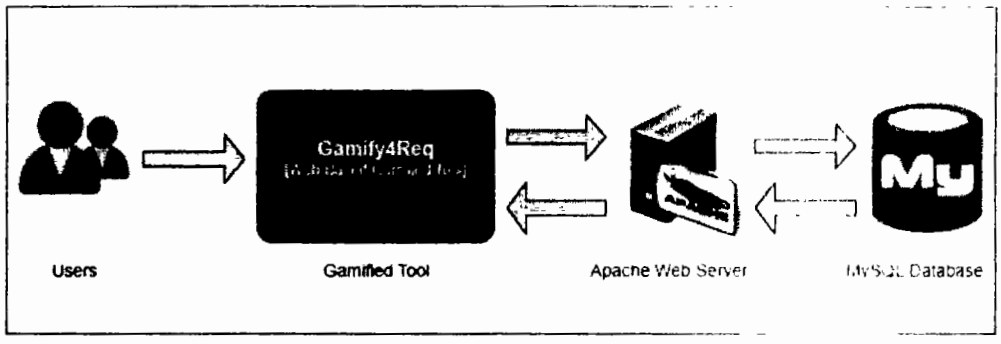



*Figure 22.5 Specifications of Gamify4Req*

Gamify4Req has three tier client-server architecture having presentation layer, application layer, and data layer.

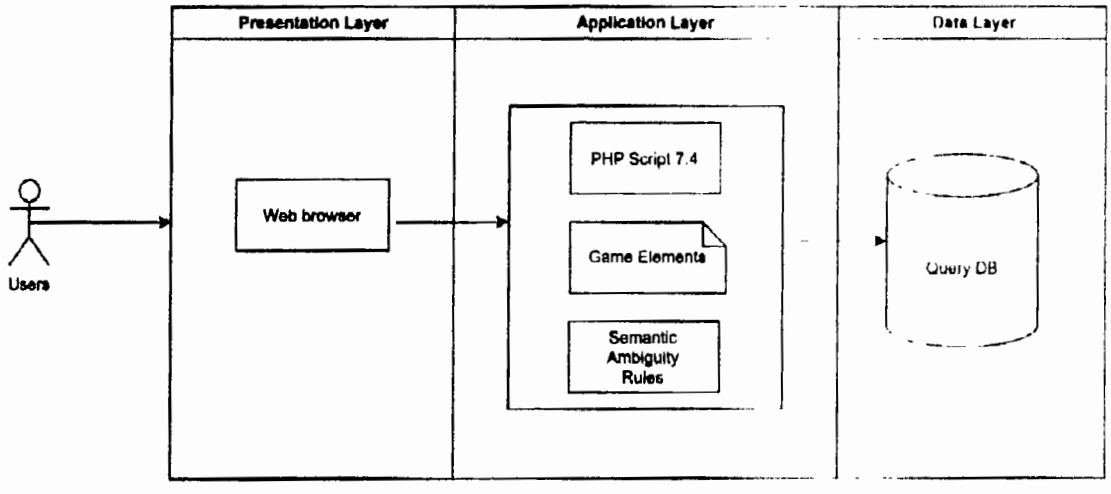



*Figure 23.5 Architecture of Gamify4Req*

The users tend to interact with the interface, which is linked to Apache web server, incorporated with the identified game elements, ambiguity rules, and connected to the database, as shown in figure 23.5.

### 5.1.4 Software Design Validation

For validity of software design and architecture, we shared a checklist of software design to experts who are working or have worked in local and global software houses. The design of Gamify4Req includes Unified Modelling Language (UML) Models, software architecture, requirements specification guidelines, ambiguity rules, game rules, and game elements.

Furthermore, we published the design of tool in IEEE conference [152] and got feedback on software design from two reviewers, which is incorporated in the system; details can be found in appendix A.

### 5.1.5 Gamify4Req – *Gamified Tool*

In this section, some of the images of PM dashboard are presented to present Gamify4Req in a glance. The following figures show the login page, dashboard, and some of the game elements. Gamify4Req has mainly three user roles performing their activities.


*Figure 24.5 Login*

Figure 24.5 displays the login for all the users. Each registered user must enter an email address and password to access the dashboard, whereas new users can create the account and proceed further after verification.

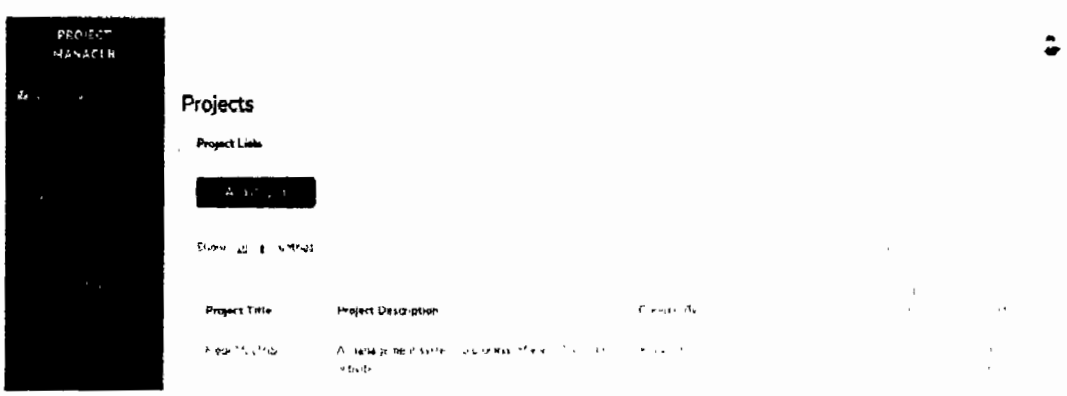



*Figure 25.5 Add projects*

Figure 25.5 shows that PM adds projects by selecting the option of 'Add Project'. The description of the project is also added and displayed to the PM.

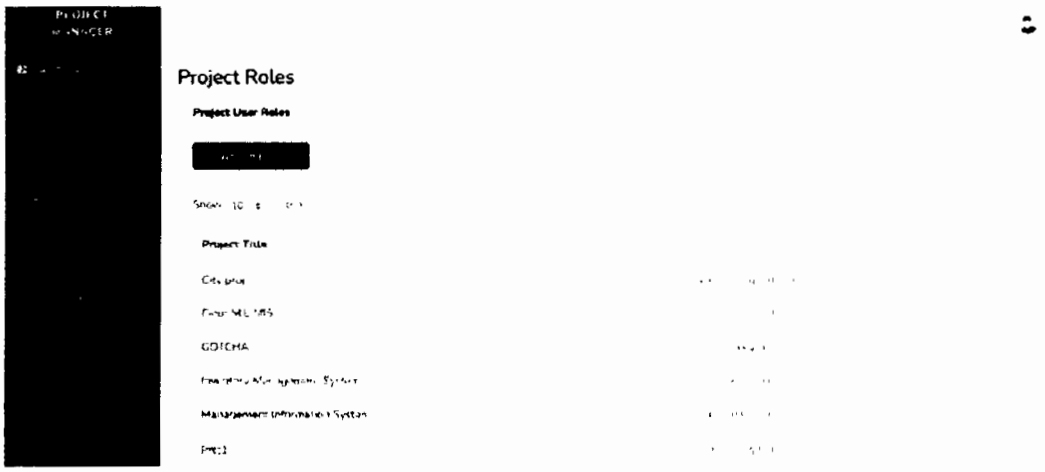



*Figure 26.5 Assign roles*

Figure 26.5 shows the project roles which are added and assigned by the PM. PM can also view the assigned users to the added projects.

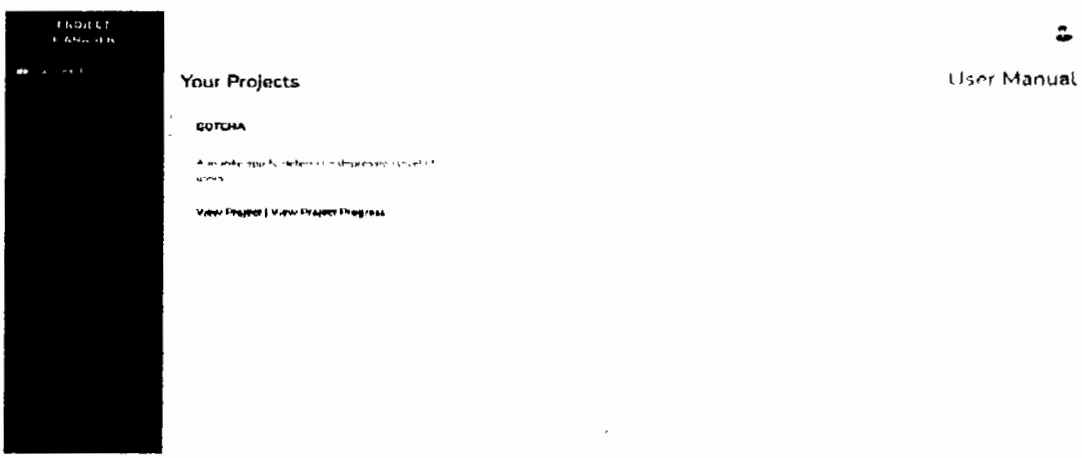



*Figure 27.5 PM dashboard*

Figure 27.5 shows the dashboard of PM, where PM can view ongoing projects, current users, assign roles, and check rules.

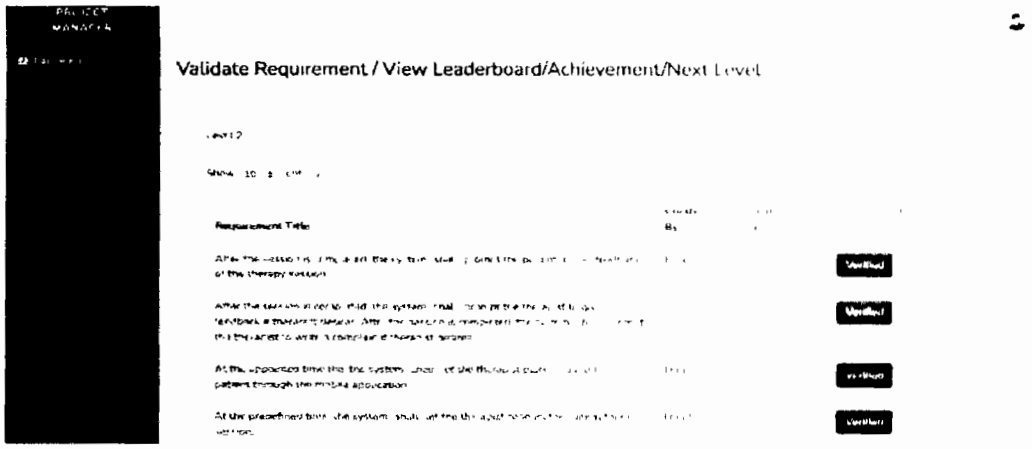



*Figure 28.5 Validate requirements*

Figure 28.5 shows all the requirements validated and verified by the users. PM can view the progress of ongoing activity and the project, as shown in figure 29.5.

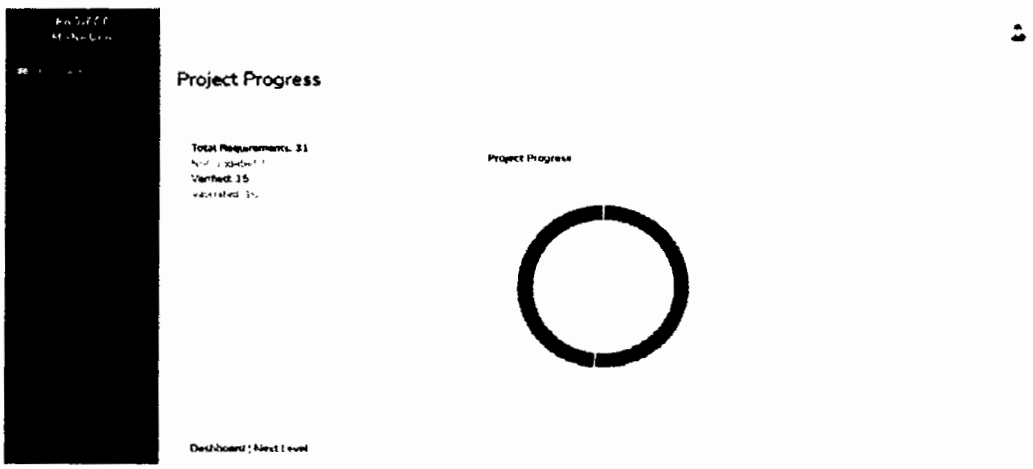



*Figure 29.5 Project progress*

Project progress helps to keep track of the progress in terms of collected requirements, verified, and validated requirements.

PM has mainly three game elements including project progress, levels, and avatars. In figure 30.5 level 1 is completed.


*Figure 30.5 PM Achievements*

Figure 31.5 shows PM leaderboard. PM can view username, designation, points, and badges achieved by each user under a project.

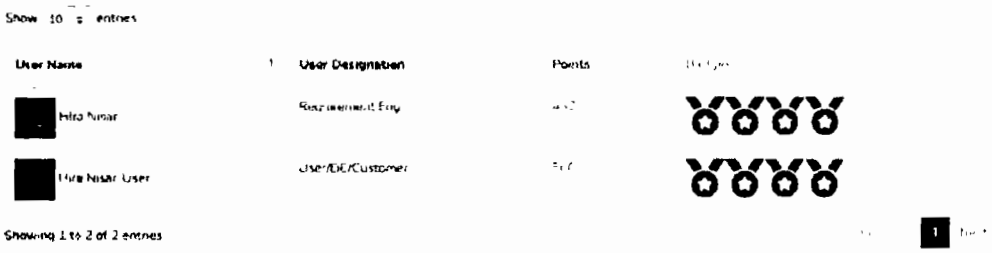



*Figure 31.5 PM leaderboard*

Furthermore, the user manual of the tool used by the users is presented in Appendix D.

### 5.1.6 Gamify4Req – *Working Example*

To demonstrate the functionality of our gamified tool Gamify4Req, we provide a dry-run example of its usage. The scenario is selected where the tool is reducing the ambiguity by involving users during elicitation.

1. Upon launching gamify4Req, User/Customer/DE and ReqEngr. are presented with the personalized profile pages where they first select the avatar of their choice, as shown in the figure 32.5.

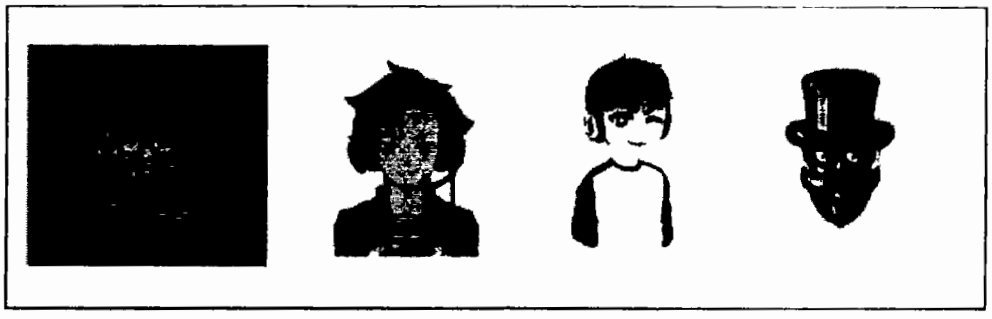


*Figure 32.5 Gamify4Req Avatar Selection*


*Figure 33.5 User Personalized Page/ dashboard*

Figure 33.5 shows the personalized page/dashboard of the users, where the assigned projects are listed.

2. The User/Customer/DE and ReqEngr. provides requirements and the tool checks the given requirement for ambiguity, as shown in the figure 34.5.

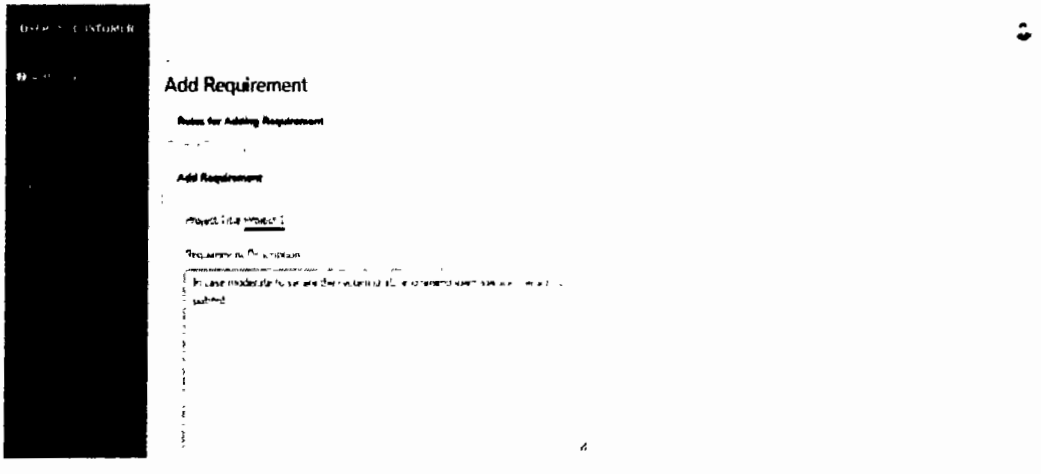



*Figure 34.5 Providing requirement*

The tool identifies the ambiguity present in the given requirement, as shown in the figure 35.5.

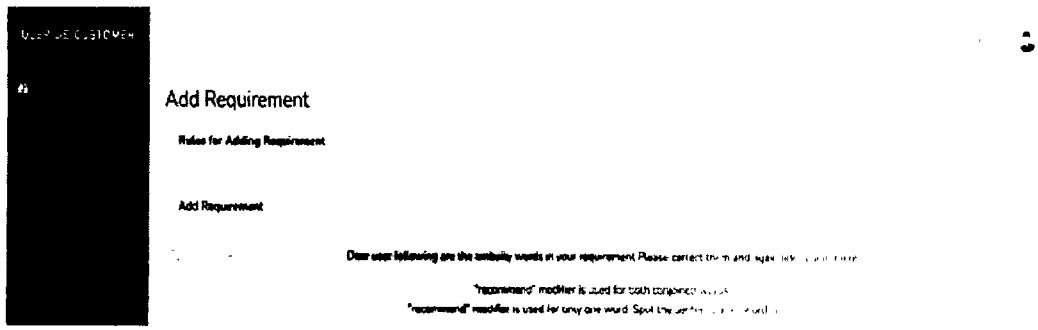



*Figure 35.5 Identifying ambiguity*

The tool highlights and categorizes the ambiguity so that the User/Customer/DE and ReqEngr. can easily remove the ambiguity and provide ambiguity-free requirement again.

3. The User/Customer/DE and ReqEngr. updates, verifies, and validates the requirements
4. Each user has to provide 15 requirements in level 1. Once User/Customer/DE provides requirements, ReqEngr. verifies the requirements, as shown in the figure 36.5.

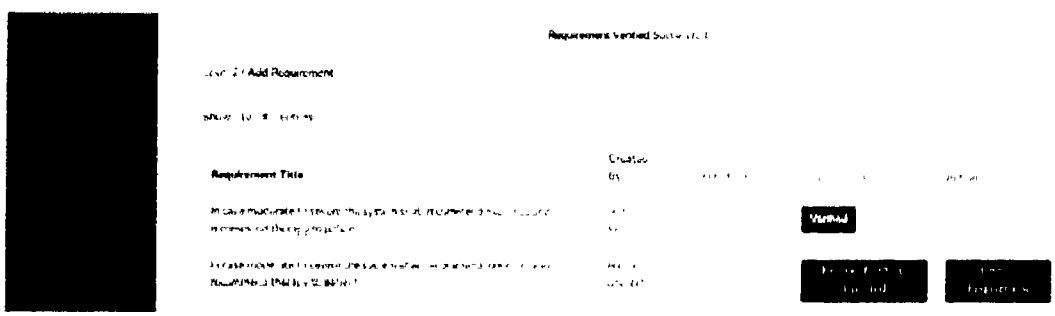


*Figure 36.5 Verify requirements*

5. User/Customer/DE also validates the requirements, once verification of requirements is completed, as shown in the figure 37.5

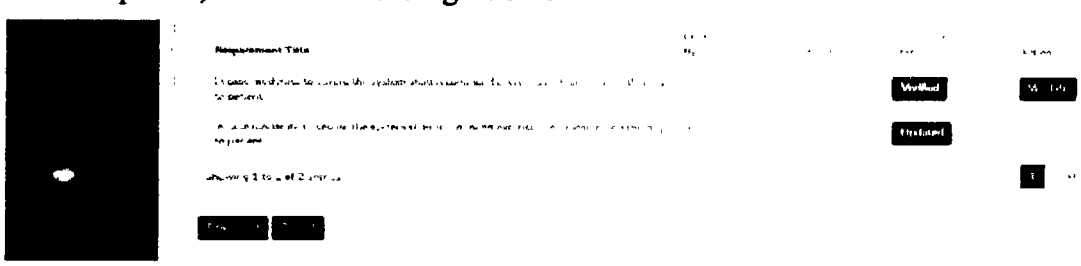


*Figure 37.5 Validate requirements*

6. As the users begins the activity, the gamified tool tracks the requirements

7. Throughout the elicitation, the tool displays points, badges, and levels to keep the users motivated and involved. Each user/Customer/DE and ReqEngr. are shown their leaderboard and achievement, as shown in the figure 38.5.

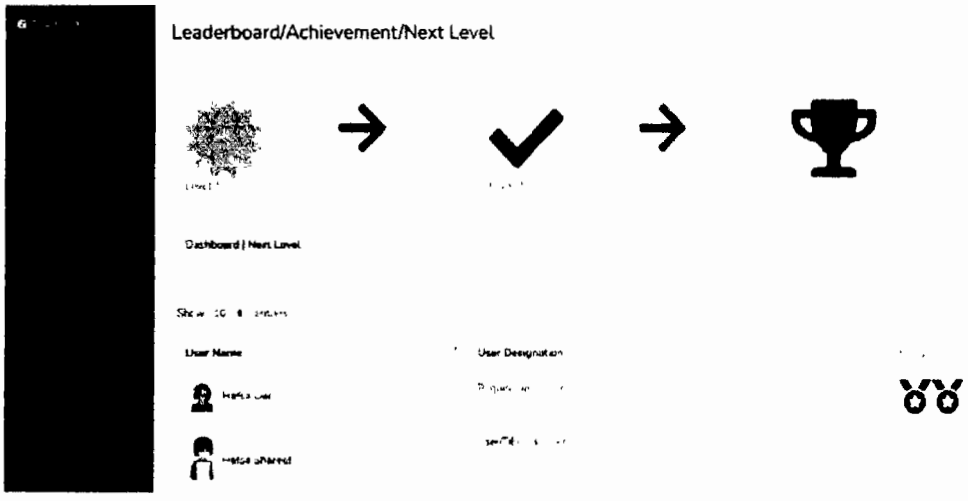



*Figure 38.5 Leaderboard*

8. All users can monitor their overall progress over time through the leaderboards
9. To ensure sustained user involvement, Gamify4Req introduces 3 levels of activities to maintain user interest and excitement.
10. After eliciting requirements, User/Customer/DE and ReqEngr downloads the requirements document to review the final requirements once again, as shown in the figure 39.5.

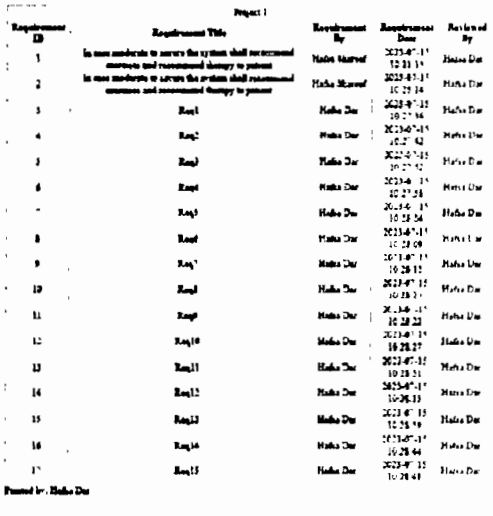


*Figure 39.5 Requirement Document*

11. User/Customer/DE and ReqEngr. view their achievement and progress on leaderboards, comparing their performance with other users and fostering a sense of healthy competition, as shown in the figure 40.5.

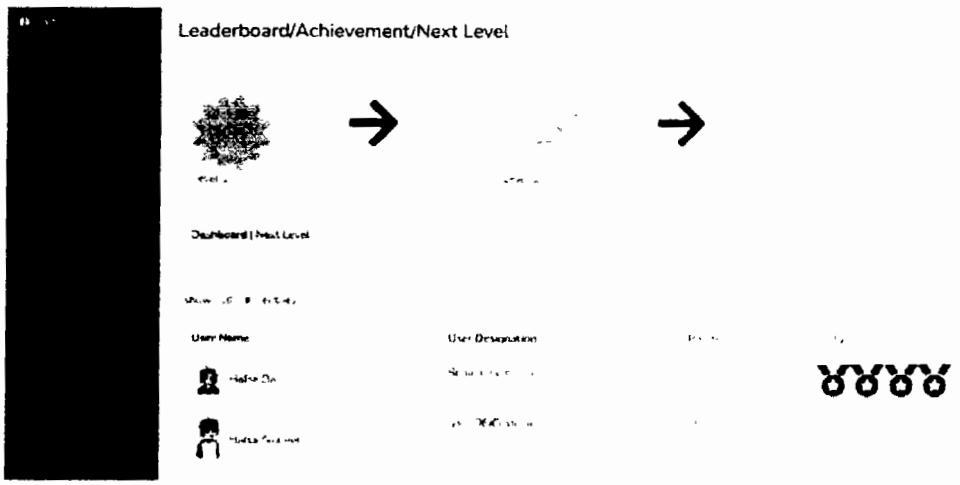



*Figure 40.5 Leaderboard for all users*

This dry-run example presents the potential functionality of our gamified tool, and user involvement in the tool in reducing requirements ambiguity during elicitation. By incorporating game elements, game rules, and ambiguity rules, the tool aims to create an enjoyable experience for the users. User manual of the tool is provided in Appendix D.

# CHAPTER 6

## *RESULTS AND ANALYSIS*

# Chapter 6 Results and Analysis

## 6.1 Results

This chapter presents the results of gamification tool Gamify4Req for reducing ambiguity in requirements during elicitation by involving and engaging users, and results of the survey collected from the users as their feedback on user engagement in the system.

Both case studies are analysed based on following factors:

1. Total number of requirements
2. Time taken to identify an ambiguity
3. Total number of ambiguities
4. Time taken to resolve/reduce an ambiguity

The above-mentioned factors help to analyse and compare approaches used in both cases. It is important to know the total number of elicited requirements, time taken to identify an ambiguity in an existing and proposed approach, and the total number of identified and reduced ambiguities.

### 6.1.1 Results from Case study I

In Chapter 3, it is presented that P1 is selected from case study I. For the comparative analysis of the results, an existing approach for reducing ambiguity in requirements has also been recorded and later compared to the results generated by the gamified tool. In the next section, results from the existing approach i.e., manual way of requirements elicitation, ambiguity reduction are discussed in detail.

#### 6.1.1.1 P1-SDA-Existing Approach

Previously in P1, for requirements elicitation 'interviews' are used as elicitation technique. The answers are recorded manually and documented on text editor. For identification and

reduction of ambiguity, documents are reviewed and if any ambiguity is found, it is discussed with the client, and updated in the document accordingly.

### 6.1.1.2 P1-SDA-Gamify4Req

Gamify4Req is used by the same RE team that elicited requirements using the existing approach in P1-SDA. In Gamify4Req, three user roles: PM, ReqEngr., and User/Customer/DE provide the requirements. Each user must provide at least 15 requirements. Gamify4Req is a multiplayer proactive tool and engages more than one user in a project(s). Currently, it allows only one PM to add projects and other related activities. The tool facilitates the team to elicit requirements and verify/validate requirements at the same time. It also generates requirements document once the activity is completed.

Gamify4Req identifies ambiguity right after a user submits a requirement. The tool identifies ambiguity and displays ambiguous words. User/Customer/DE is asked to remove the ambiguity and enter the correct requirement. ReqEngr. verifies each requirement, and user validates all requirements. As shown in table 1.6, requirements collected using both approaches are presented.

*Table 21.6 Requirements ambiguity in case I*

| No. | NL Req. | SDA-Existing | | SDA-Gamify4Req | |
|---|---|---|---|---|---|
| | | *Ambiguity Identification* | *Ambiguity Rules* | *Ambiguity Identification* | *Ambiguity Rules* |
| 1 | The school administration shall ***quickly*** register and login into the system. | X | X | ✓ | CA2 |
| 2 | The school administration shall enter the details of each user to register ***him.*** | X | X | ✓ | RA |
| 3 | Teachers shall be able to enter marks and ***upload*** the attendance of students. | X | X | ✓ | CA3 |
| 4 | The teacher shall be able to add an activity or task with ***its*** deadline | X | X | ✓ | RA |
| 5 | Teachers shall be able to upload helping material or notes of ***their*** course. | ✓ | X | ✓ | RA |
| 6 | Teachers shall be able to view inquiry requests of parents about ***their*** children. | X | X | ✓ | RA |

| | | | | | |
|---|---|---|---|---|---|
| 7 | Teachers shall be able to reply to complaints, ***and*** inquiries, ***and*** assistance requests through chat box. | X | X | ✓ | CA1 |
| 8 | Teacher shall be able to view the timetable to know ***his*** daily schedule | X | X | ✓ | RA |
| 9 | Students and parents shall be able to view ***their*** daily homework | X | X | ✓ | RA |
| 10 | Students and parents shall be able to view the timetable to know ***their*** daily schedule | X | X | ✓ | RA |
| 11 | Parents shall be able to make a complaint to the teacher about ***their*** child's performance | X | X | ✓ | RA |
| 12 | Parents shall be able to make an inquiry request to a teacher about ***their*** child's performance | X | X | ✓ | RA |

Table 21.6 displays identified ambiguities using existing and gamified approaches. User/Customer/DE removes the ambiguity in given requirement and provides the correct requirement. In P1-SDA-Existing approach, 41 requirements are elicited from the user and product owner. In manual ambiguity identification of requirements, the team identified ambiguity which is later discarded due to irrelevance to the scope of the tool. Using Gamify4Req, 41 requirements are elicited by the ReqEngr. and User/Customer/DE. In a similar way requirements are provided by the users. Once the activity is completed, the team downloads the requirements document for final review and approval. After the completion of the activity, the PM formally completes the activity on the tool.

Gamify4Req not only identified 12 ambiguous requirements but also categorized ambiguity type as well. According to the factors mentioned earlier, the tool identified ambiguity in given requirement in two seconds, and almost 35 seconds to resolve ambiguity.

### 6.1.1.3 Comparative Analysis of the Results from Case I

A comparison of results using both approaches i.e., existing and Gamify4Req, based on the factors given in table 22.6 are given below.

*Table 22.6 Comparative analysis of results from case I*

| No. | Factors | P1- Existing Approach | P1-Gamify4Req |
|---|---|---|---|
| 1. | Total number of requirements | 41 | 41 |
| 2. | Total number of ambiguities | 1 | 12 |
| 3. | Time taken to identify each ambiguity | 20 minutes | 2 seconds |
| 4. | Time taken to resolve/reduce an ambiguity | 5 minutes | 35 seconds |

The comparison in table 22.6 is based on the factors that have been identified earlier to compare the results of both approaches. It is evident that Gamify4Req produced far better results than the existing approach used in P1-SDA. Not only are more ambiguities identified in a short time, but also reduced in visibly less time. Using Gamify4Req, the tool identified 12 ambiguities during elicitation, out of which 9 belonged to RA, 1 to CA1, 1 to CA2, and 1 to CA3.

Using the existing approach, the team was unable to categorize the ambiguity type, but with Gamify4Req, the tool identified 12 ambiguities during elicitation, out of which 9 belonged to RA, 1 to CA1, 2 to CA2, and 1 to CA3. Table 3.6 presents the categorization of ambiguity of each rule.

*Table 23.6 Ambiguity categorization in case I*

| Users | P1-Existing Approach | P1-Gamify4Req | | | |
|---|---|---|---|---|---|
| | *Categorization* | *Categorization* | | | |
| | | RA | CA1 | CA2 | CA3 |
| ReqEngr. | X | 8 | 1 | - | 1 |
| User/Customer/DE | X | 1 | - | 1 | - |

Gamify4Req also categorizes each ambiguity according to its ambiguity type. In table 23.6, the breakdown of ambiguities is given according to the user roles who have identified and removed ambiguity in requirements.

The results of Gamify4Req shows a significant difference in identification and reduction of ambiguity. Similarly, figure 41.6 shows a graphical presentation of both approaches and number of identified ambiguities.

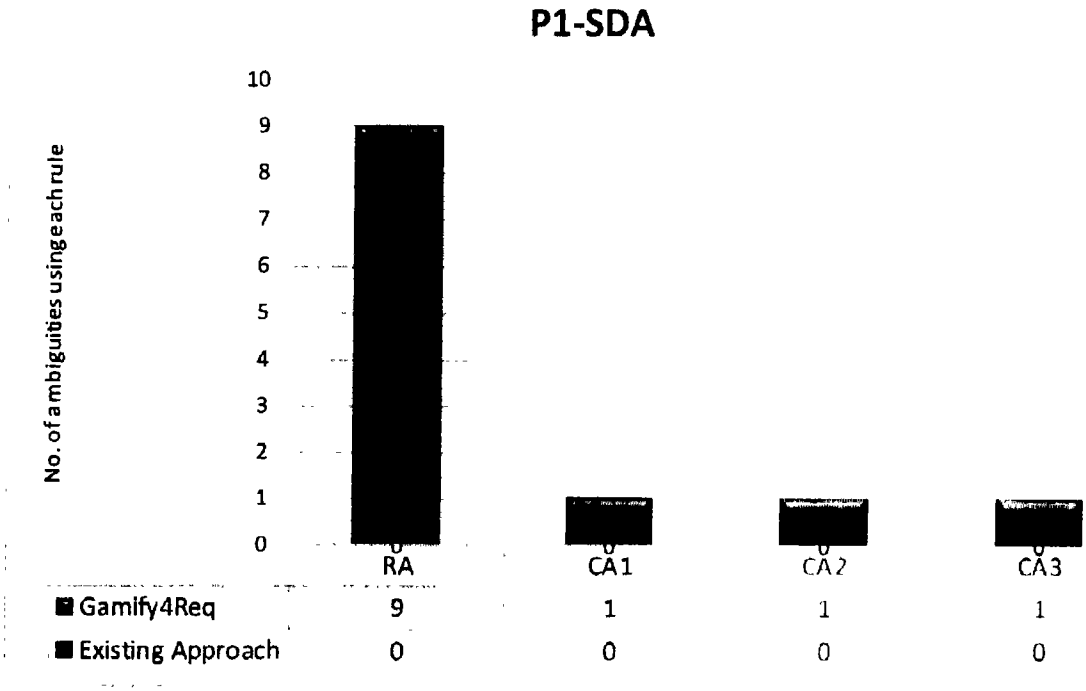



*Figure 41.6 Comparative analysis of both approaches in case I*

This comparative analysis presents a clear view of ambiguities identified in both approaches during elicitation.

## 6.1.2 Results from Case study II

As mentioned in the previous chapter, P2 is selected as our case study II. For comparative analysis of the results, existing requirements elicitation and ambiguity reduction method is selected, which is later compared to the results of Gamify4Req. In the next section, the case is presented using existing approach which is manual elicitation of requirements, ambiguity identification, and reduction.

### 6.1.2.1 P2-GOTCHA-Existing Approach

Interviews are used for requirements elicitation of P2-GOTCHA. During interviews, requirements are recorded manually which are later specified in formal template by a

technical writer. If any ambiguity is found during inspection, the client is inquired about it via telephonic call to resolve the ambiguity.

### 6.1.2.2 P2-GOTCHA-Gamify4Req

Gamify4Req is used for elicitation and reduction of ambiguity in requirements by the same RE team. In P2-GOTACHA-Gamify4Req, a total number of 35 requirements are elicited. The tool identifies ambiguity after the user provides any requirement.

Table 24.6 shows requirements elicited from both approaches.

*Table 24.6 Requirements ambiguity in case II*

| No. | Requirements | P1-Existing | | P1-Gamify4Req | |
|---|---|---|---|---|---|
| | | *Ambiguity Identification* | *Ambiguity Rules* | *Ambiguity Identification* | *Ambiguity Rules* |
| 1. | In case moderate to severe the system shall ***recommend*** exercises and therapy to patient | X | X | ✓ | CA2 |
| 2. | The system shall ***perform*** authentication and verification of therapist | X | X | ✓ | CA2 |
| 3. | The system shall prompt the therapist to tell ***his*** available working hours | ✓ | X | ✓ | RA |
| 4. | The system shall show the therapist his schedule every time ***he*** logs in | ✓ | X | ✓ | RA |
| 5. | The system shall show the therapist ***his*** schedule of pending therapy sessions that therapist need to conduct | X | X | ✓ | RA |
| 6. | After the session is completed, the system shall prompt the therapist ***and*** patient to give feedback ***and*** write complain | X | X | ✓ | CA1 |
| 7. | The system shall require the admin to log into the system to perform ***his*** duties | ✓ | X | ✓ | RA |
| 8. | The system shall give access to admin for ***registered*** therapist and patient record | ✓ | X | ✓ | CA2 |
| 9. | The system shall let the admin to view feedback ***and*** complains from patients ***and*** therapists | ✓ | X | ✓ | CA1 |
| 10. | The system shall let the admin to block ***registered*** patient and therapist account after reviewing the complains | ✓ | X | ✓ | CA2 |

Table 24.6 shows 10 ambiguous requirements. Using the existing approach, 35 requirements are elicited from the users. These requirements are recorded manually. Among the

ambiguous requirements, 6 ambiguous requirements are discarded due to their irrelevance to the scope of the tool whereas the approach does not categorize ambiguity into any type. All requirements are checked in a similar way. All requirements are elicited and checked for ambiguity using Gamify4Req. Once the process is completed, the user generates the requirements document to review the requirements. After requirements are elicited, activity is formally completed by the PM. Further details are given in Appendix C (Section 1).

### 6.1.2.3 Comparative Analysis of the Results from Case study II

A comparison of results using both approaches i.e., existing and Gamify4Req of case II, based on the factors are given in table 25.6 below.

*Table 25.6 Comparative analysis of results from case II*

| No. | Factors | P2- Existing Approach | P2- Gamify4Req |
|---|---|---|---|
| 1. | Total number of requirements | 35 | 35 |
| 2. | Total number of ambiguities | 6 | 10 |
| 3. | Time taken to identify each ambiguity | 10 minutes | 2 seconds |
| 4. | Time taken to resolve/reduce an ambiguity | 7 minutes | 30 seconds |

The comparison in table 25.6 is based on the factors that have been identified earlier to compare the results of both approaches. It is evident that Gamify4Req produced better results than the existing approach used in P2-GOTCHA. The number of ambiguities identified are more in number than the ones identified manually. However, using Gamify4Req, the ambiguities are identified in short time, but also reduced in visibly less time.

As mentioned earlier, using the existing approach the team was unable to categorize ambiguity during elicitation, but with Gamify4Req, the system categorized ten ambiguities during elicitation. Table 26.6 presents categorization of ambiguity.

*Table 26.6 Ambiguity categorization in case II*

| Users | P1-Existing Approach | P1-Gamify4Req | | | |
|---|---|---|---|---|---|
| | | Categorization | | | |
| | Categorization | RA | CA1 | CA2 | CA3 |
| ReqEngr. | X | 4 | 2 | 3 | - |
| User/Customer/DE | X | - | - | 1 | - |

Detailed breakdown of ambiguity categorization by users is given in table 26.6. The results of Gamify4Req show a significant difference in reducing ambiguity in requirements. Similarly, figure 42.6 presents a graphical presentation of both approaches and number of identified ambiguities.

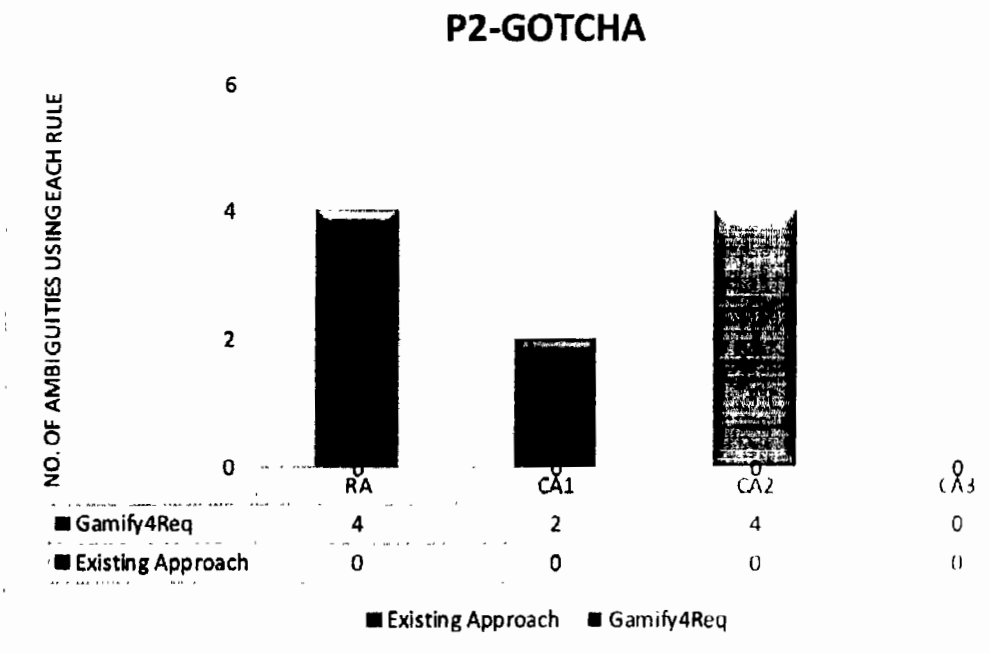



*Figure 42.6 Comparative analysis of both approaches in cose II*

This comparative analysis presents a clear view of ambiguities identified and reduced in both approaches during elicitation.

*Table 27.6 Time taken to identify and reduce ambiguity in Case I and II*

| No. | Factors | P1-SDA | | P2-GOTCHA | |
|---|---|---|---|---|---|
| | | *Existing Approach* | *Gamify4Req* | *Existing Approach* | *Gamify4Req* |
| 1. | Time taken to identify each ambiguity | 20 minutes | 2 seconds | 10 minutes | 2 seconds |
| 2. | Time taken to resolve/reduce an ambiguity | 5 minutes | 35 seconds | 7 minutes | 30 seconds |

Table 27.6 shows time taken to identify and resolve/reduce ambiguity in both cases and approaches. A significant difference can be seen between both approaches. Gamify4Req takes 2 seconds to identify ambiguity, and almost 30 seconds to reduce ambiguity.

Another aspect of the system is to engage users in the system by using game elements. For user involvement, we incorporated five game elements for User/Customer/DE and ReqEngr., and three game elements for the PM. Table 28.6 shows the game elements of each user role.

*Table 28.6 Game elements in Gamify4Req*

| ReqEngr. and User/Customer/DE | PM |
|---|---|
| Avatar | Avatar |
| Levels | Levels |
| Points | Progress |
| Badges | - |
| Leaderboard | - |

These game elements are incorporated to enhance user involvement in the system. To get feedback on user involvement in the tool, a survey is conducted from the users and RE teams who have used the system.

## 6.2 Survey Results

The purpose of conducting this survey is to know about user involvement in the tool. Another purpose is to get their feedback on ambiguity identification and reduction. The results of the survey are analysed using MS Excel; however, a statistical test is applied to determine the significance difference in results. Survey results of the users involved in the projects P1-SDA and P2-GOTCHA are presented in detail. Survey questionnaire is attached in Appendix E.

Before asking for user involvement, ambiguity identification and reduction, the demographic information is collected from the respondents, which includes:

- User roles
- Experience
- Software development methodology

As shown in figure 43.6, users are asked about the role they have played while using the system. The roles are already defined i.e., PM, ReqEngr., and User/Customer/DE.

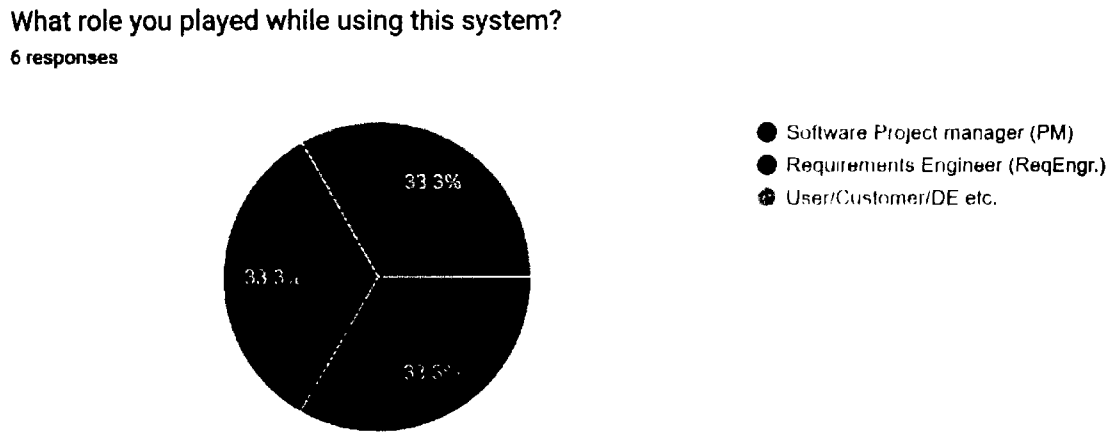



*Figure 43.6 User roles*

There are 2 PM, 2 ReqEngr, and 2 User/Customers/DE as respondents. Among 6 respondents, 2 User/Customer/DE and 1 ReqEngr. has less than equal to 1 year of experience, whereas 1 PM has 8 years of experience while the other 1 has 2-4 years of experience. Similar, 1 ReqEngr. has almost 5-7 years of experience as shown below in figure 44.6 and 45.6.

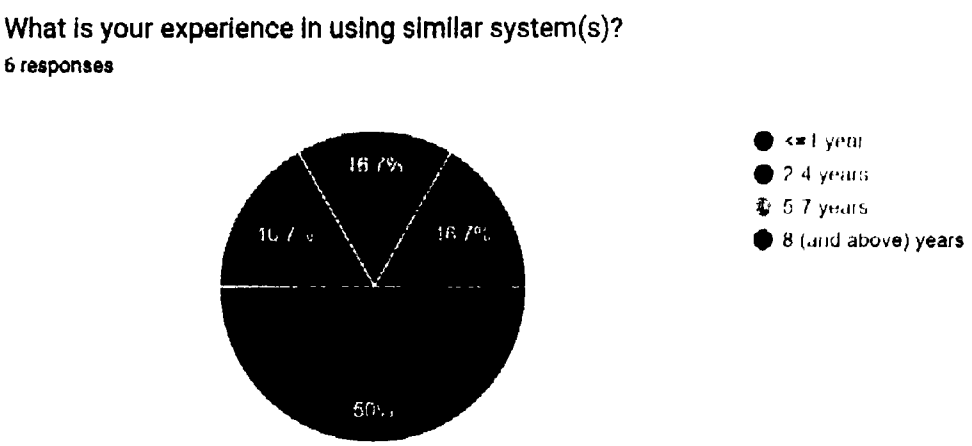



*Figure 44.6 User experience*

Figure 446.6 shows that almost 50% of the respondents have less than 1 year of experience of using a similar system. Whereas figure 37.6 shows that there are 3 respondents who have less than 1 year of experience. Only 1 PM has almost 8 years of experiencing a similar system.

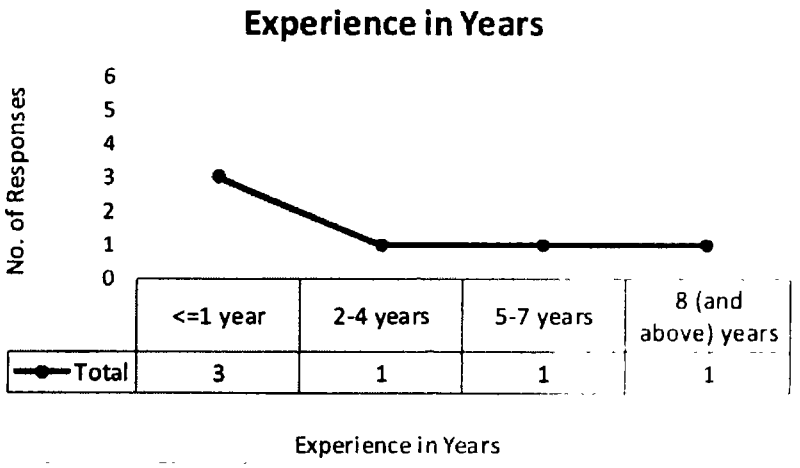



*Figure 45.6 User experience graph year-wise*

It is also asked to mention software development methodology the software house has been using for projects: agile or the traditional one. In both projects agile software development methodology has been used. After demographic information, the response is collected on user involvement and engagement. As shown in figure 46.6, the purpose and objective of using the tool is asked from the respondents.

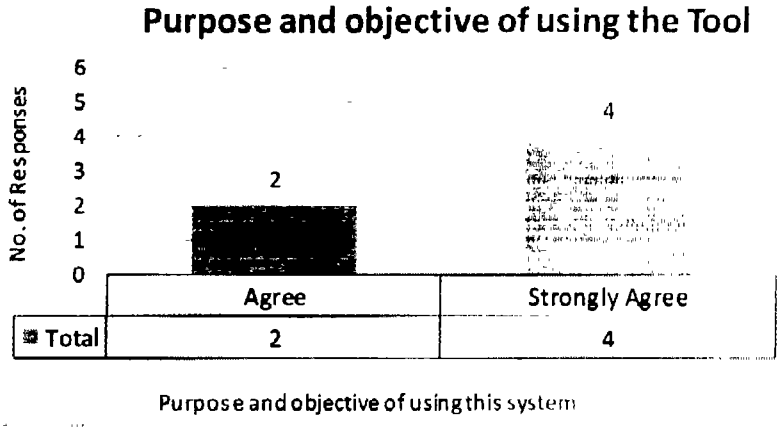



*Figure 46.6 Purpose and objective*

Among 6 respondents 4 strongly agreed that the purpose and objective of using this tool is clear to them, whereas 2 respondents agreed that the purpose and objective of using this tool is clear.

*Avatar*

Figure 47.6 further shows the response on use of avatars, where 3 respondents agree that avatar is easy to select, and 3 strongly agree to it.

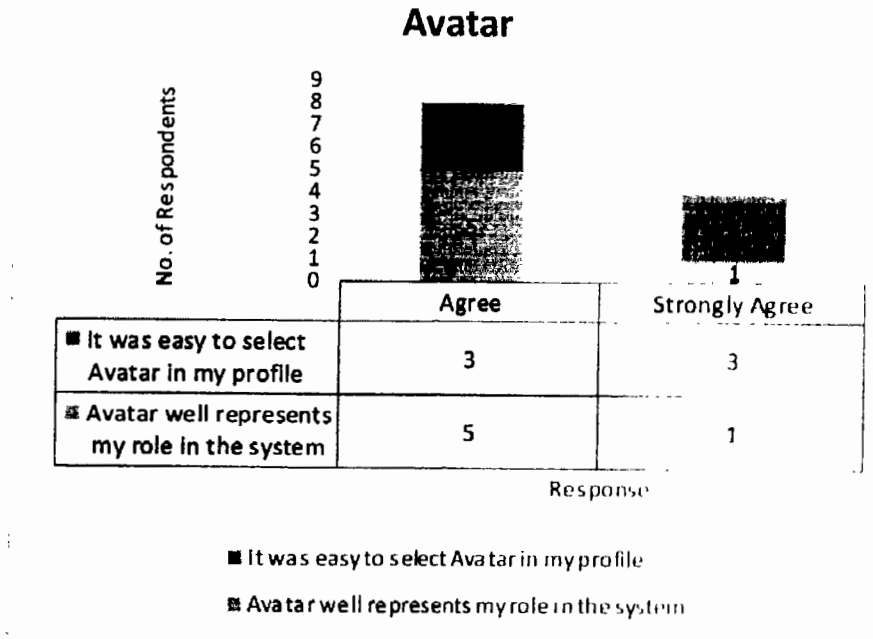



*Figure 47.6 Avatar*

Figure 47.6 presents that the 5 respondents agree that avatar well represents their role in the system, whereas 1 respondent strongly agrees to it.

*Levels*

After avatar, response on another game element 'levels' is selected, to which 2 users agree that levels kept the curiosity of what is coming next in the system, whereas 3 are neutral, and 1 strongly agrees to this, as shown in figure 48.6.

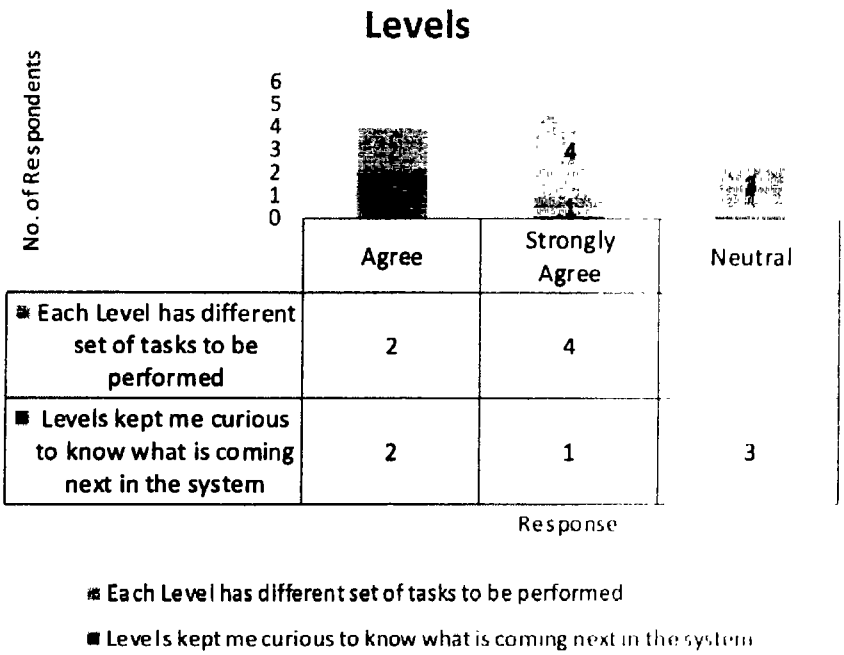



*Figure 48.6 Level*

Figure 48.6 presents the response on different set of tasks in a level. 2 respondents agree that each level has a unique set of tasks, whereas 4 respondents strongly agree to this.

### *Badges*

Figure 49.6 shows the number of responses on feeling of achievement related to the 'badges.

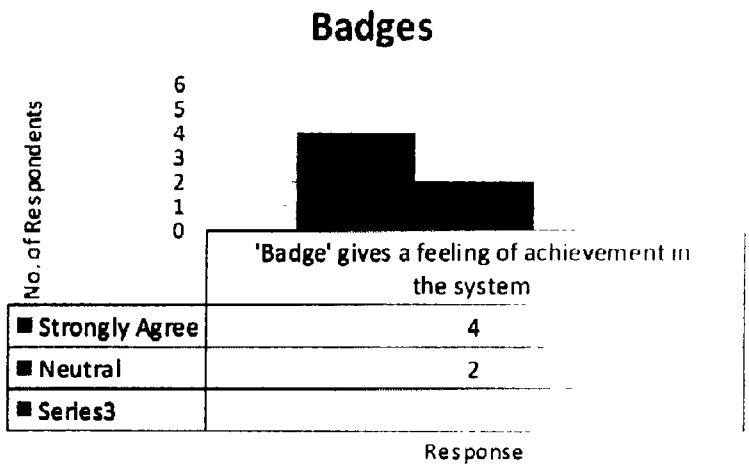



*Figure 49.6 Badge*

Figure 49.6 shows 4 respondents strongly agreeing that badges give a feeling of achievement while using the system.

***Points***

In figure 50.6 respondents provided their responses on 'points' upon adding each requirement. 2 respondents agree that points are rewarding, 2 stay neutral, whereas 2 strongly agree on this.

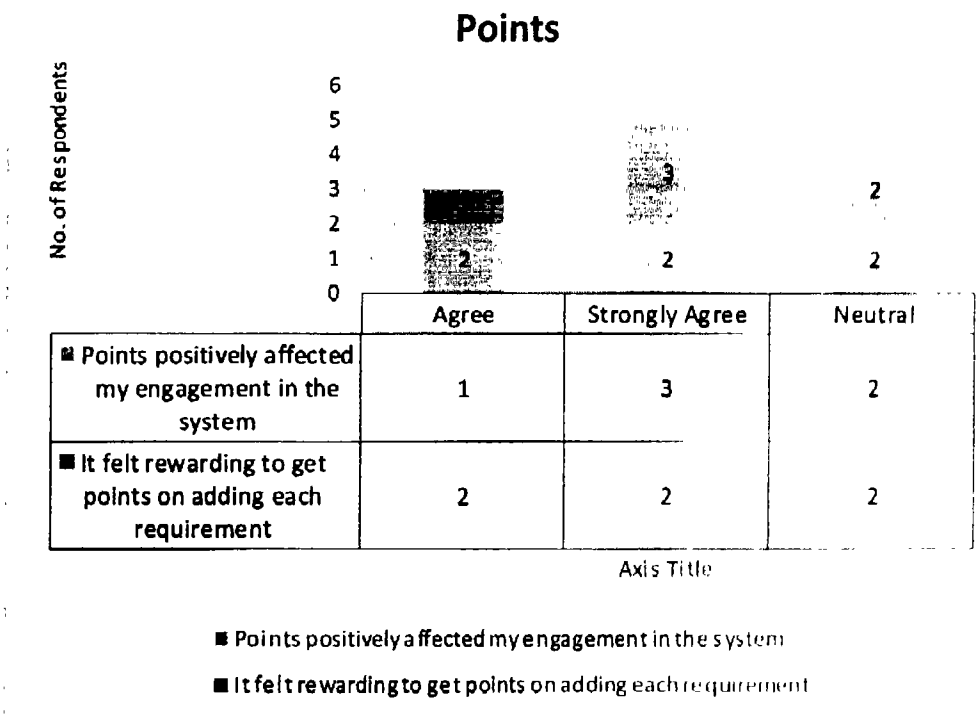



*Figure 50.6 Points*

In figure 50.6, 3 respondents strongly agree that points helped to engage them in the system, 1 respondent agree to this, and 2 stay neutral on it.

***Leaderboard***

Figure 51.6 shows the response on user engagement in using leaderboard. According to the response, 4 respondents agree that leaderboard is significant to check the points of other users, whereas 2 users stay neutral about it.

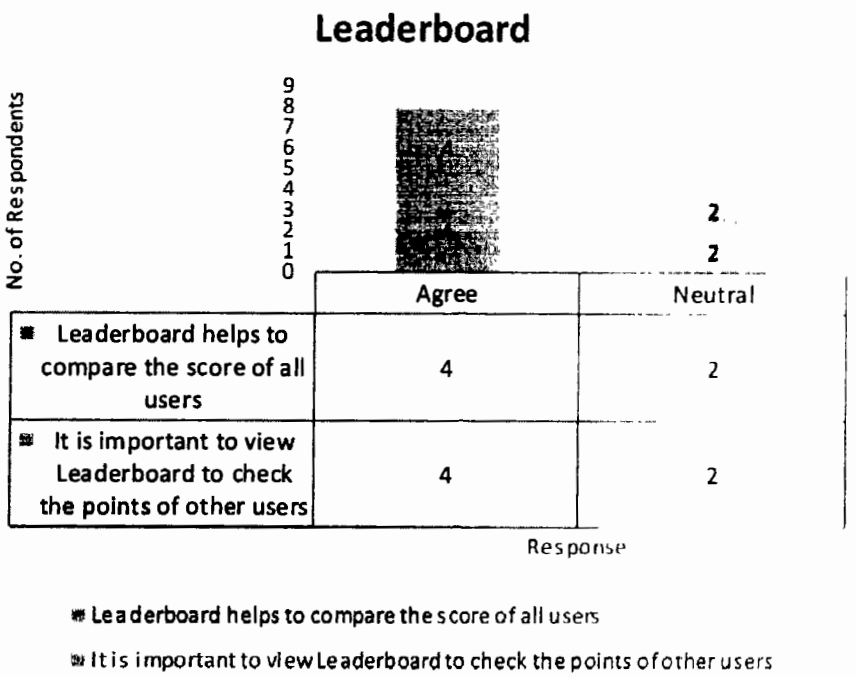



*Figure 51.6 Leaderboard*

According to response, leaderboard also help to compare the score of all users involved in the activity. 4 respondents agree to it and 2 stay neutral on it.

***Progress***

Project progress is an important game element especially for RE teams. In figure 52.6, 4 respondents strongly agree that project managers can view progress of the project when required, whereas 1 respondent agree to it and 1 stay neutral on it.

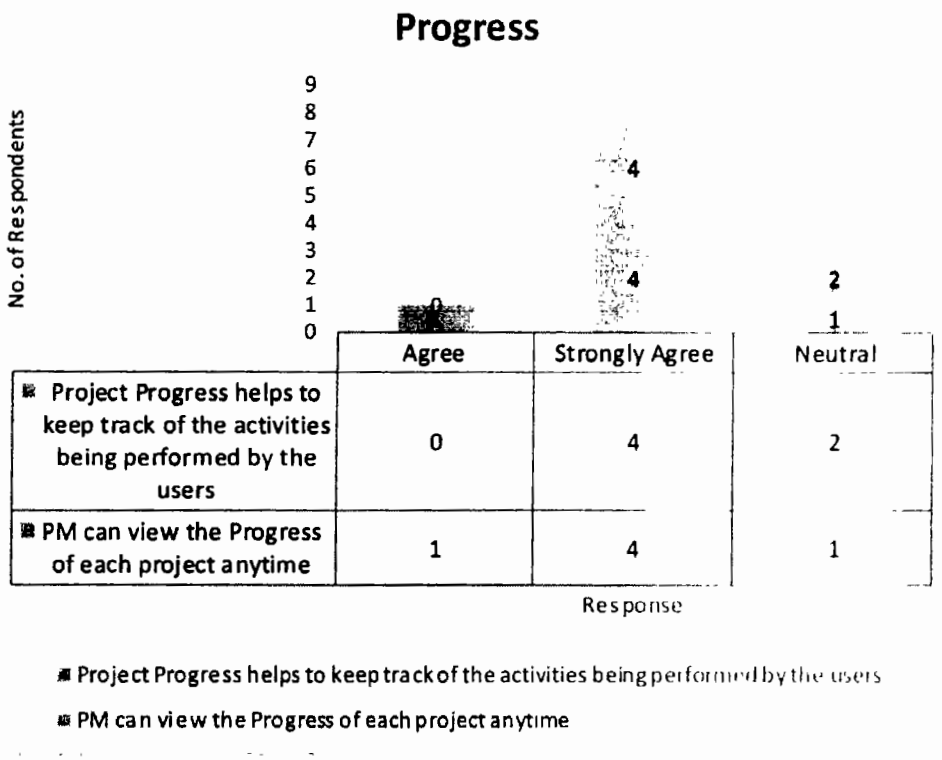



*Figure 52.6 Progress*

Project progress helps to keep track of activities being performed by the users involved in the project, which is strongly agreed by 4 respondents, while 2 respondents stay neutral on it, as shown in figure 52.6.

### *Game Elements*

Game elements motivate the user to use the system. In figure 53.6, 1 respondent agrees to this, and 5 respondents strongly agree. The responses show that inclusion of game elements is a source of fun while using the system. 4 respondents strongly agree while 2 respondents agree.

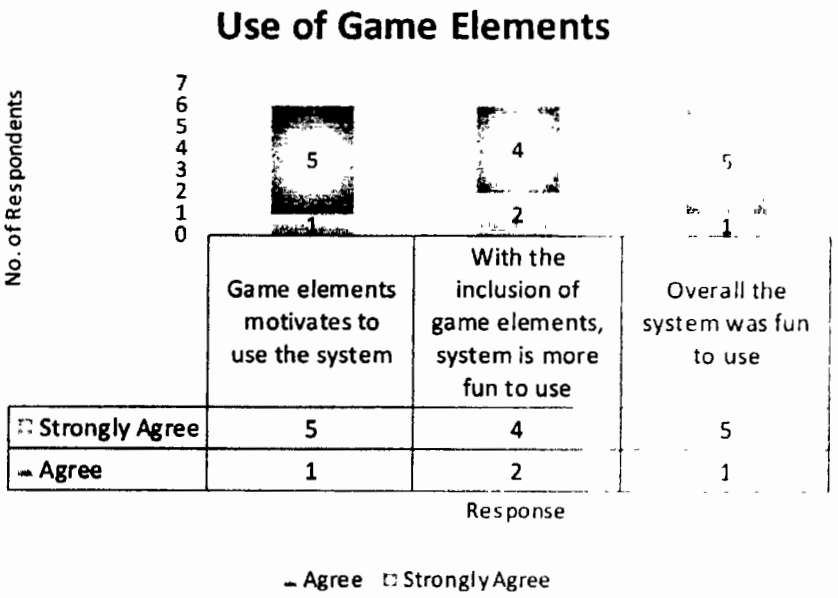



*Figure 53.6 Response on game elements*

Lastly, in user engagement, 5 respondents strongly agree that overall, the system was fun to use, whereas 1 respondent agrees to this, according to figure 53.6.

### *Most Useful Game Elements*

In figure 54.6, respondents selected the most exciting and useful game elements.

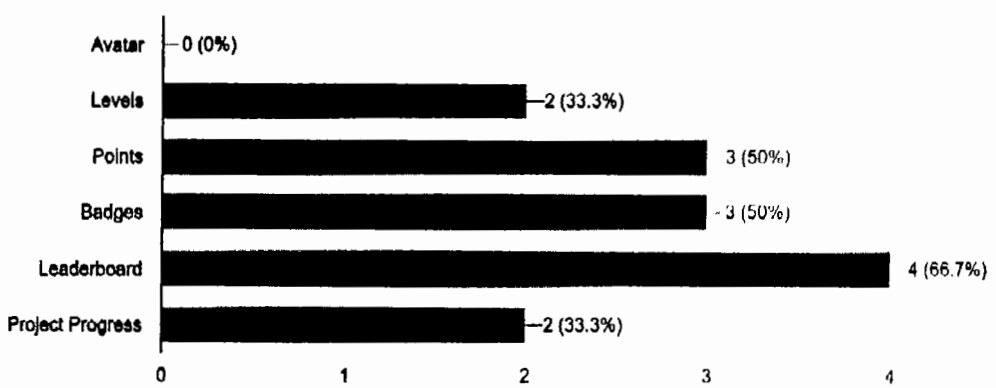



*Figure 54.6 Response on most liked game elements*

It is shown that leaderboard is more significant according to 4 respondents, followed by points and badges responded by 3 and 3 respondents respectively. Levels and project progress are selected as liked element by 2, 2 respondents respectively. Respondents also

selected the least liked game elements. According to their response, avatar is the least liked item responded by 6 respondents, followed by levels which are responded by the 2 respondents. 1 respondent also selected points as the least liked game element.

The second section of the survey is about identification and reduction of ambiguity using Gamify4Req.

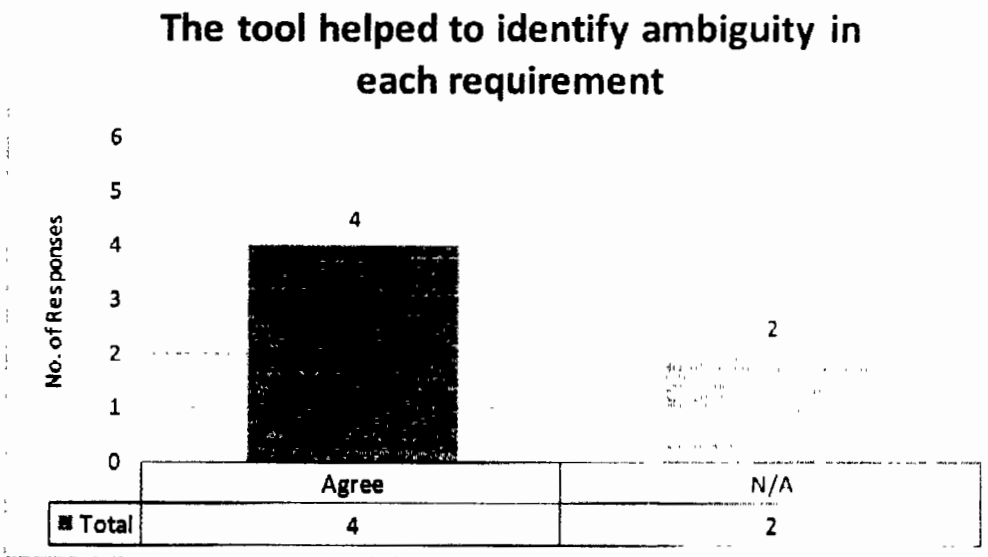



*Figure 55.6 Response on ambiguity identification*

4 respondents agree that the system helps to identify requirements ambiguity, whereas 2 stay neutral on this, as presented is figure 55.6.

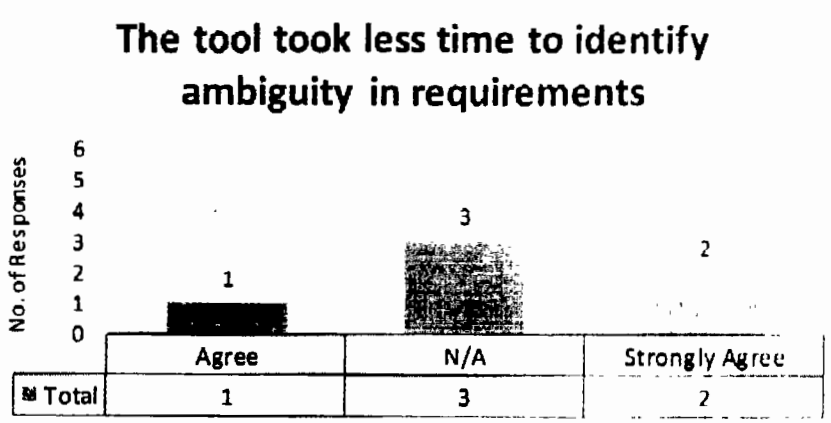



*Figure 56.6 Response on time taken on ambiguity identification*

In figure 56.6, 2 respondents strongly agree that the tool identifies ambiguity in less time, 1 respondent agrees, whereas 3 respondents remain neutral on this.

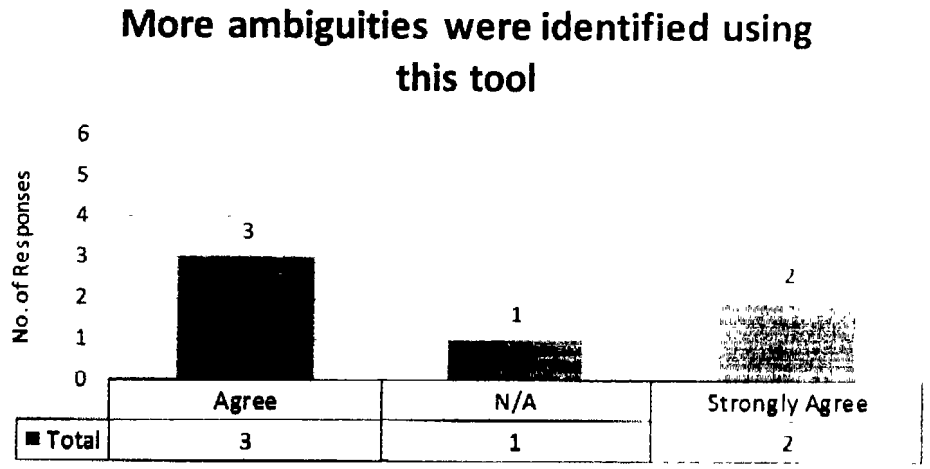



*Figure 57.6 Response on number of identified ambiguities*

In figure 57.6, 2 respondents strongly agree that the system identifies more ambiguities in the given time, 3 respondents agree, whereas 1 respondent remains neutral on this.

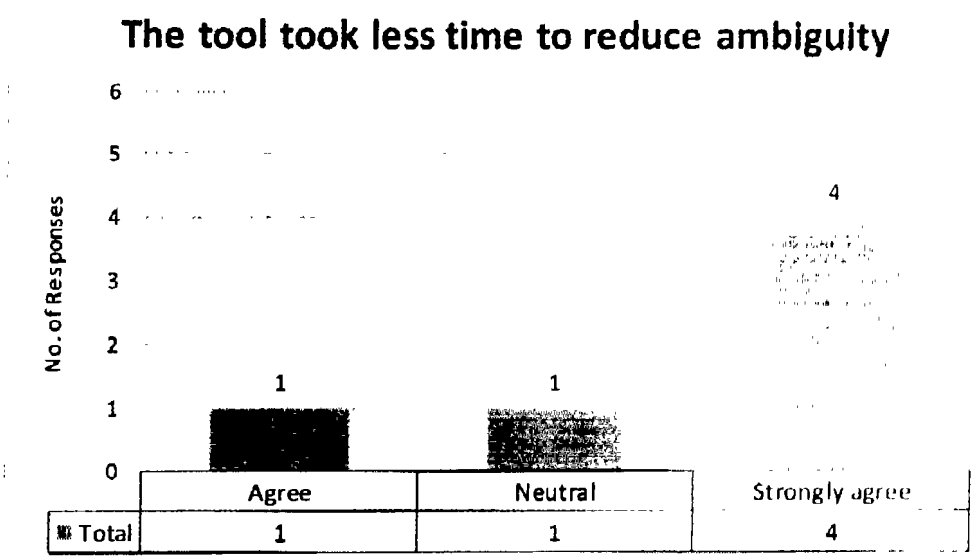



*Figure 58.6 Response on time taken to reduce ambiguity*

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, as shown in figure 58.6.

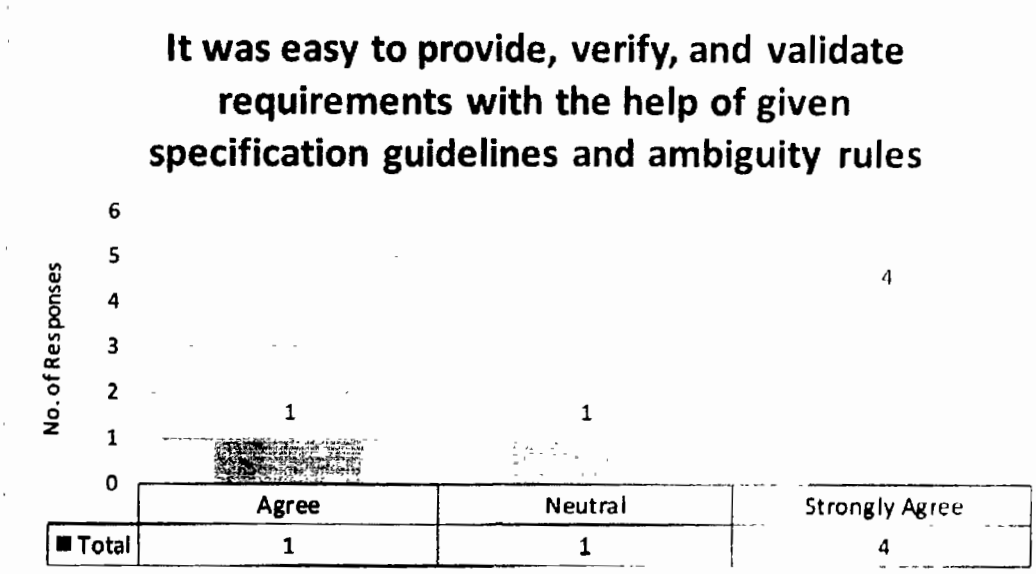



*Figure 59.6 Response on ease in requirements elicitation activity*

In figure 59.6, respondents believe that with the help of given specification guidelines and ambiguity rules in the system, it is easy to provide, verify, and validate requirements. 4 respondents strongly agree with this, whereas 1 respondent agrees and 1 stays neutral on this.

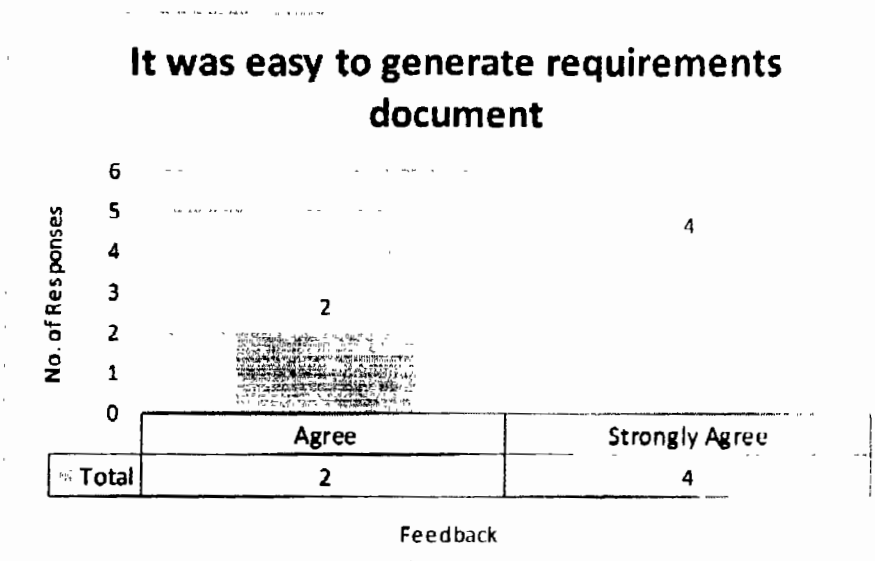



*Figure 60.6 Response on ease in generating requirements document*

In figure 60.6, 4 respondents strongly agree that requirements document is easy to generate from the system, and 2 respondents agrees to this.

The respondents also shared the challenges while using the tool. They reported that the tool is used under a controlled environment where training of the tool requires more time. The users are also convinced that gamified tool is less cost effective as compared to manual way of requirements elicitation and ambiguity reduction.

### 6.2.1 Statistical Analysis

The responses received from the survey are analysed and presented in this section. A tool named 'Jamovi' [174] is used to perform all the tests. Jamovi is a free alternative to SPSS tool for performing statistical analysis. Significance difference is checked to know if nominal data has any significance over the results.

#### 6.2.1.1 Significance Difference

Significance difference is checked to know if there is any difference between ordinal and nominal data. In our study, we need to see if the roles of PM, ReqEngr. and User/Customer/DE, and experience has led to any significant difference. Firstly, groups of nominal data have been made, and then tests are selected. We used the Mann Whitney U test because the data is ordinal. Mann Whitney U Test compares two sample means that belongs to the same population [175]. It is also used to check if two sample means are equal or not. The data is checked for test applicability, and groups are made. The following grouping is used for execution of Mann Whitney test in Jamovi.

Roles: User/Customer/DE., ReqEngr., PM (Measured in 3 groups)

- Group 1 User/Customer/DE. – ReqEngr.
- Group 2 User/Customer/DE/. – PM
- Group 3 ReqEngr. – PM

Experience: 4 levels (Measured in 6 groups)

- Group 1 <=1 – 2-4 years
- Group 2 <=1 – 5-7 years
- Group 3 <=1 – 8 (and above) years

- Group 4 2-4 years – 5-7 years
- Group 5 2-4 years – 8 (and above) years
- Group 6 5-7 years – 8 (and above) years

Software development methodology: Agile (Scrum, XP, DSDM, FDD, Kanban), Traditional (Waterfall, Spiral, Prototype, RAD, V-Model).

- We don't see any group in methodology because both organizations are following Agile Scrum.

For this test we are ignoring the 'other' option. The grouping is composed in a way that each group has exactly two values to find the difference. Furthermore, if the p value is between 0.01 and 0.05 then there is a significant difference. The detailed Mann Whitney U test is given in Appendix F. The main findings of the tests are given below:

**Group I** *User/Customer/DE. and ReqEngr.*

i. To evaluate the difference between frequency of the 'use of game elements' and 'ambiguity reduction', Mann Whitney test is conducted. The test revealed no significant difference in the use of game elements and ambiguity reduction between the roles of ReqEngr. and User/Customer/DE, whereas User/Customer/DE experience of using the similar system is <=1 year and ReqEngr. has <=1 – 5-7 years of experience.

**Group II** *User/Customer/DE. and PM*

ii. There is no significant difference between the roles of User/Customer/DE and PM having experience of using the similar system <=1 – 5-7 years, and <=1 – 8 (and above) years.

**Group III** *ReqEngr. and PM*

iii. To evaluate the difference between frequency of the 'use of game elements' and 'ambiguity reduction', Mann Whitney test is conducted on the roles of ReqEngr. and PM having experience <=1 – 5-7 years and 2-4 years – 8 (and above) years. The test

revealed no significant difference in the use of game elements and ambiguity reduction between both roles.

As shown in figure 61.6 below.

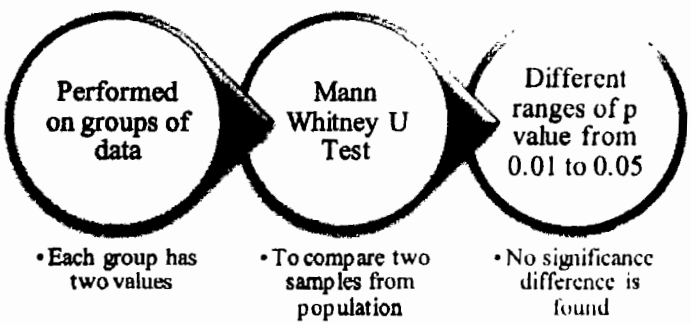



*Figure 61.6 Summary of Test*

Figure 61.6 summarizes the outcome of Mann Whitney U Test. After running the test on groups taken from the population, no significant difference is found.

# CHAPTER 7

## *FINDINGS AND DISCUSSION*

# Chapter 7 Findings and Discussion

## 7.1 Findings

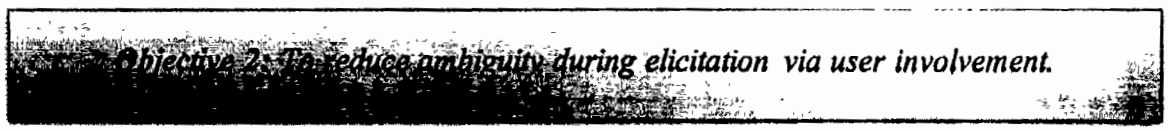
The aim of this study is to reduce requirements ambiguity with the help of gamified tool by involving users during elicitation. This tool is helpful for RE teams to timely elicit requirements, identify and reduce ambiguity in requirements. An interesting part of the tool is to involve users in the tool with the help of useful game elements such as avatar, levels, points, badges, leaderboard, and progress. The purpose is to get the user involved in the tool in fun ways, so the user gives requirements by involving and engaging in the activity. The motivation of this work is to reduce ambiguity in requirements by i). involving users to the system, and 2). using ambiguity rules. Furthermore, ambiguity in requirements is identified and reduced at the time of elicitation. Whereas in previous studies, ambiguity is identified in requirements inspection phase with SRS. However, this is a proactive approach that identifies and reduces ambiguity during elicitation, by involving and engaging users in the activity.

Our research study has three research objectives which helped us to achieve the goal of this study. Each objective is listed below along with the contributions of the research.


First objective of this study is to identify the game elements for requirements elicitation. For this purpose, in-depth literature survey is conducted to know the existing studies on gamification and game elements in RE. After reviewing related studies, useful game elements are selected for different phases of RE including requirements elicitation, analysis, specification, and validation. Almost 17 game elements are listed down from literature.

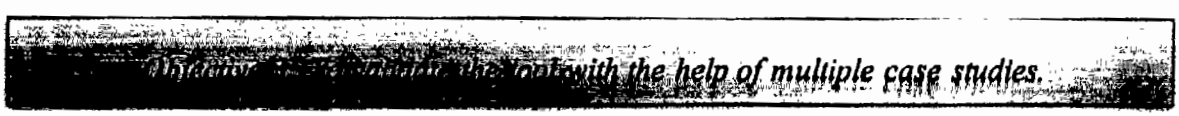
SLR is also conducted to systematically identify game elements other than the PBL, used in requirements elicitation. We identified 16 game elements from SLR that are useful during elicitation.

Objective 1 is also linked with objective 2 of the study.




Along with selecting game elements, ambiguity rules, and requirements specifications guidelines are also covered from literature. Semantic ambiguity rules are also adopted that further address referential and coordination sub-types of ambiguity. After thorough analysis of requirements ambiguity rules and game elements, the tool is designed and developed. The design of gamified tool includes game rules, ambiguity rules, user roles and responsibilities, user interfaces, use-case model, and other related models. The design is validated by the industry and academia experts. The design of tool in a technical track of IEEE 5th International Conference on Computing and Informatics, Cairo Egypt [152] was presented where reviewers provided the feedback on the tool.

After the completion of design, the tool was developed using NLP technique i.e., POS tagging, in PHP and Java Virtual Machine (JVM). Apache web server is used at the back-end. The tool is tested and revised according to the feedback received.




The validation of tool is an important step of the study. Firstly, the steps of case study were designed, variables were defined, instrument was selected, and other protocols were observed. Secondly, software houses were listed down and contacted for implementation of the tool. The participants were given a training session on how to use the tool. Using the gamified tool, requirements from the users were elicited and then analysed for further action. For getting user feedback on user involvement and engagement in the tool, we conducted a feedback survey. In the previous approach, user involvement was not considered as an important factor during requirements elicitation. Whereas it is an important part of our research study. The users fully participated in the survey.

The tool supports RE teams to elicit requirements from the users, identifies ambiguity, performs requirements verification and validation, and generates requirements document in less time with more involved users.

## 7.2 Discussion

Our study comprises of three research questions which are closely linked to the objectives of this study. In this section, with the support of findings, these questions are discussed and answered for concluding our work.

### 7.2.1 Answering Research Question 1

We investigate the answer to research question 1 *'What are useful game elements for elicitation of requirements?'*. In the first phase, literature review is conducted to find which game elements are useful for requirements elicitation via gamified tool. In literature review, 15 studies are reviewed, and 18 game elements are listed down. These game elements are used in gamified systems covering RE phases such as requirements elicitation, requirements analysis, requirements specification, and requirements validation. It is observed that:

- Game elements are mostly used in requirements elicitation than any other activity of RE
- Game elements in requirements elicitation are points (68%), leaderboard (50%) and badges (37%)
- Other than elicitation, game elements are also used in requirements analysis with points (31%), leaderboard (18%), badges (18%) and roles (18%)
- The use of game elements in requirements specification is very low and widely used game element is point (12%)
- The useful game element in requirements validation is points (18%)

Among the 18 game elements, points, leaderboard, and badges are the useful game elements that are widely used in gamified tools and systems, as shown in the figure 62.7.

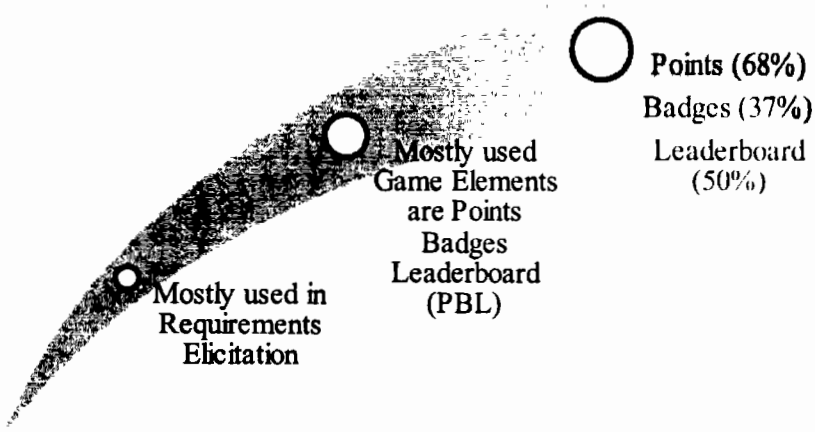



*Figure 62.7 Results of RQ1*

Moreover, according to user feedback survey after using the gamified tool, the useful game elements are points (50%), leaderboard (66%), and badges (50%).

### 7.2.2 Answering Research Question 2

To investigate the answer to research question 2 *'What are the game elements empirically validated for requirements elicitation?'* A detailed SLR, covering background knowledge and related studies, is conducted. SLR is planned, conducted, and reported for the identified game elements that are used during elicitation. After investigation, we identified 16 game elements including trophy, levels, feedback, virtual goods, time limit, medals, etc. According to SLR outcomes, it is observed that:

- There are useful game elements other than the PBL that are used in RE.
- The most useful game elements among 16 game elements are level (61.1%), feedback (44.4%), and progress (38.8%).
- The game elements other than the PBL are useful in elicitation.
- Level, feedback, and progress are widely used, yet other game elements are not frequently occurred during investigation.

As shown in the figure 63.7.

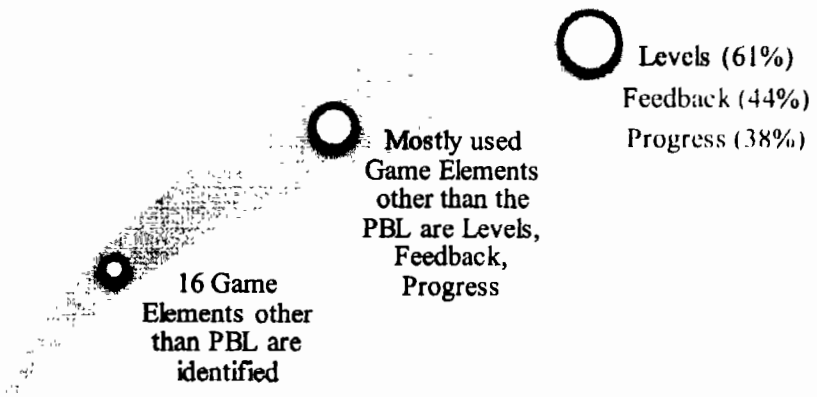



*Figure 63.7 Results of RQ2*

The results of RQ 2 in figure 63.7 shows that in our gamified tool, the game elements play a vital role in collecting user requirements. The active involvement of users during requirements elicitation directly impacts the quality of collected requirements. Therefore, employing game elements provide a valuable advantage in increasing user involvement within the system.

### 7.2.3 Answering Research Question 3

We investigate the answer to research question 3 *'How effective is the developed gamification tool in reducing ambiguity during elicitation?'* To answer this question, a technique to identify ambiguity in requirements and ambiguity rules is selected. The gamified tool incorporates game rules and game elements to involve users. For this purpose, a gamified tool is designed. After the design, gamified elicitation tool is developed. During elicitation, User/Customer/DE and RE teams provides user requirements which are checked for ambiguity at the time of elicitation. On provision of unambiguous requirement, gamified tool awards users with points and badges, and maintain leaderboard of all the users. The tool prompts the user to provide unambiguous requirement after highlighting the identified ambiguity in requirement. Gamification engages the user to use the tool for what's coming

next and who has achieved what (points and badges). In this way unambiguous requirements are elicited via gamification during requirements elicitation.

The validation of the developed gamified tool is also performed. The validation is performed on industrial case studies. Software houses are selected for implementation of the tool. Out of four software houses, two were willing to participate in the study. Unfortunately, the response rate was not good. The participants were given training on using the gamified tool. The results from both case studies were recorded and analysed. Another aspect of the study is to compare the results of existing approaches with the results generated after using our gamified tool. The comparison is based on the following factors: number of requirements, number of ambiguities, time to detect ambiguity, and time to reduce ambiguity.

The comparison is made while conducting the case study. The results showed that gamified tool helps to reduce ambiguity in requirements by involving users in elicitation. The effectiveness of Gamify4Req is evaluated. In P1, 41 requirements are collected and in P2, 35 requirements are collected. Gamify4Req identified the ambiguities in each project, as shown in the figure 64.7.

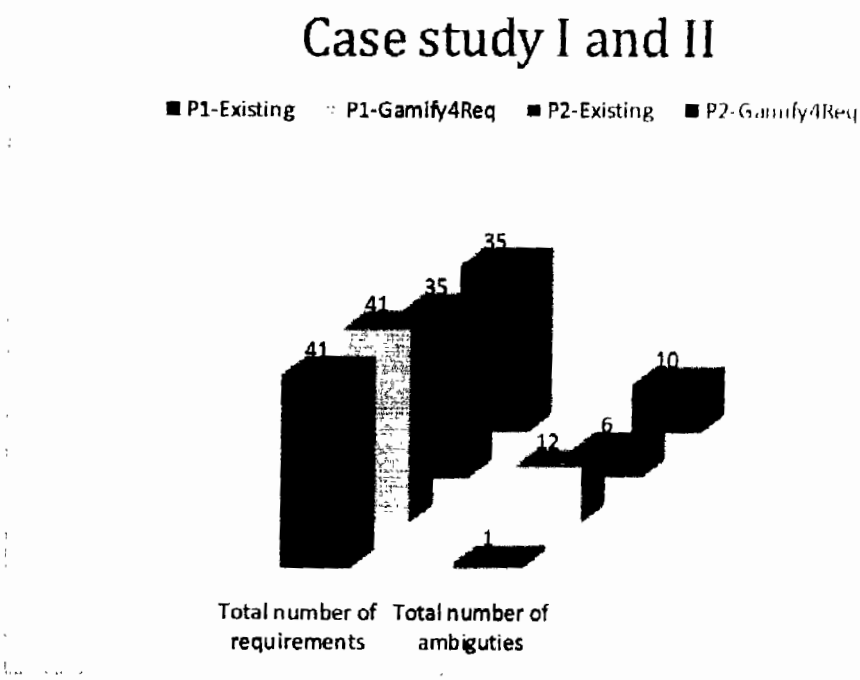



*Figure 64.7 Case I and Case II*

After analysing the results of both case studies, a significant difference is observed between existing and gamified approaches. Gamify4Req supports elicitation process and helps the team and users to provide requirements in a time saving manner. As shown in figure 64.7, P1 and P2 acquired 41 and 35 requirements respectively. Total number of ambiguities vary in both cases and approaches as well. The user feedback survey is also conducted with the participants. The user involvement survey has two major sections: user involvement, and ambiguity identification and reduction.

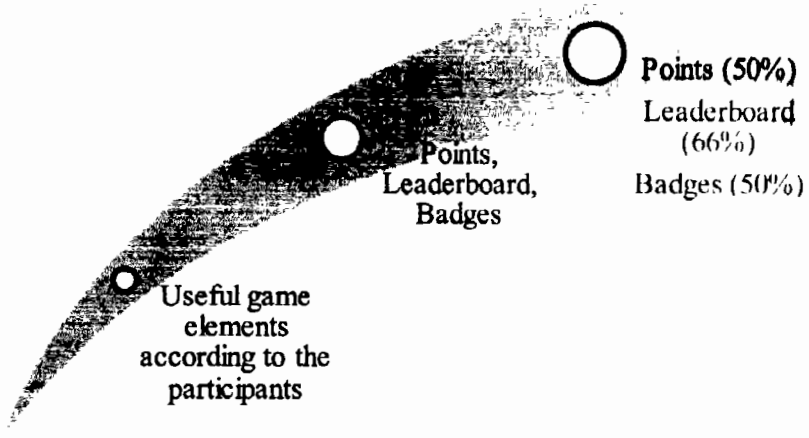



*Figure 65.7 User feedback on user involvement in Gamify4Req*

The results showed that using gamification to reduce ambiguity during elicitation works very well, as shown in the figure 65.7. The process took less time in elicitation and efficiently identified and reduced ambiguity in requirements.

Furthermore, summary of mapping answers to RQs and Ros are presented in the figure 66.7 below.

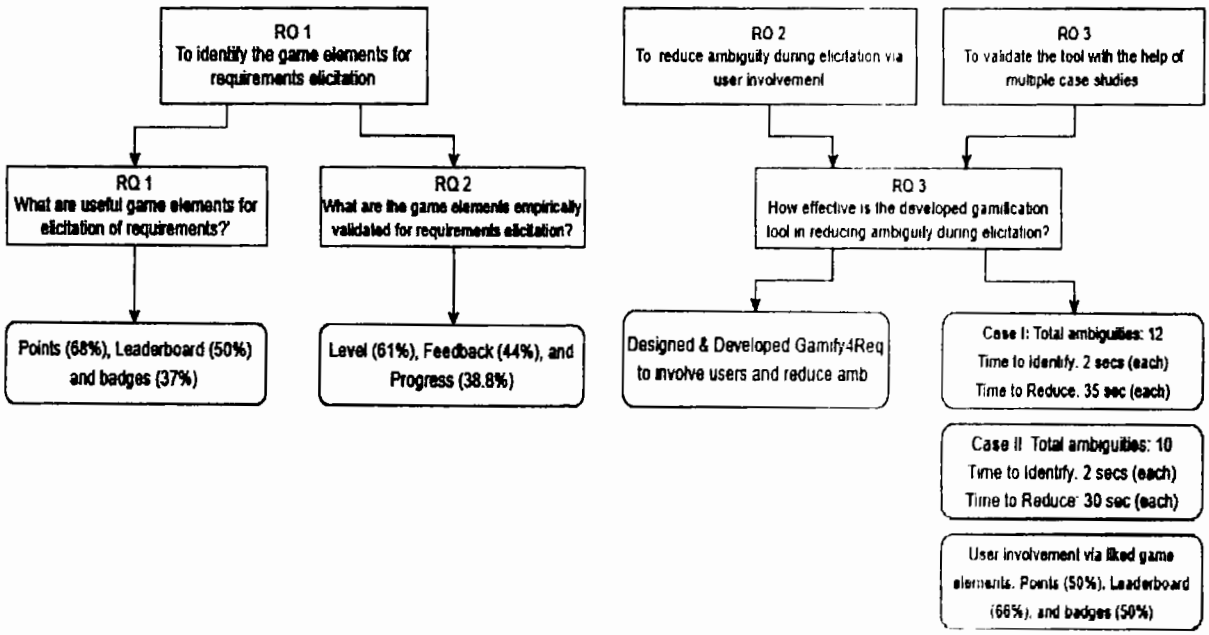



*Figure 66.7 Mapping of answers on ROs and RQs*

In figure 66.7, we have mapped research findings to their respective RQ and then RO. All the RQs are answered, hence the Ros of this study are successfully met.

### 7.2.4 Threats to Validity

While designing and conducting case study, there were possible threats to its validity such as case selection bias, documentation and information quality, and sampling issues. These threats are discussed as follows.

- *Internal Validity*

  In internal validity of the study, the purpose of this research was focused for minimizing the threat of selection bias of IT company, projects, and participants. Different aspects like type of the company and project, information availability, and overall feasibility of conducting this research were considered.

- *Construct Validity*

  In construct validity, the validity of tool design and case study design were performed with the help of industry experts and researchers. The requirements document provided by the PM was inadequate, thus it was difficult to establish a clear cause-

and-effect relationship between different elements of the study. Moreover, it was first completed and then verified by the PM.

- *External Validity*

  In external validity of the study, different factors like population, sample size, and tasks given to them were considered. The small sample size limits the generalizability of the findings. Similarly, convincing PMs and project teams to spend time on using the tool and recording the results was also very difficult. Due to which it was hard to draw broader conclusions from limited cases and participants.

We address these threats by careful consideration of each aspect during the design and execution of the case study. The transparency and rigor to enhance the overall validity of the research was also ensured.

# CHAPTER 8

## *CONCLUSION AND FUTURE WORK*

# Chapter 8 Conclusion and Future Work

## 8.1 Conclusion

In this study we aim to reduce ambiguity in requirements during elicitation. NL requirements are ambiguous due to language complexity. User may encounter challenges in expressing their requirements due to unclear system function and objectives. They might not possess a clear understanding of the requirements. Additionally, the absence of user involvement in elicitation leads to misunderstanding and ambiguities in requirements specification. According to previous literature, researchers have primarily focused on identifying ambiguity, while less attention is given to reducing ambiguities in requirements. Furthermore, the existing approaches are reactive and do not involve users in the process of reducing ambiguity during elicitation. The aim of conducting this research is to reduce ambiguity in requirements during elicitation by involving users during the elicitation and incorporating ambiguity rules to identify ambiguity.

In this study we have designed and developed a gamified tool for reducing ambiguity in requirements during elicitation. Our aim is to support RE teams and users to elicit unambiguous requirements by encouraging user involvement. Existing approaches are reactive and do not involve users in reducing ambiguity in requirement. The web-based gamified tool incorporates interesting game elements like avatar, levels, points, badges, progress, and leaderboard to involve and engage user in requirements elicitation for reducing ambiguity in requirements. The results show that our tool not only identifies and reduces ambiguity in NL requirements, but it also involves user in requirements elicitation. Furthermore, Gamify4Req is an interactive tool which is based on conceptual foundations of gamification and NL requirements. The tool validation is performed to show the effectiveness of the tool, which is later compared with the existing method of addressing requirements ambiguity.

### 8.1.1 Implication of Research

The implications of this research are significant and far-reaching. By incorporating gamification into requirements elicitation, we have created an involving and interactive environment for the users and RE teams. This proactive approach has the potential to enhance user involvement in reducing ambiguity in requirements, ultimately leading to clearer and more concise requirements. Furthermore, the gamified tool uses the technique of gamification. Gamification makes the requirements elicitation activity more enjoyable and motivating for users. It uses game elements that motivates the users to provide their requirements.

The implications of this research extend to other processes of software development such as project management and user-centred design, where clear and unambiguous requirements are vital for successful software development.

### 8.1.2 Research Contribution

The idea behind this research is novel in a way that gamification is not previously used for identification and reduction of requirements ambiguity during elicitation. Following are the research contributions:

1. Reviewing and analysing existing literature on requirements ambiguity and gamification
2. Conducting literature review to get an insight on useful game elements used in requirements elicitation
    a. Identification of game useful game elements
3. Conducting SLR to systematically review the literature for empirically validated useful game elements
    a. Identification of useful game elements other than the PBL
4. Designing a gamified tool based on ambiguity rules and game elements
5. Developing a web-based gamified tool
    a. Incorporating POS tags for identification of ambiguity in given requirements

b. Using gamification to involve users for reducing ambiguity in given requirements

6. Performing validation of gamified tool while conducting multiple case studies. Though it was a difficult stage to convince software houses to implement and use the tool, and it took time to get the results, it is done in a comprehensive manner.

### 8.1.3 Limitations of Research

Working on this idea is a whole new learning experience. However, the study contains certain limitations.

1. Gamification is comparatively a young concept in RE. Therefore, there is not much literature present in using gamification for ambiguity identification and reduction in requirements.
2. Due to traditional approaches present for requirements elicitation in software houses, it was difficult to convince the PMs to invest their time on using the tool.
3. For design validation, lack of responses was the challenge. We reached out to several experts but only a few of them responded. The response time was quite slow, therefore only two case studies are conducted.
4. Currently, the system considers only those requirements that are written in natural language (English).
5. Currently, the system allows only one PM to manage the activity, projects, and users. However, Gamify4Req is a multiplayer system and more than one ReqEngr. and User/Customer/DE can be added.
6. The tool is used in a controlled environment with a fixed number of users including RE teams and users.
7. The tool may not be a cost-effective solution for small-sized software houses. Trade-offs could be possible depending upon the priorities of the higher management.

### 8.1.4 Research Ethics

Throughout the execution of this research, utmost attention has been given to ethical considerations. All participants involved in this study were provided with comprehensive information regarding the research objectives, and their consent was obtained prior to their participation. We have also made sure that no information on participants and their data provided is disclosed in the document. Furthermore, for case study and survey design generic guidelines are used. Statistical analysis of the data from Google Forms is saved in MS Excel file and the results are not manipulated. The research is conducted in compliance with ethical guidelines and regulations. We firmly believe that our research can serve as a model for conducting ethical research in the domain of software engineering and allied domains. Furthermore, we ensure that our research contributes well to knowledge.

A part of research work has been published in conferences and journals and is available online. All the submissions are made by keeping in view the research ethics.

## 8.2 Future Work

Requirements ambiguity is a serious challenge in NL requirements. Generally, ambiguity is handled during inspection phase when requirements are specified in SRS. Handling ambiguity in inspection is a time consuming and costly activity. In this study, we have designed and developed a web based gamified tool for reducing ambiguity in requirements during elicitation by involving users in elicitation. Gamify4Req is developed to help teams and users to not only elicit requirements but identify and reduce ambiguity in requirements by using game elements, game rules, and ambiguity rules to keep user interest in the activity. However, we encourage researchers and practitioners to work in the following future directions:

1. Due to scope and limited time-frame of the study, we conducted SLR to identify useful game elements other than the PBL. Whereas for more detailed coverage of game mechanics in RE, SLR can be conducted to cover more related studies.

2. Currently, only semantic ambiguity is addressed in a proactive approach. Other types of requirements ambiguity like lexical, pragmatic, language errors, and syntactic are equally challenging. Therefore, gamification with other types of requirements ambiguity can be incorporated.
3. Due to scope of the study, user involvement in requirements elicitation activity is focused, whereas other aspects such as user communication and collaboration with other users, while reducing ambiguity, is an interesting idea to explore.
4. The influence of using a gamified tool for reducing requirements ambiguity is an interesting area to explore. Therefore, the scope of the study can be expanded further on mid and large-scale projects.
5. The investigation of long-term effects of gamification on the quality and completeness of requirements would be an interesting idea.
6. As ambiguity types and preferences differ, a personalized gamification solution can be developed that caters preferences of RE teams for more effective ambiguity reduction.
7. Another direction is to integrate gamification with other elicitation techniques such as prototyping or scenario-based analysis, to provide a comprehensive approach for reducing requirements ambiguity.
8. Sustainability in requirements is an interesting and trending research area. A gamified RE platform mapped on software development goals of sustainability can be helpful for project teams.
9. Using gamification as prompt for ambiguity reduction in requirements engineering can be helpful for project teams.

# REFERENCES

# References


[1] K. Swathine and J. .KomalaLakshm, "Requirement Elicitation for Requirement in Software Engineering," *Int. J. Eng. Sci. Res. Technol.*, vol. 3, no. 12, 2014.

[2] H. Maria and Z. Ali, "Requirement Elicitation Techniques for Open Source Systems: A Review," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 1, 2018, doi: 10.14569/IJACSA.2018.090145.

[3] T. Iqbal, "Requirement Elicitation Technique: - A Review Paper," *Int. J. Comput. Math. Sci.*, vol. 3, no. 9, Nov. 2014.

[4] H. Dar, M. I. Lali, H. Ashraf, M. Ramzan, T. Amjad, and B. Shahzad, "A Systematic Study on Software Requirements Elicitation Techniques and its Challenges in Mobile Application Development," *IEEE Access*, vol. 6, pp. 63859–63867, 2018, doi: 10.1109/ACCESS.2018.2874981.

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

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

[7] W. H. B. Wan Hassim, "A Review on Effective Requirement Elicitation Techniques," *Int. J. Adv. Comput. Sci. Technol.*, vol. 6, no. 1, pp. 4–8, 2017.

[8] Omar Isam Al Mrayat, N. Norwawi, and N. Basir, "Requirements Elicitation Techniques: Comparative Study," *Int. J. Recent Dev. Eng. Technol.*, vol. 1, no. 3, Dec. 2013.

[9] T. Keller, "Contextual Requirements Elicitation An Overview," Department of Informatics, University of Zurich, 2011.

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

[11] S. Lane, P. O'Raghallaigh, and D. Sammon, "Requirements gathering: the journey," *J. Decis. Syst.*, vol. 25, no. sup1, pp. 302–312, Jun. 2016, doi: 10.1080/12460125.2016.1187390.

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

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

[14] M. Sadiq and S. K. Jain, "An Insight into Requirements Engineering Processes," in *Advances in Communication, Network, and Computing*, vol. 108, 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.

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

[16] N. Alflen, E. Prado, and A. Grotta, "A Model for Evaluating Requirements Elicitation Techniques in Software Development Projects:," in *Proceedings of the 22nd International Conference on Enterprise Information Systems*, Prague, Czech Republic: SCITEPRESS - Science and Technology Publications, 2020, pp. 242–249. doi: 10.5220/0009397502420249.

[17]R. Fatima, A. Yasin, L. Liu, J. Wang, W. Afzal, and A. Yasin, "Improving software requirements reasoning by novices: a story-based approach," *IET Softw.*, vol. 13, no. 6, pp. 564–574, Dec. 2019, doi: 10.1049/iet-sen.2018.5379.

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

[19]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.

[20]B. Davey and K. R. Parker, "Requirements Elicitation Problems: A Literature Analysis," *Issues Informing Sci. Inf. Technol.*, vol. 12, pp. 71–82, 2015.

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

[22]E. Kamsties, "Understanding Ambiguity in Requirements Engineering," in *Engineering and Managing Software Requirements*, A. Aurum and C. Wohlin, Eds., Berlin/Heidelberg: Springer-Verlag, 2005, pp. 245–266. doi: 10.1007/3-540-28244-0_11.

[23]V. Gervasi and D. Zowghi, "On the Role of Ambiguity in RE," in *Requirements Engineering: Foundation for Software Quality*, vol. 6182, R. Wieringa and A. Persson, Eds., in Lecture Notes in Computer Science, vol. 6182. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 248–254. doi: 10.1007/978-3-642-14192-8_22.

[24]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.

[25]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.

[26]M. Luisa, F. Mariangela, and N. I. Pierluigi, "Market research for requirements analysis using linguistic tools," *Requir. Eng.*, vol. 9, no. 1, pp. 40–56, Feb. 2004, doi: 10.1007/s00766-003-0179-8.

[27]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.

[28]A. V. Lamsweerde, "Formal Specification: a Roadmap," *Future Softw. Eng. Limerick Irel.*, 2000.

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

[30]D. M. Berry, "Ambiguity in Natural Language Requirements Documents," in *Innovations for Requirement Analysis. From Stakeholders' Needs to Formal Designs*, vol. 5320, 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.

[31]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.

[32]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: IEEE, 2015, pp. 812–817.

[33]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.

[34]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.

[35]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.

[36]F. de Bruijn and H. L. Dekkers, "Ambiguity in Natural Language Software Requirements: A Case Study," in *Requirements Engineering: Foundation for Software Quality*, vol. 6182, R. Wieringa and A. Persson, Eds., in Lecture Notes in Computer Science, vol. 6182. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 233–247. doi: 10.1007/978-3-642-14192-8_21.

[37]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.

[38]B. Gleich, O. Creighton, and L. Kof, "Ambiguity Detection: Towards a Tool Explaining Ambiguity Sources," in *Requirements Engineering: Foundation for Software Quality*, vol. 6182, R. Wieringa and A. Persson, Eds., in Lecture Notes in Computer Science, vol. 6182. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 218–232. doi: 10.1007/978-3-642-14192-8_20.

[39]C. C, Shan and B. Mutty, "Lexical and Syntactic Ambiguity In The Vaccine News Headlines Of The Star," *Int. J. Educ. Psychol. Couns.*, vol. 7, no. 47, pp. 221–239, 2022.

[40]M. C. MacDonald, "Probabilistic constraints and syntactic ambiguity resolution," *Lang. Cogn. Process.*, vol. 9, no. 2, pp. 157–201, May 1994, doi: 10.1080/01690969408402115.

[41]D. M. Berry, E. Kamsties, and M. M. Krieger, *From Contract Drafting to Software Specification: Linguistic Sources of Ambiguity*. University of California at Los Angeles USA, 2003.

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

[43]D. Wang and M. Sadrzadeh, "The Causal Structure of Semantic Ambiguities." arXiv, Jun. 14, 2022. Accessed: Nov. 26, 2022. [Online]. Available: http://arxiv.org/abs/2206.06807

[44]A. Fantechi, S. Gnesi, G. Ristori, M. Carenini, M. Vanocchi, and P. Moreschini, "Assisting requirement formalization by means of natural language translation," *Form. Methods Syst. Des.*, vol. 4, no. 3, pp. 243–263, May 1994, doi: 10.1007/BF01384048.

[45]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.

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

[47]C. Rupp and R. Goetz, "Linguistic methods of requirements-engineering (NLP," in *EuroSPI 2000 Industrial Proceedings*, Finland, 2000.

[48]D. M. Berry and E. Kamsties, "The syntactically dangerous all and plural in specifications," *IEEE Softw.*, vol. 22, no. 1, pp. 55–57, Jan. 2005, doi: 10.1109/MS.2005.22.

[49]F. Chantree, B. Nuseibeh, A. de Roeck, and A. Willis, "Identifying Nocuous Ambiguities in Natural Language Requirements," in *14th IEEE International Requirements Engineering Conference (RE'06)*, Minneapolis/St. Paul, MN: IEEE, Sep. 2006, pp. 59–68. doi: 10.1109/RE.2006.31.

[50]J. Bölte, A. Böhl, C. Dobel, and P. Zwitserlood, "Effects of referential ambiguity, time constraints and addressee orientation on the production of morphologically complex words," *Eur. J. Cogn. Psychol.*, vol. 21, no. 8, pp. 1166–1199, Dec. 2009, doi: 10.1080/09541440902719025.

[51]H. Yang, A. de Roeck, V. Gervasi, A. Willis, and B. Nuseibeh, "Extending Nocuous Ambiguity Analysis for Anaphora in Natural Language Requirements," in *2010 18th IEEE International Requirements Engineering Conference*, Sydney, Australia: IEEE, Sep. 2010, pp. 25–34. doi: 10.1109/RE.2010.14.

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

[53]I. S. Bajwa, M. Lee, and B. Bordbar, "Resolving Syntactic Ambiguities in Natural Language Specification of Constraints," in *Computational Linguistics and Intelligent Text Processing*, vol. 7181, 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.

[54]S. Deterding, R. Khaled, L. E. Nacke, and D. Dixon, "Gamification: Toward a Definition," in *CHI 2011*, 2011, pp. 1–4.

[55]D. Healey, "Gamification." Macmillan Education Springer Nature, 2019. [Online]. Available: https://www.macmillaneducation.es/wp-content/advantage/Gamification_White%20Paper_Mar%202019.pdf

[56]S. Deterding, M. Sicart, L. Nacke, K. O'Hara, and D. Dixon, "Gamification. using game-design elements in non-gaming contexts," in *Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems - CHI EA '11*, Vancouver, BC, Canada: ACM Press, 2011, p. 2425. doi: 10.1145/1979742.1979575.

[57]A. Darejeh and S. S. Salim, "Gamification Solutions to Enhance Software User Engagement—A Systematic Review," *Int. J. Hum.-Comput. Interact.*, vol. 32, no. 8, pp. 613–642, Aug. 2016, doi: 10.1080/10447318.2016.1183330.

[58]D. J. Dubois and G. Tamburrelli, "Understanding gamification mechanisms for software development," in *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering - ESEC/FSE 2013*, Saint Petersburg, Russia: ACM Press, 2013, p. 659. doi: 10.1145/2491411.2494589.

[59]K. Prasad and M. R. Mangipudi, "Gamification Framework with Reference Business Perspective," *Bus. Manag. Econ. Eng.*, vol. 20, no. 2, pp. 271–281, 2022.

[60]M. ZehiR, S. Bozkurt, S. Oktay, and S. B. Ünlü, "Gamification as A Promising Strategy For Sustainable Businesses," *J. Acad. Soc. Sci.*, vol. 129, no. 129, pp. 327–352, 2022, doi: 10.29228/ASOS.62204.

[61]B. Burke, "Gamification Trends and Strategies to Help Prepare for the Future." Gartner, 2012.

[62]S. Rajamarthandan, "Using Gamification to Build a Passionate and Quality-Driven Software Development Team." Cognizant, 2014.

[63]V. Platonova and S. Bērziša, "Gamification in Software Development Projects," *Inf. Technol. Manag. Sci.*, vol. 20, no. 1, Jan. 2017, doi: 10.1515/itms-2017-0010.

[64]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.

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

[66]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.

[67]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.

[68]B. Morschheuser, J. Hamari, K. Werder, and J. Abe, "How to Gamify? A Method For Designing Gamification," presented at the Hawaii International Conference on System Sciences, 2017. doi: 10.24251/HICSS.2017.155.

[69]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.

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

[71]P. Lombriser, F. Dalpiaz, G. Lucassen, and S. Brinkkemper, "Gamified Requirements Engineering: Model and Experimentation," in *Requirements Engineering: Foundation for Software Quality*, vol. 9619, 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.

[72]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.

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

[74]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 Comput. Sci.*, vol. 15, pp. 66–77, 2012, doi: 10.1016/j.procs.2012.10.059.

[75]K. Magylaitė, K. Kapočius, R. Butleris, and L. Čeponienė, "Towards High Usability in Gamified Systems: A Systematic Review of Key Concepts and Approaches," *Appl. Sci.*, vol. 12, no. 16, p. 8188, Aug. 2022, doi: 10.3390/app12168188.

[76]J. Hamari and J. Koivisto, "Why do people use gamification services?," *Int. J. Inf. Manag.*, vol. 35, no. 4, pp. 419–431, Aug. 2015, doi: 10.1016/j.ijinfomgt.2015.04.006.

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

[78]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')*, vol. 49, 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.

[79]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

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

[81]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.

[82]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.

[83]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*, vol. 5320, 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.

[84]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

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

[86]T. J. Vimalraj and B. Seema, "Identification of Ambiguity in Requirement Specification using Multilingual Word Sense," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 5, no. 6, Jun. 2016, doi: 10.17148/IJARCCE.2016.5681.

[87]G. Huzooree and V. D. Ramdoo, "A Systematic Study on Requirement Engineering Processes and Practices in Mauritius," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 5, no. 2, pp. 40–46, 2015.

[88]S. Osama and M. Aref, "Detecting and Resolving Ambiguity Approach In Requirement Specification: Implementation, Results And Evaluation," *Int. J. Intell. Comput. Inf. Sci.*, vol. 18, no. 1, pp. 27–36, 2018.

[89]A. Umber, I. S. Bajwa, and M. Asif Naeem, "NL-Based Automated Software Requirements Elicitation and Specification," in *Advances in Computing and Communications*, vol. 191, A. Abraham, J. Lloret Mauri, J. F. Buford, J. Suzuki, and S. M. Thampi, Eds., in Communications in Computer and Information Science, vol. 191. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 30–39. doi: 10.1007/978-3-642-22714-1_4.

[90]A. Yadav, A. Patel, and M. Shah, "A comprehensive review on resolving ambiguities in natural language processing," *AI Open*, vol. 2, pp. 85–92, 2021, doi: 10.1016/j.aiopen.2021.05.001.

[91]M. Mohanan and P. Samuel, "Software Requirement Elicitation Using Natural Language Processing," in *Innovations in Bio-Inspired Computing and Applications*, vol. 424, V. Snášel, A. Abraham, P. Krömer, M. Pant, and A. K. Muda, Eds., in Advances in Intelligent Systems and Computing, vol. 424. , Cham: Springer International Publishing, 2016, pp. 197–208. doi: 10.1007/978-3-319-28031-8_17.

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

[93]Y. Saleh, "Semantic Ambiguity in English Language," *Al Bahith Al-Aalami*, vol. 13, no. 48, 2017.

[94]R. Garigliano, D. Perini, and L. Mich, "Which Semantics for Requirements Engineering: from Shallow to Deep," in *oint Proceedings of REFSQ-2018 Workshops*, Utrecht, The Netherlands: CEUR, 2018.

[95]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*, vol. 10753, 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.

[96]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.

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

[98]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.

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

[100] A. R. Amna, "Identifying Ambiguity Problems in User Stories : A Proposed Framework," in *Proceedings of the Doctoral Consortium Papers Presented at the 34th International Conference on Advanced Information Systems Engineering (CAiSE 2022)*, Belgium: CEUR Workshop Proceedings, 2022.

[101] 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.

[102] A. Fantechi, S. Gnesi, and L. Semin, "Rule-based NLP vs ChatGPT in Ambiguity Detection, a Preliminary Study," in . *Joint Proceedings of REFSQ-2023 Workshops, Doctoral Symposium, Posters & Tools Track, and Journal Early Feedback Track Co-located with REFSQ 2023*, Barcelona, Catalunya, Spain, 2023.

[103] 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.

[104] 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.

[105] 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.

[106] F. S. Baumer 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.

[107] 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.

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

[109] A. Moharil and A. Sharma, "Identification of intra-domain ambiguity using transformer-based machine learning," in *Proceedings of the 1st International Workshop on Natural Language-based Software Engineering*, Pittsburgh Pennsylvania: ACM, May 2022, pp. 51–58. doi: 10.1145/3528588.3528651.

[110] C. Pacheco, I. García, and M. Reyes, "Requirements elicitation techniques: a systematic literature review based on the maturity of the techniques," *IET Softw.*, vol. 12, no. 4, pp. 365–378, Aug. 2018, doi: 10.1049/iet-sen.2017.0144.

[111] D. Duarte, C. Farinha, M. M. da Silva, and A. R. da Silva, "Collaborative Requirements Elicitation with Visualization Techniques," in *2012 IEEE 21st International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises*, Toulouse, Cedex 04, France: IEEE, Jun. 2012, pp. 343–348. doi: 10.1109/WETICE.2012.14.

[112] R. Mollá, V. Santamarina-Campos, F. Abad, and G. Tipantuña, "Storyboarding as a Means of Requirements Elicitation and User Interface Design: An Application to the Drones' Industry," in *Drones and the Creative Industry*, V. Santamarina-Campos and M. Segarra-Oña, Eds., Cham: Springer International Publishing, 2018, pp. 83–97. doi: 10.1007/978-3-319-95261-1_6.

[113] K. Seaborn and D. I. Fels, "Gamification in theory and action: A survey," *Int. J. Hum.-Comput. Stud.*, vol. 74, pp. 14–31, Feb. 2015, doi: 10.1016/j.ijhcs.2014.09.006.

[114] D. Cambridge, "Gamification," *Cambridge Advanced Learner's Dictionary & Thesaurus*. Cambridge University Press. [Online]. Available: https://dictionary.cambridge.org/dictionary/english/gamification

[115] Dictionary, "Gamification," *Oxford Learner's Dictionaries*. Oxford Learner's Dictionaries. [Online]. Available: https://www.oxfordlearnersdictionaries.com/definition/english/gamification

[116] A. Syaifulloh Imron, T. Raharjo, B. Hardian, and T. Simanungkalit, "Gamification to Improve Scrum Adoption: A Case Study At Poultry Startup In Indonesia," *J. Theor. Appl. Inf. Technol.*, vol. 100, no. 20, pp. 5854–5864, 2022.

[117] G. F. Tondello, A. Mora, and L. E. Nacke, "Elements of Gameful Design Emerging from User Preferences," in *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, Amsterdam The Netherlands: ACM, Oct. 2017, pp. 129–142. doi: 10.1145/3116595.3116627.

[118] A. Mazarakis, "Gamification Reloaded: Current and Future Trends in Gamification Science," *-Com*, vol. 20, no. 3, pp. 279–294, Dec. 2021, doi: 10.1515/icom-2021-0025.

[119] J. Krath, L. Schürmann, and H. F. O. von Korflesch, "Revealing the theoretical basis of gamification: A systematic review and analysis of theory in research on gamification, serious games and game-based learning," *Comput. Hum. Behav.*, vol. 125, p. 106963, Dec. 2021, doi: 10.1016/j.chb.2021.106963.

[120] F. Faiella and M. Ricciardi, "Gamification and learning: a review of issues and research," *J. E-Learn. Knowl. Soc.*, vol. Vol 11, p. No 3 (2015): Gamification and Serious Game for Learning, Sep. 2015, doi: 10.20368/1971-8829/1072.

[121] M. Passalacqua, S. Sénécal, M. Frédette, L. E. Nacke, R. Pellerin, and P.-M. Léger, "A Motivational Perspective on the Personalization of Gamification," in *Proceedings of the Nineteenth Annual Pre-ICIS Workshop on HCI Research in MIS*, Virtual: ICIS, 2020, pp. 1–5.

[122] L. Rodrigues *et al.*, "Personalization Improves Gamification: Evidence from a Mixed-methods Study," *Proc. ACM Hum.-Comput. Interact.*, vol. 5, no. CHI PLAY, pp. 1–25, Oct. 2021, doi: 10.1145/3474714.

[123] G. Barata, S. Gama, J. Jorge, and D. Gonçalves, "Improving participation and learning with gamification," in *Proceedings of the First International Conference on Gameful Design, Research, and Applications*, Toronto Ontario Canada: ACM, Oct. 2013, pp. 10–17. doi: 10.1145/2583008.2583010.

[124] Polytechnique Montreal *et al.*, "Should Gamification be Personalized? A Self- deterministic Approach," *AIS Trans. Hum.-Comput. Interact.*, pp. 265-286, Sep. 2021, doi: 10.17705/1thci.00150.

[125] Pulasthi. D. Gunawardhana and S. Palaniappan, "Gamification," *J. Adv. Appl. Sci.*, vol. 3, no. 2, pp. 51–58, 2015.

[126] 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.

[127] S. Deterding, D. Dixon, R. Khaled, and L. Nacke, "From game design elements to gamefulness: defining 'gamification,'" in *Proceedings of the 15th International Academic MindTrek Conference on Envisioning Future Media Environments - MindTrek '11*, Tampere, Finland: ACM Press, 2011, p. 9. doi: 10.1145/2181037.2181040.

[128] S. Balci, J. M. Secaur, and B. J. Morris, "Comparing the effectiveness of badges and leaderboards on academic performance and motivation of students in fully versus partially gamified online physics classes," *Educ. Inf. Technol.*, vol. 27, no. 6, pp. 8669–8704, Jul. 2022, doi: 10.1007/s10639-022-10983-z.

[129] T. Barik, E. Murphy-Hill, and T. Zimmermann, "A perspective on blending programming environments and games: Beyond points, badges, and leaderboards," in *2016 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, Cambridge, United Kingdom: IEEE, Sep. 2016, pp. 134–142. doi: 10.1109/VLHCC.2016.7739676.

[130] 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.

[131] N. U. Shpigel, "Towards a Systematic Approach for Designing Gamification for RE." Information Systems Department, University of Haifa, 2018. [Online]. Available: https://ceur-ws.org/Vol-2075/DS-paper4.pdf

[132] 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.

[133] L. Piras, P. Giorgini, and J. Mylopoulos, "Acceptance Requirements and Their Gamification Solutions," in *2016 IEEE 24th International Requirements Engineering Conference (RE)*, Beijing, China: IEEE, Sep. 2016, pp. 365–370. doi: 10.1109/RE.2016.43.

[134] L. Piras, E. Paja, P. Giorgini, J. Mylopoulos, R. Cuel, and D. Ponte, "Gamification solutions for software acceptance: A comparative study of Requirements Engineering and Organizational Behavior techniques," in *2017 11th International Conference on Research Challenges in Information Science (RCIS)*, Brighton, United Kingdom: IEEE, May 2017, pp. 255–265. doi: 10.1109/RCIS.2017.7956544.

[135] L. Piras, E. Paja, P. Giorgini, and J. Mylopoulos, "Goal Models for Acceptance Requirements Analysis and Gamification Design," in *Conceptual Modeling*, vol. 10650, 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.

[136] L. Piras, D. Dellagiacoma, A. Perini, A. Susi, P. Giorgini, and J. Mylopoulos, "Design Thinking and Acceptance Requirements for Designing Gamified Software," in *2019 13th International Conference on Research Challenges in Information Science (RCIS)*, Brussels, Belgium: IEEE, May 2019, pp. 1–12. doi: 10.1109/RCIS.2019.8876973.

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

[138] 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.

[139] A. Yasin, R. Fatima, Z. JiangBin, J. Ali Khan, and A. Ali Khan, "Gamifying requirements: An empirical analysis of game-based technique for novices," *J. Softw. Evol. Process*, p. e2617, Sep. 2023, doi: 10.1002/smr.2617.

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

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

[142] 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.

[143] 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.

[144] 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.

[145] I. Alvertis, D. Papaspyros, S. Koussouris, S. Mouzakitis, and D. Askounis, "Using Crowdsourced and Anonymized Personas in the Requirements Elicitation and Software Development Phases of Software Engineering," in *2016 11th International Conference on Availability, Reliability and Security (ARES)*, Salzburg, Austria: IEEE, Aug. 2016, pp. 851–856. doi: 10.1109/ARES.2016.71.

[146] E. B. Nilsen, D. E. Bowler, and J. D. C. Linnell, "Exploratory and confirmatory research in the open science era," *J. Appl. Ecol.*, vol. 57, no. 4, pp. 842–847, Apr. 2020, doi: 10.1111/1365-2664.13571.

[147] O. P. Atieno, "An Analysis of The Strengths and Limitation of Qualitative And Quantitative Research Paradigms," *Probl. Educ. 21st Century*, vol. 13, pp. 13–18.

[148] S. Easterbrook, J. Singer, M.-A. Storey, and D. Damian, "Selecting Empirical Methods for Software Engineering Research," in *Guide to Advanced Empirical Software Engineering*, 2008, pp. 285–311.

[149] R. G. Butler, "Exploratory vs Confirmatory Research." [Online]. Available: https://www.butlerscientifics.com/single-post/2014/10/08/exploratory-vs-confirmatory-research

[150] B. Kitchenham, "Guidelines for performing systematic literature reviews in software engineering," Durham, UK, Technical report EBSE-2007-01, 2007.

[151] 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.

[152] 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.

[153] R. G. Butler, "Exploratory vs Confirmatory Research," Butler Scientifics, Technical TS-2, 2014.

[154] 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

[155] M. Kasunic, *Designing an Effective Survey*. in CMUISEI-2005-HB-004. CarneggieMellon Software Engineering Institute, 2005.

[156] M. Garcia-Iruela and R. Hijon-Neira, "What Perception Do Students Have About the Gamification Elements?," *IEEE Access*, vol. 8, pp. 134386 134392, 2020, doi: 10.1109/ACCESS.2020.3011222.

[157] A. a Carolina Tome Klock, M. Soares Pimenta, and I. Gasparini, "A Systematic Mapping of the Customization of Game Elements in Gamified Systems," in *SBC Proceedings of SBGames 2018*, Brazil, 2018.

[158] R. Gafni, D. Biran Achituv, S. Eidelman, and T. Chatsky, "The effects of gamification elements in e-learning platforms," *Online J. Appl. Knowl. Manag.*, vol. 6, no. 2, pp. 37–53, 2018.

[159] L.-H. Shih and Y.-C. Jheng, "Selecting Persuasive Strategies and Game Design Elements for Encouraging Energy Saving Behavior," *Sustainability*, vol. 9, no. 7, p. 1281, Jul. 2017, doi: 10.3390/su9071281.

[160] S. Brull and S. Finlayson, "Importance of Gamification in Increasing Learning," *J. Contin. Educ. Nurs.*, vol. 47, no. 8, pp. 372–375, Aug. 2016, doi: 10.3928/00220124-20160715-09.

[161] M. H. Abdul Rahman, I. Ismail Yusuf Panessai, N. A. Z. Mohd Noor, and N. S. Mat Salleh, "Gamification Elements and Their Impacts On Teaching And Learning - A Review," *Int. J. Multimed. Its Appl.*, vol. 10, no. 06, pp. 37–46, Dec. 2018, doi: 10.5121/ijma.2018.10604.

[162] E. Nasirzadeh and M. Fathian, "Investigating the effect of gamification elements on bank customers to personalize gamified systems," *Int. J. Hum.-Comput. Stud.*, vol. 143, p. 102469, Nov. 2020, doi: 10.1016/j.ijhcs.2020.102469.

[163] M. Sailer, J. U. Hense, S. K. Mayr, and H. Mandl, "How gamification motivates: An experimental study of the effects of specific game design elements on psychological need satisfaction," *Comput. Hum. Behav.*, vol. 69, pp. 371–380, Apr. 2017, doi: 10.1016/j.chb.2016.12.033.

[164] A. M. Toda *et al.*, "Analysing gamification elements in educational environments using an existing Gamification taxonomy," *Smart Learn. Environ.*, vol. 6, no. 1, p. 16, Dec. 2019, doi: 10.1186/s40561-019-0106-1.

[165] S.-C. Wee and W.-W. Choong, "Gamification: Predicting the effectiveness of variety game design elements to intrinsically motivate users' energy conservation behaviour," *J. Environ. Manage.*, vol. 233, pp. 97–106, Mar. 2019, doi: 10.1016/j.jenvman.2018.11.127.

[166] K. Robson, K. Plangger, J. H. Kietzmann, I. McCarthy, and L. Pitt, "Is it all a game? Understanding the principles of gamification," *Bus. Horiz.*, vol. 58, no. 4, pp. 411–420, Jul. 2015, doi: 10.1016/j.bushor.2015.03.006.

[167] W. Ghaban and R. Hendley, "How Different Personalities Benefit From Gamification," *Interact. Comput.*, vol. 31, no. 2, pp. 138–153, Mar. 2019, doi: 10.1093/iwc/iwz009.

[168] R. Ferrer Conill, "Points, badges, and news. A study of the introduction of gamification into journalism practice," *Comun. Rev. Recer. D'Anàlisi*, no. 33, pp. 45–63, 2016, doi: 10.2436/20.3008.01.148.

[169] A. P. Bertholdo, C. Melo, A. Rozestraten, M. Gerosa, and H. O'Brien, "User engagement in an open collaboration community after the insertion of a game design element: An online field experiment," in *AMCIS 2018 Proceedings*, 2018.

[170] S. M. Schöbel, A. Janson, and M. Söllner, "Capturing the complexity of gamification elements: a holistic approach for analysing existing and deriving novel gamification designs," *Eur. J. Inf. Syst.*, vol. 29, no. 6, pp. 641–668, Nov. 2020, doi: 10.1080/0960085X.2020.1796531.

[171] C. Mese and O. Ozgur Dursun, "Effectiveness of Gamification Elements In Blended Learning Environments," *Turk. Online J. Distance Educ.-TOJDE*, vol. 20, no. 3, pp. 119–142, 2019.

[172] I. ANSI, "IEEE Guide to Software Requirements Specifications." The Institute of Electical and Electronics Engineers, 1984.

[173] 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.

[174] J. Love, "Jamovi." in jamovi Desktop. Jamovi Stats, 2017. [Online]. Available: https://www.jamovi.org/

[175] A. Buthmann, "Making Sense of Mann-Whitney Test for Median Comparison," I Six Sigma. [Online]. Available: 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.

# APPENDIX

# Appendices

## Appendix A Design Validation Checklist

### Response 1 *(from industry experts)*

**General Information**

Project Name: Gamify4Req (Reducing Requirements Ambiguity via Gamification)
Reviewer: Mr. Shahzad Badar
Designation: CTO
Organization: SafeKids.ai
Date: 10/11/2021
Artifacts reviewed: General design, models and UI, game elements and rules

| Artifacts | Assessment (Y/N) | Recommendations (if any) |
|---|---|---|
| **General Design** | | |
| Does the design support development step? | Y | |
| Does the design support project goals? | Y | |
| Is the design feasible with the technology used? | Y | |
| Are the notations, methodology etc. used in the design appropriate? | Y | |
| Does the design have conceptual integrity? | Y | |
| Is the design complex? | N | |
| Is the design flexible? | Y | UI should be responsive |
| Does the design address all issues from the requirements? | Y | |
| **Models and User Interfaces (Storyboards)** | | |
| When appropriate, are there multiple, consistent, models and/or views that represent the design? | Y | |
| Where there are multiple models of the software are those models consistent with each other? | Y | |
| Have all interfacing systems been identified? | Y | |
| Is the flow of data in all interfaces consistent? | Y | |
| Is the design consistent with related artifacts (i.e., other modules, designs, etc.)? | Y | |
| Does user interfaces provide services and inputs and outputs? | Y | |
| Do the colors adhere to industry or organizational style? | Y | |
| Is the wording appropriate and consistent? | Y | |
| **Game Elements and Rules** | | |
| Does the system adhere to the game rules? | Y | |
| Does the system adhere to semantic ambiguity rules? | Y | |
| Are requirements specified according to the defined rules? | Y | |
| Does the game elements according to the user roles? | Y | |
| Does game elements specific to the task? | Y | |

Figure 1 Appendix A Response 1

- Give user option to update requirements once submitted
- Can we show progress of the project to the PM
- Add charts to the dashboard to show progress of the project
- User guide will help new users to understand the system
- Make UI responsive, sometimes buttons get overlapped

## Response 2

**General Information**

Project Name: Gamify4Req (Reducing Requirements Ambiguity via Gamification)
Reviewer: Dr. Ansar
Designation: Assistant Professor
Organization: Dept. of Software Engineering, University of Gujrat
Review Date: 1/7/2022
Artifacts Reviewed:

a. General Design
b. Models and User Interfaces (Storyboards)
c. Game Elements and Rules

Table Appendix A Response 2

| Artifacts | Assessment (Y/N) | Recommendations (if any) |
|---|---|---|
| **General Design** | | |
| Does the design support development step? | Y | |
| Does the design support project goals? | Y | |
| Is the design feasible with the technology used? | Y | |
| Are the notations, methodology etc. used in the design appropriate? | Y | |
| Does the design have conceptual integrity? | Y | |
| Is the design complex? | N | |
| Is the design flexible? | Y | |
| Does the design address all issues from the requirements? | Y | |
| **Models and User Interfaces (Storyboards)** | | |
| When appropriate, are there multiple, consistent, models and/or views that represent the design? | Y | |
| Where there are multiple models of the software are those models consistent with each other? | Y | |
| Have all interfacing systems been identified? | Y | |
| Is the flow of data in all interfaces consistent? | Y | |
| Is the design consistent with related artifacts (i.e., other modules, designs, etc.)? | Y | |
| Does user interfaces provide services and inputs and outputs? | Y | |
| Do the colors adhere to industry or organizational style? | Y | |
| Is the wording appropriate and consistent? | Y | |
| **Game Elements and Rules** | | |
| Does the system adhere to the game rules? | Y | |
| Does the system adhere to semantic ambiguity rules? | Y | |
| Are requirements specified according to the defined rules? | Y | |
| Does the game elements according to the user roles? | Y | |
| Does game elements specific to the task? | Y | |

## Response 3

**General Information**

Project Name: Gamify4Req (Reducing Requirements Ambiguity via Gamification)
Reviewer: Amna Mukhtar
Designation: Associate Manager -Software Development Services
Organization: Contour Software
Review Date: 8-07-2022
Artifacts Reviewed:

d. General Design
e. Models and User Interfaces (Storyboards)
f. Game Elements and Rules

Table Appendix A Response 3

| Artifacts | Assessment (Y/N) | Recommendations (if any) |
|---|---|---|
| **General Design** | | |
| Does the design support development step? | Y | |
| Does the design support project goals? | Y | |
| Is the design feasible with the technology used? | Y | |
| Are the notations, methodology etc. used in the design appropriate? | Y | |
| Does the design have conceptual integrity? | Y | |
| Is the design complex? | N | |
| Is the design flexible? | Y | |
| Does the design address all issues from the requirements? | N | |
| **Models and User Interfaces (Storyboards)** | | |
| When appropriate, are there multiple, consistent, models and/or views that represent the design? | Y | |
| Where there are multiple models of the software are those models consistent with each other? | Y | |
| Have all interfacing systems been identified? | Y | |
| Is the flow of data in all interfaces consistent? | N | |
| Is the design consistent with related artifacts (i.e., other modules, designs, etc.)? | Y | |
| Does user interfaces provide services and inputs and outputs? | | May be |
| Do the colors adhere to industry or organizational style? | Y | |
| Is the wording appropriate and consistent? | Y | |
| **Game Elements and Rules** | | |
| Does the system adhere to the game rules? | Y | |
| Does the system adhere to semantic ambiguity rules? | Y | |
| Are requirements specified according to the defined rules? | Y | |
| Does the game elements according to the user roles? | Y | |
| Does game elements specific to the task? | Y | |

## Response 4 and 5 *(from the reviewers of conference)*

The reviewers provided comments on research paper covering software design and architecture. We have incorporated the changes before submission. The paper is published now. The following figure shows their reviews on paper.

### #24 Gamification Tool Design for Reducing Requirements Ambiguity during Elicitation

### Reviewer #1

**Decision**

Accepted after Modification

**Notes for Autors**

Figures are mixed up. Figure 1 is not referred to by the text. Figure 2 "the system for automated software testing based on ML " needs to be explained.
Figure 5: is it "Proposed Model" or "pseudocode".
Table 1 is referred in text as "table 7" need to be discussed in a separate paragraph before conclusion.

### Reviewer #2

**Decision**

Accepted after Modification

**Notes for Autors**

1.Fig 1 is not a.software architeral design.
2. The describtion of the proposed tool is not a tool design,
3. The author has to specify both the software architeral design and the detailed design according to software engineering basics.

Figure 2 Appendix A Response 4 and 5

# Appendix B Case Study SDA

## *P1-SDA-Manual*

### User response

Project Title: Student Direction App

Filled By (User Role): RE team (ReqEngr + User/Customer/DE)

Mode (Manual/Tool): Manual

Table Appendix B Response P1-M

| Factors | Response |
|---|---|
| Total number of requirements | 41 |
| Time taken to detect ambiguity | 20 mints each |
| Total number of ambiguities | 1 |
| Time taken to reduce/resolve an ambiguity | 5 mints each |
| Time taken to finalize requirements | 50 mints |

Table Appendix B Factors P1-M

| User Experience Factors | Response (User percentage/Good/Fair/Not good) comment if any |
|---|---|
| User involvement in the system (e.g., game elements etc.) | Good |
| Experience of the tool (e.g., efficiency etc.) | - |
| Ambiguity reduction process | Fair |

## *P1-SDA-Gamify4Req*

### User response

Project Title: Student Direction App

Filled By (User Role): RE team (ReqEngr + User/Customer/DE)

Mode (Manual/Tool): Tool / Gamify4Req

Table Appendix B Response P1-G

| Factors | Response |
|---|---|
| Total number of requirements | 41 |
| Time taken to detect ambiguity | 2 secs each |
| Total number of ambiguities | 12 |
| Time taken to reduce/resolve an ambiguity | 35 secs each |
| Time taken to finalize requirements | 27 mints |

Table Appendix B Factors P1-G

| User Experience Factors | Response (User percentage/Good/Fair/Not good) comment if any |
|---|---|
| User involvement in the system (e.g., game elements etc.) | Good |
| Experience of the tool (e.g., efficiency etc.) | Good |
| Ambiguity reduction process | Good |

# Appendix C Case Study GOTCHA

## *P2-GOTCHA-Manual*

### User response

Project Title: GOTCHA
Filled By (User Role): RE team (ReqEngr + User/Customer/DE)
Mode (Manual/Tool): Manual

Table Appendix C Response P2-M

| Factors | Response |
|---|---|
| Total number of requirements | 35 |
| Time taken to detect ambiguity | 10 mints each |
| Total number of ambiguities | 6 |
| Time taken to reduce/resolve an ambiguity | 7 mints each |
| Time taken to finalize requirements | 80 mints |

Table Appendix C factors P2-M

| User Experience Factors | Response (User percentage/Good/Fair/Not good) comment if any |
|---|---|
| User involvement in the system (e.g., game elements etc.) | Good |
| Experience of the tool (e.g., efficiency etc.) | |
| Ambiguity reduction process | Good |

## *P2-GOTCHA-Gamify4Req*

### User response

Project Title: GOTCHA
Filled By (User Role): RE team (ReqEngr + User/Customer/DE)
Mode (Manual/Tool): Tool / Gamify4Req

Table Appendix C Response P2-G

| Factors | Response |
|---|---|
| Total number of requirements | 35 |
| Time taken to detect ambiguity | 2 secs each |
| Total number of ambiguities | 10 |
| Time taken to reduce/resolve an ambiguity | 30 secs each |
| Time taken to finalize requirements | 20 mints |

Table Appendix C Factors P2-G

| User Experience Factors | Response (User percentage/Good/Fair/Not good) comment if any |
|---|---|
| User involvement in the system (e.g., game elements etc.) | Good |
| Experience of the tool (e.g., efficiency etc.) | Good |
| Ambiguity reduction process | Good |

# Appendix D User Manual

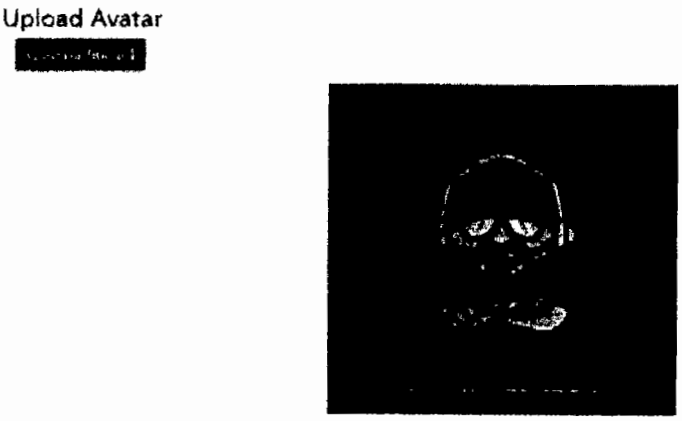
***FOR PROJECT MANAGER***

**STEP 1:**

Login Into the system


**STEP 2:**

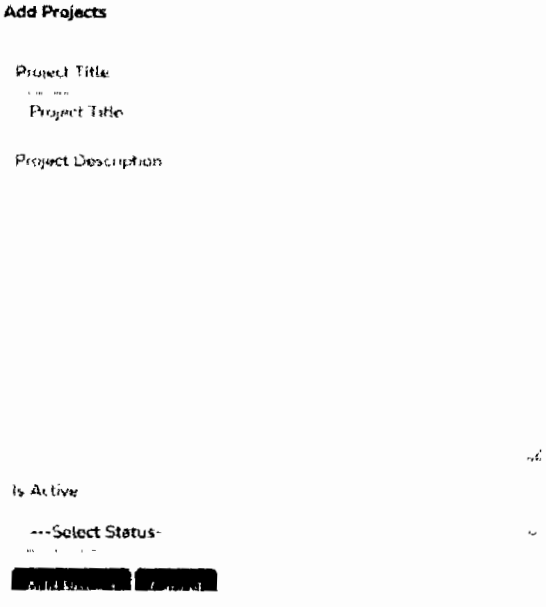
If profile avatar is not selected, then select it.



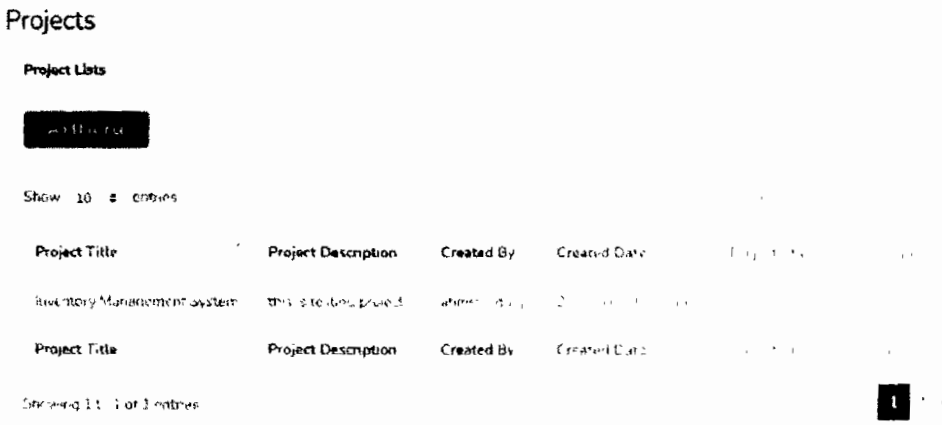

**STEP 3:**

PM can view assigned projects in dashboard section.

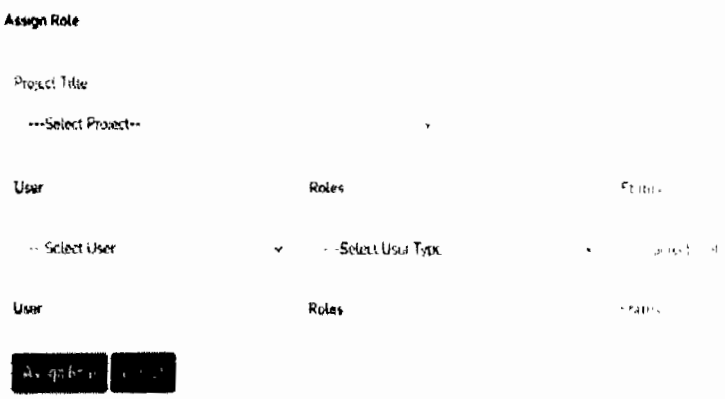

**STEP 4:**
Add project at first level.



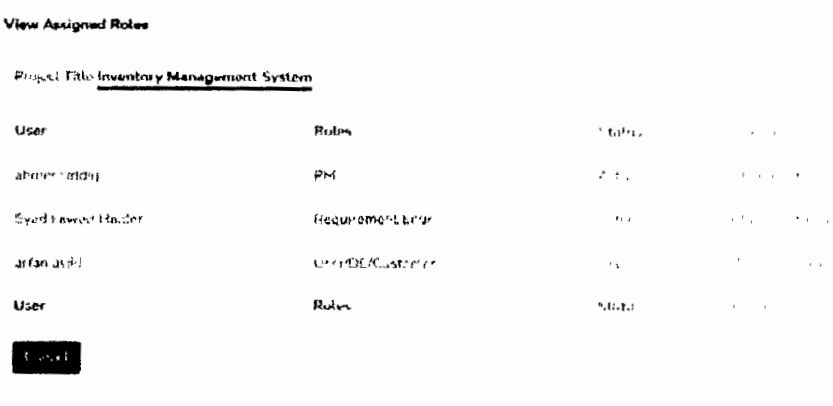

After adding project PM can view projects lists.




**STEP 5:**
Now, add users and ReqEngr. of projects from assign users option.

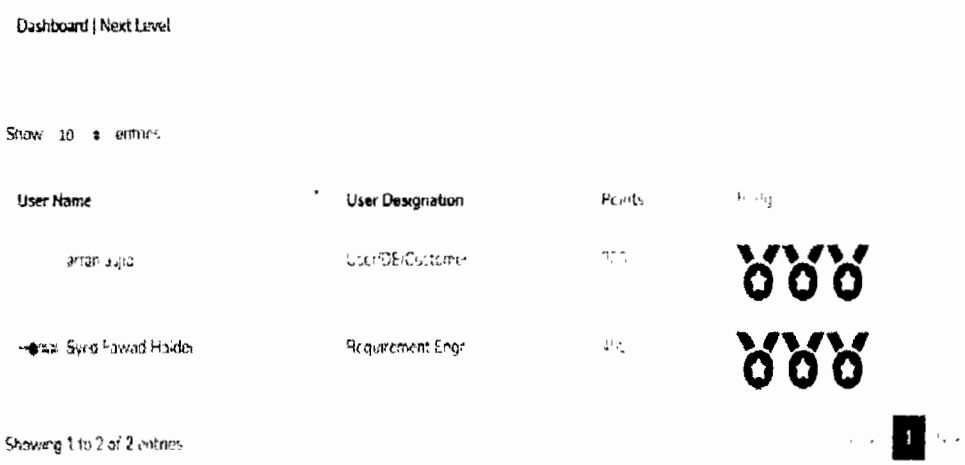



**STEP 6:**
View assigned roles.

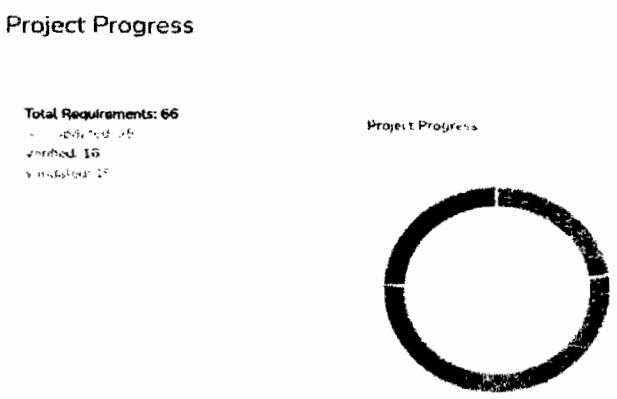


**STEP 7:**
Level 1 is complete after adding project and assigned roles

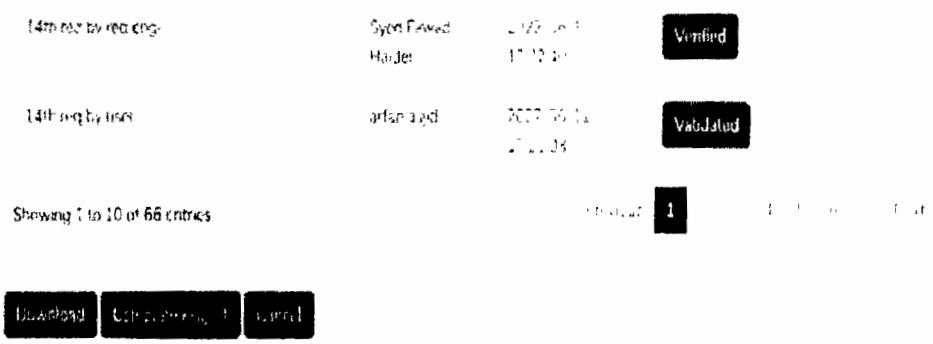

Also, PM can view badges from dashboard.



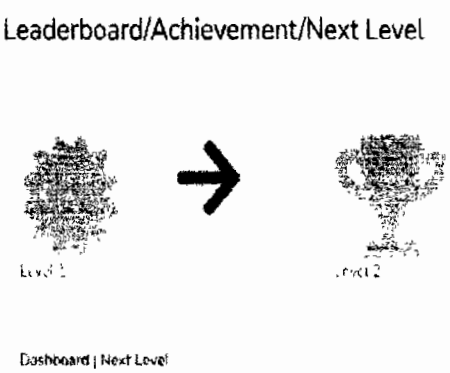

PM can view project progress.




**STEP 8:**
PM can download project requirement document and mark project status as complete to achieve his second level.




PM can view his level 2 status from dashboard

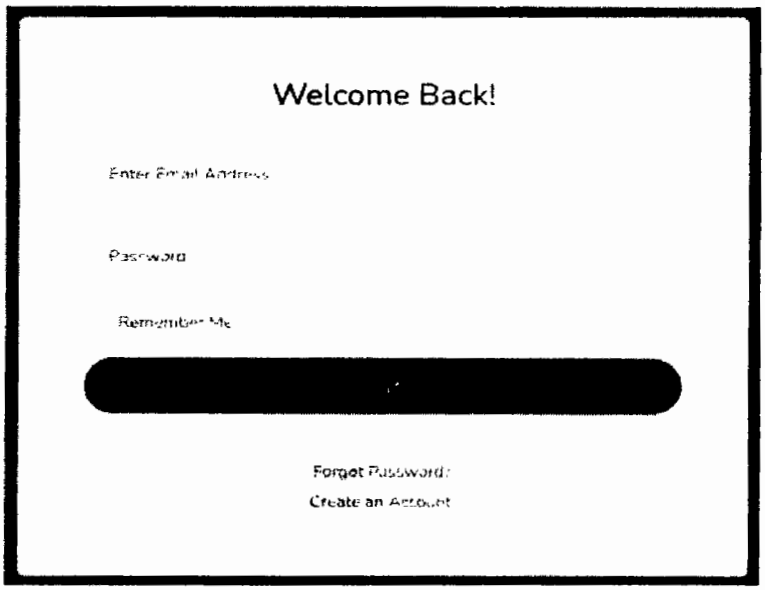



**PM Manual END**

***FOR USER/CUSTOMER/DE***

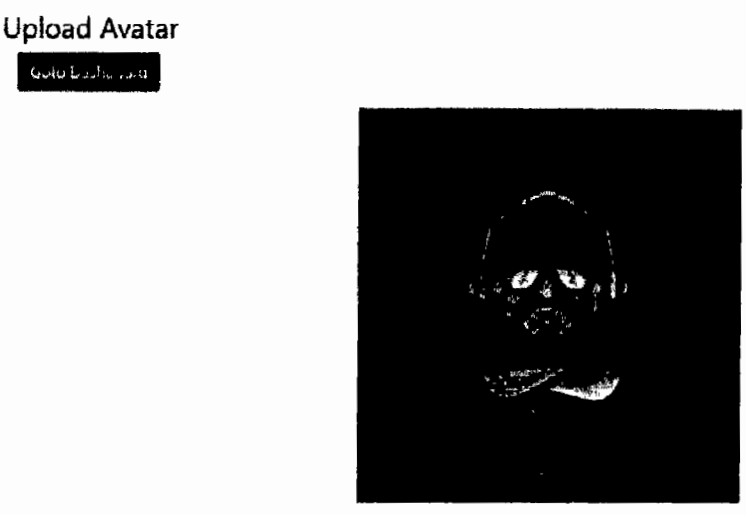
**STEP 1:**

Login Into the system



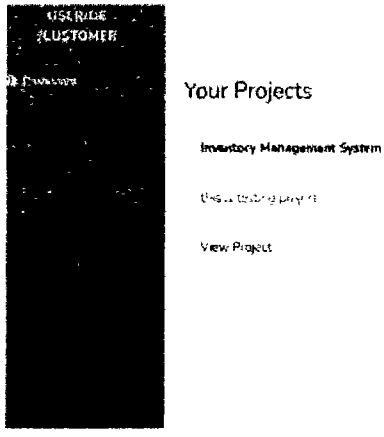

**STEP 2:**

If profile avatar is not selected, then first select it.



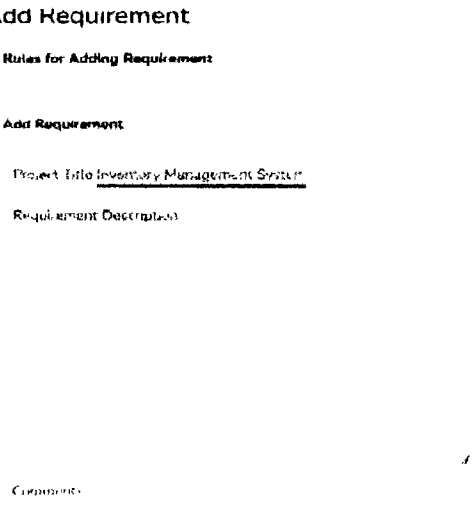

**STEP 3:**

User/Customer/DE can view his assigned projects.



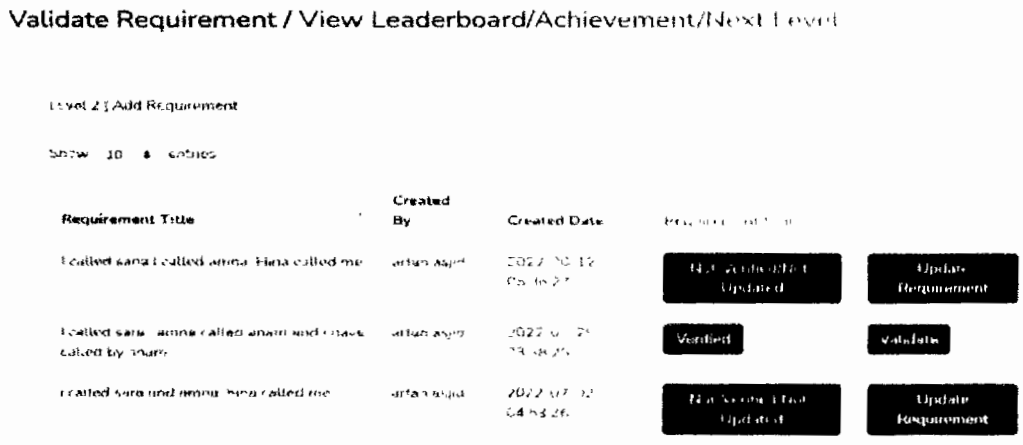

**STEP 4:**

User/Customer/DE can add new requirement from add requirement option.



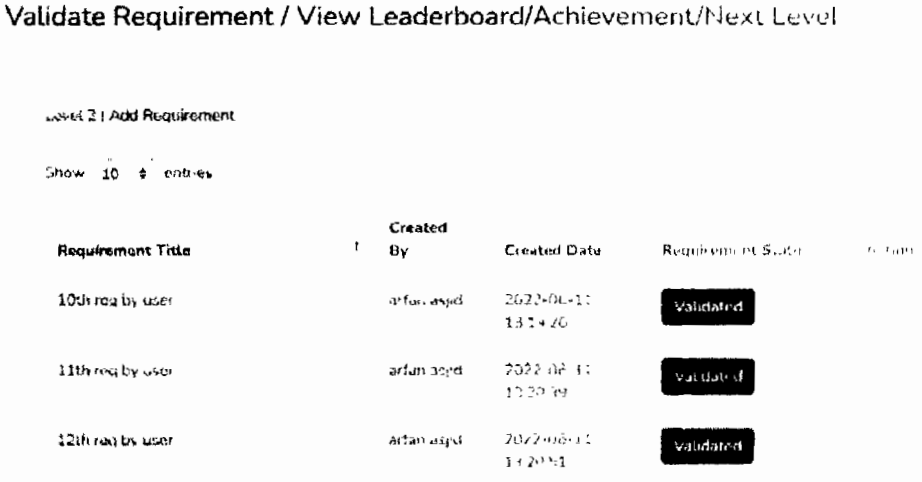

**STEP 5:**

User/Customer/DE can add at least 15 requirements to complete level 1 After completing level one User/Customer/DE can update given requirements and can add new requirements also.



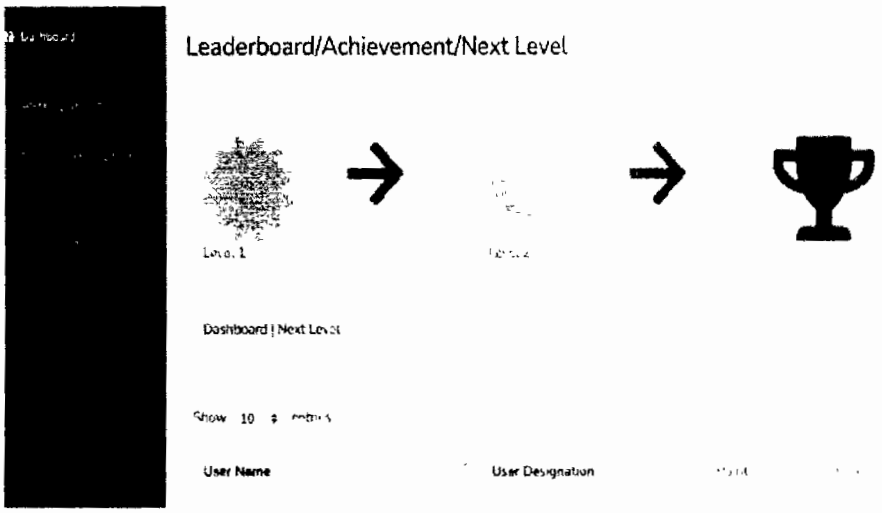

**STEP 6:**

User/Customer/DE can view validated/updated requirements.




**STEP 7:**

User/Customer/DE can view his leaderboard and view badges.

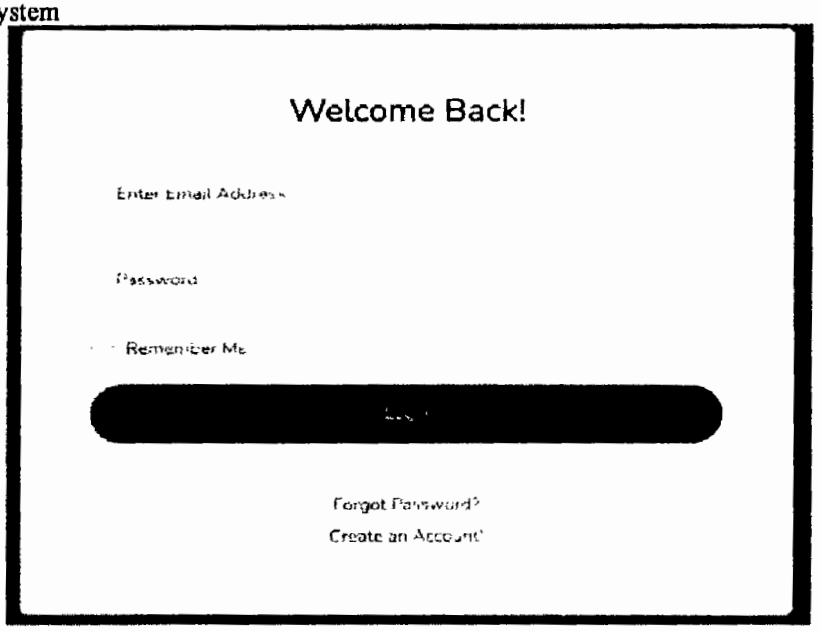



**USER/CUSTOMER/DE Manual END**

***FOR REQENGR.***

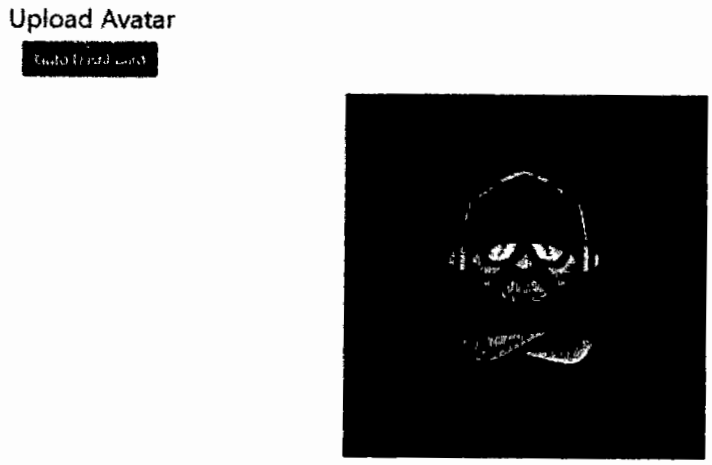
**STEP 1:**

Login Into the system

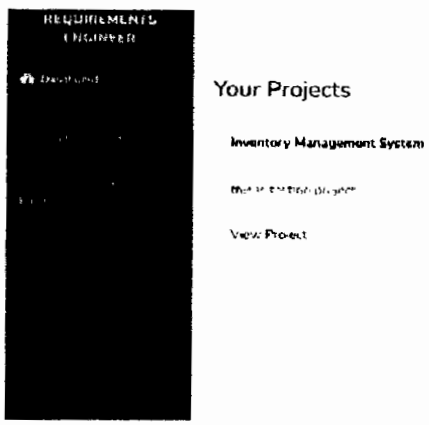



**STEP 2:**

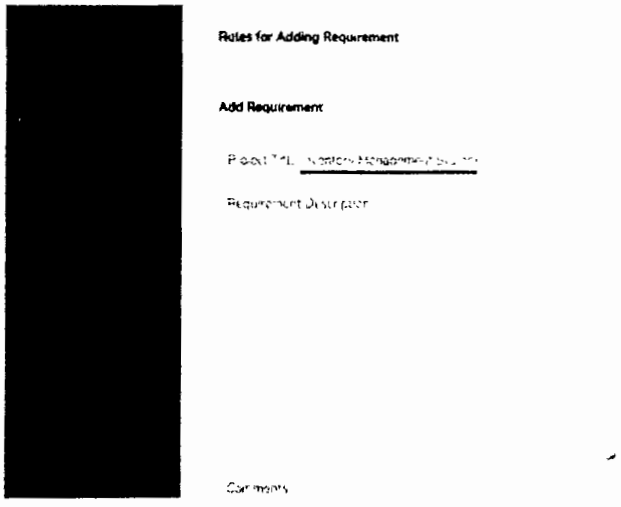
If profile avatar is not set, then select it first.




**STEP 3:**
ReqEngr. can view his assigned projects in dashboard



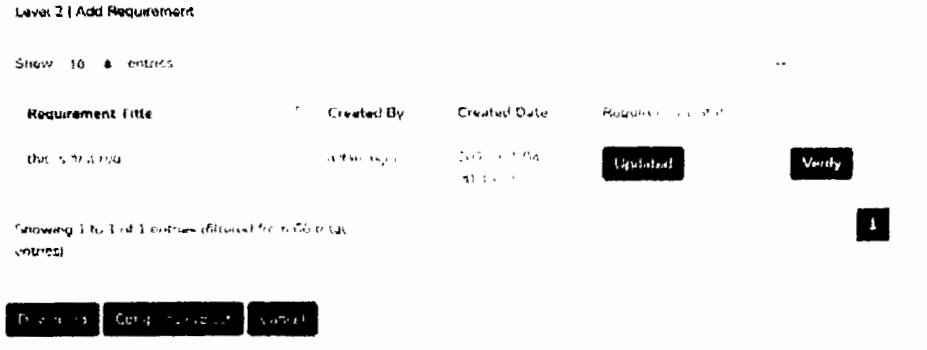

**STEP 4:**
ReqEngr. can give at least 15 requirements to complete level 1




**STEP 5:**

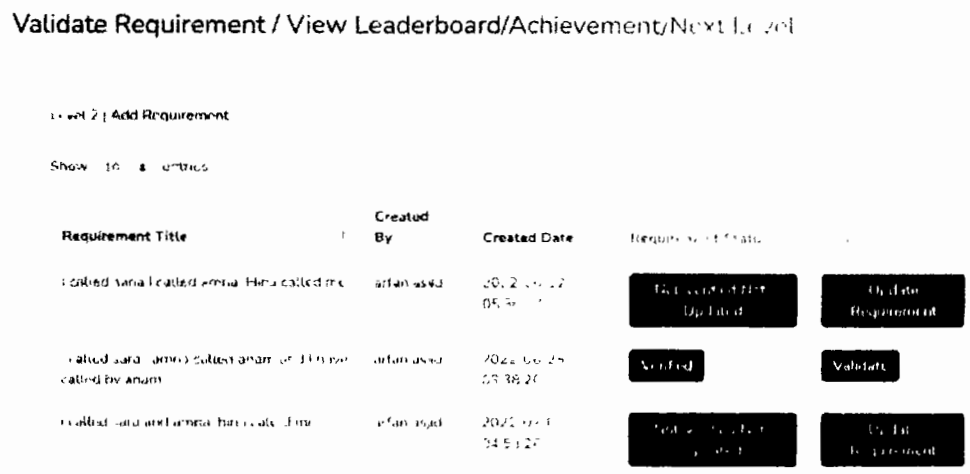
ReqEngr. can verify User/Customer/DE and his own updated requirements.

Validate Requirement / View Leaderboard/Achievement/Next Level



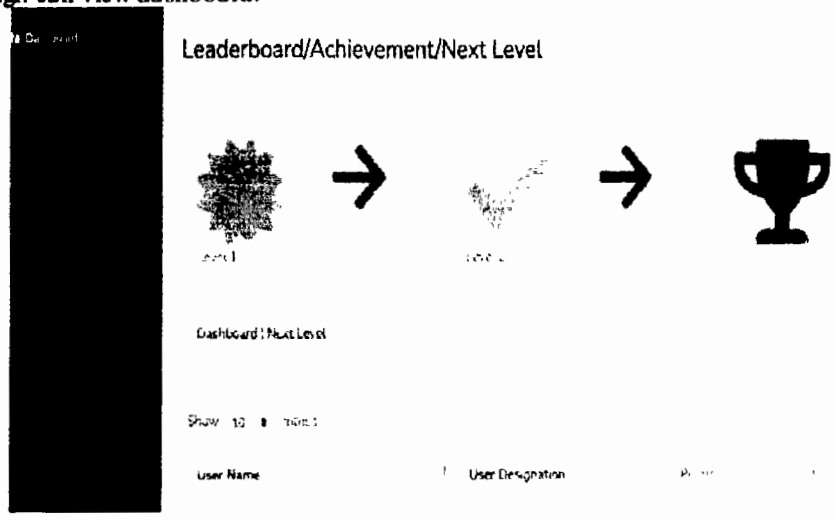

**STEP 6:**

ReqEngr. can update his given requirements.




**STEP 7:**

ReqEngr. can view dashboard.




**Login Details (For Testing)**

Web Link: http://www.gamify4req.com

**Credentials (For Testing)**

PM: muhammadghazan82@gmail.com (Password: Hd22910mg)

ReqEngr.: hafsa_darr@yahoo.com (Password: 514haha514)

User/Customer/DE: hafsa.dar@uog.edu.pk (Password: 709036725)

# Appendix E User Engagement Survey

## Section I - Demographic Information

*This section collects the demographic information of the users*

What role you played while using this system?

- Software Project manager (PM)
- Requirements Engineer (ReqEngr.)
- User/Customer/DE etc.

What is your experience in using similar system(s)?

- <=1 year
- 2-4 years
- 5-7 years
- 8 (and above) years

Which of the following software development methodology is used in your software house?

- Agile - Scrum
- Agile - XP
- Agile - DSDM
- Agile - FDD
- Agile - Kanban
- Traditional - Waterfall
- Traditional - Spiral
- Traditional - Prototype
- Traditonal - RAD
- Traditional - V-Model
- Other

## Section II - Game Elements and Mechanics

*This section inquiries about user engagement in a gamified system, including game elements, rules*

I know the purpose and objective of using this tool

- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree
- N/A

The 'Avatar' well represents my role in the tool

- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree
- N/A

It was easy to select 'Avatar' in my profile

- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree

- N/A

'Levels' kept me curious to know what is coming next in the tool
- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree
- N/A

Each 'Level' has different set of tasks to be performed
- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree
- N/A

It felt rewarding to get 'Points' on adding each requirement
- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree
- N/A

'Points' positively affected my engagement in the tool
- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree
- N/A

'Badge' gives a feeling of achievement in the tool
- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree
- N/A

It is important to view 'Leaderboard' to check the points of other users (players)
- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree
- N/A

'Leaderboard' helps to compare the score of all users (players)
- Strongly Disagree
- Disagree
- Neutral
- Agree

- Strongly Agree
- N/A

PM can view the 'Progress' of each project anytime

- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree
- N/A

Project 'Progress' helps to keep track of the activities being performed by the users

- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree
- N/A

Game elements motivates to use the tool

- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree
- N/A

With the inclusion of game elements, system is more fun to use

- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree
- N/A

I liked to use this game element the most

- Avatar
- Levels
- Points
- Badges
- Leaderboard
- Project Progress

This game element attracted me the least

- Avatar
- Levels
- Points
- Badges
- Leaderboard
- Project Progress

Overall, the tool was fun to use

- Strongly Disagree
- Disagree
- Neutral

- Agree
- Strongly Agree
- N/A

**Section III - Reduction of Requirements Ambiguity**

*This section inquiries about identification and reduction of requirements ambiguity in a gamified system*

The system helped to identify ambiguity in each requirement

- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree
- N/A

The system took less time to identify ambiguity

- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree
- N/A

More ambiguities were identified using this tool

- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree
- N/A

The tool took less time to reduce ambiguity

- Strongly disagree
- Disagree
- Neutral
- Agree
- Strongly agree
- N/A

Overall time taken to identify and reduce requirements ambiguity is less

- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree
- N/A

It was easy to provide, verify, and validate requirements with the help of given specification guidelines and ambiguity rules

- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree
- N/A

It was easy to generate requirements document

- Strongly Disagree
- Disagree
- Neutral
- Agree
- Strongly Agree
- N/A

# Appendix F Statistical Analysis

## *Mann Whitney U Test*

Roles: Group 1 (User/Customer/DE and ReqEngr.)

| | | Statistic | df | p |
|---|---|---|---|---|
| F | Student's t | 0.000 | 2.00 | 1.000 |
| | Mann-Whitney U | 2.00 | | 1.000 |
| P | Student's t | 1.000* | 2.00 | 0.423 |
| | Mann-Whitney U | 1.00 | | 0.617 |
| G | Student's t | 1.000* | 2.00 | 0.423 |
| | Mann-Whitney U | 1.00 | | 0.617 |
| H | Student's t | 1.000* | 2.00 | 0.423 |
| | Mann-Whitney U | 1.00 | | 0.617 |
| I | Student's t | 0.000 | 2.00 | 1.000 |
| | Mann-Whitney U | 2.00 | | 1.000 |
| J | Student's t | 0.000 | 2.00 | 1.000 |
| | Mann-Whitney U | 2.00 | | 1.000 |
| K | Student's t | 1.000* | 2.00 | 0.423 |
| | Mann-Whitney U | 1.00 | | 0.617 |
| L | Student's t | NaN* | | |
| | Mann-Whitney U | | | |
| M | Student's t | NaN* | | |
| | Mann-Whitney U | | | |
| N | Student's t | NaN* | | |
| | Mann-Whitney U | | | |
| O | Student's t | 1.500* | 2.00 | 0.272 |
| | Mann-Whitney U | | | |
| P | Student's t | 0.000 | 2.00 | 1.000 |
| | Mann-Whitney U | | | |
| Q | Student's t | 1.000* | 2.00 | 0.423 |
| | Mann-Whitney U | | | |
| R | Student's t | 0.000 | 2.00 | 1.000 |
| | Mann-Whitney U | | | |
| U | Student's t | 1.000* | 2.00 | 0.423 |
| | Mann-Whitney U | | | |
| V | Student's t | NaN* | | |
| | Mann-Whitney U | | | |
| W | Student's t | 0.784* | 2.00 | 0.515 |
| | Mann-Whitney U | | | |
| X | Student's t | -1.00* | 2.00 | 0.423 |
| | Mann-Whitney U | | | |
| Y | Student's t | 0.447* | 2.00 | 0.698 |
| | Mann-Whitney U | | | |
| Z | Student's t | 0.00* | 2.00 | 0.423 |
| | Mann-Whitney U | | | |
| AA | Student's t | 1.00* | 2.00 | 0.423 |
| | Mann-Whitney U | | | |
| AB | Student's t | NaN* | | |
| | Mann-Whitney U | | | |

Figure 1 Appendix E

Roles: Group 3 (ReqEngr. and PM)

| | | Statistic | df | p |
|---|---|---|---|---|
| E | Student's t | 1.00[a] | 2.00 | 0.423 |
| | Mann-Whitney U | 1.000 | | 0.617 |
| F | Student's t | 1.00[a] | 2.00 | 0.423 |
| | Mann-Whitney U | 1.000 | | 0.617 |
| G | Student's t | 1.00[a] | 2.00 | 0.423 |
| | Mann-Whitney U | 1.000 | | 0.617 |
| H | Student's t | 1.41 | 2.00 | 0.293 |
| | Mann-Whitney U | 0.500 | | 0.414 |
| I | Student's t | 1.00[a] | 2.00 | 0.423 |
| | Mann-Whitney U | 1.000 | | 0.617 |
| J | Student's t | 9.00[a] | 2.00 | 0.012 |
| | Mann-Whitney U | 0.000 | | 0.221 |
| K | Student's t | NaN[a] | | |
| | Mann-Whitney U | | | |
| L | Student's t | NaN[a] | | |
| | Mann-Whitney U | | | |
| M | Student's t | NaN[a] | | |
| | Mann-Whitney U | | | |
| N | Student's t | NaN[a] | | |
| | Mann-Whitney U | | | |
| O | Student's t | -1.50[a] | 2.00 | 0.272 |
| | Mann-Whitney U | | | |
| P | Student's t | 1.00[a] | 2.00 | 0.423 |
| | Mann-Whitney U | | | |
| Q | Student's t | NaN[a] | | |
| | Mann-Whitney U | | | |
| R | Student's t | 1.00[a] | 2.00 | 0.423 |
| | Mann-Whitney U | | | |
| U | Student's t | 1.00[a] | 2.00 | 0.423 |
| | Mann-Whitney U | | | |
| V | Student's t | [illegible] | | |
| | Mann-Whitney U | | | |
| W | Student's t | [illegible] | | |
| | Mann-Whitney U | | | |
| X | Student's t | [illegible] | | |
| | Mann-Whitney U | | | |
| Y | Student's t | [illegible] | | |
| | Mann-Whitney U | | | |
| Z | Student's t | [illegible] | | |
| | Mann-Whitney U | | | |
| AA | Student's t | [illegible] | | |
| | Mann-Whitney U | | | |
| AB | Student's t | [illegible] | | |
| | Mann-Whitney U | | | |

Figure 2 Appendix E