Closing the Gaps in Requirement Gathering for Human-focused AI Software Design
Saurabh Bhattacharya
Chitkara University
Corresponding Author: Saurabh Bhattacharya saurabhbhattacharya18@gmail.com

ARTICLE INFO
Keywords: SDLC, Software Development, Requirement Gathering

Received : 2 October
Revised : 23 October
Accepted : 19 November

©2023 Bhattacharya: This is an open-access article distributed under the terms of the Creative Commons Atribusi 4.0 Internasional

ABSTRACT
Requirement gathering is the technique used by analysts to determine the software demands of users. The process of obtaining requirements is rarely effective, and many software projects have failed because the requirements from consumers were not fully or accurately understood. Most people agree that obtaining requirements is a crucial step in the development process. The chapter is going to analyze the reasons why inadequate requirement collecting causes numerous technology failures. Obtaining requirements is a difficult process. Defects are more prevalent when requirements are solicited. The article aims to examine these difficulties and offer a solution for the many obstacles that come with meeting the demands of the regulatory framework. The absence of stakeholders, ambiguous needs, irregular fluctuations in demand, and insufficient expertise among analysts pose difficulties for experts when it comes to gathering requirements. Preferential system surveys were shown to be one of the most effective methods for acquiring information in a range of situations and domains.

DOI: https://doi.org/10.59890/ijaeam.v1i4.931
https://journal.multitechpublisher.com/index.php/ijaeam
INTRODUCTION

The first step in any software development project is "requirements gathering." Among these are phrases like "What the consumer wants to do?" How does that happen? The process of accumulating a variety of needs—functional, systemic, technical, etc.—is known as "requirements gathering." Poorly written requirements can cause problems throughout the design process since they define the project, but more significantly, they can result in software failing if the goals are not clear, which could result in the project being unsuccessful. Establishing the needs is an essential phase in the building of technological solutions, but there is compelling evidence that the process can and should be improved (Bukhsh et al., 2020). Since most necessary identification occurs at the start of an architecture's design stage, changes can have a significant positive impact on every aspect of the software development process. For requirements gathering for IoT-enabled green towns, an AI approach is proposed to compare requirement collection and administration solutions (Nadeem et al., 2022) (Bhattacharya, 2023).

Good project planning involves many different factors, such as allocating staff to tasks based on their expertise and abilities, identifying relationships among duties, estimating the length of time needed to fulfill the project deadline, and anticipating the requirement to revise the schedule as the project moves forward (Singh & Sambhav, 2023).

LITERATURE REVIEW
Why is it so Important?

Requirements collecting is the process of identifying all of the specific needs needed to complete the project. This procedure occurs during the project's first phase, handling the demands across the project's duration. The words "gathering" might mean to get, amass, reclaim, or find. Throughout the requirements-gathering procedure, the analyst concentrates on understanding the requirements, objectives, and constraints of the system that is going to be produced. The Software Development Lifecycle (SDLC)'s requirements collection and assessment phase is particularly important because it is at this step that managers of projects begin to understand what the project's ultimate consumer wants. During the requirements collecting meetings, the assignment team interacts with the client to go over each need in detail. While programmers usually expect business customers to express their needs as succinctly and precisely as possible, customers in business often expect development teams to provide a solution that satisfies unclear, imprecise, or ambiguous specifications. The needs must be accurately recorded in one location to make the process of building software easier (Ferreira et al., 2018).

It is widely known that one of the most difficult processes in creating software solutions is identifying specifications. This process entails an analyst gathering and modeling information about the key components of a concept. It is possible to greatly increase the effectiveness of products and the planning strategy's efficiency by making modifications to requirement identification approaches.
METHODOLOGY
What Makes Stakeholders Crucial to the Requirement Gathering Process?

The waterfall model in its conventional form consists of discrete, consecutive steps. Although the model has a straight distribution, it is limited by the need to specify every demand at the outset of a project. This makes it impossible to handle unforeseen circumstances or allow for modifications. Agile techniques gained popularity in the IT (Information Technology) sector because they provided mobility and predictability across the whole project creation process. Incorporating stakeholders at the beginning of the developing cycle allows for more opportunities to test the feature and gather input from them at each phase, as opposed to doing it later on (Rasheed et al., 2021) (Rahy & Bass, 2022) (Inayat et al., 2015).

1. Customer-focused Research: Information on the requirements, choices, and needs of stakeholders—especially end users and clients—is extremely useful. Comprehending various viewpoints is important to devise an alternative that conforms to consumer demands and guarantees contentment (Bukhsh et al., 2020).

2. Domain Expertise: To create software that complies with legal and business criteria, stakeholders must acquire knowledge related to the domain. Their knowledge aids in precisely specifying both functional and non-functional requirements.

3. Clear Communication: Development teams, business analysts, and end users can all communicate with each other thanks to stakeholders. Their involvement helps to minimize misconceptions and potential mistakes by ensuring that requirements are communicated correctly (Saeeda et al., 2020). Stakeholders assist in prioritizing requirements based on commercial value and urgency, as well as trade-offs. When trade-offs are required, they may also serve as a guide for decisions, balancing competing criteria to get the best result for the project (Idoughi et al., 2012; Rasheed et al., 2021).

4. Change Management: As a result of shifting business demands or outside influences, requirements may change during the development process. To manage these changes and make sure that the software stays in line with business objectives, stakeholders are crucial (Aldave et al., 2019; Dar et al., 2018).

5. Detection of hazards: Stakeholders are in an excellent place to recognize any hazards related to the assignment. Their advice helps the production team foresee problems and come up with solutions by assisting with proactive risk mitigation.

6. User Acceptance Testing (UAT): Stakeholders engage an active role in UAT in the final phases of a project to confirm that the supplied product satisfies their demands and needs. To improve the product before its official release, this process of feedback is essential.

7. Successful Project Measurements: Stakeholders frequently establish the project’s success parameters. They will be more likely to comprehend these requirements and have them included in the production roadmap if they are involved in the requirement-gathering approach.
RESULT AND DISCUSSION
1. Challenges with Requirement Gathering and Development

The swiftness at which technology is developing these days creates issues for software requirement collecting since it causes requirements for projects to change quickly and user demands to fluctuate. Researchers are globally distributed, which creates difficulties with interaction that are made worse by various daylight hours and customs within cultures. There is increased demand to continually improve and modify requirements due to the widespread use of agile and iterative design approaches. Moreover, the growing prevalence of distant employment exacerbates obstacles to interaction (Baruah, 2015; Bhattacharya, 2023; Rasheed et al., 2021; Zubcoff et al., 2019). In the contemporary rapid computer programming environment, solving these difficulties requires versatile and adaptable techniques that emphasize straightforward interaction, collaborative instruments, and frequent input from stakeholders to assure current, reliable, and thoroughly comprehended specifications. (Curcio et al., 2018; Dafaalla et al., 2022; Ferreira et al., 2018; Rahy & Bass, 2022).

2. How AI Can Be Used in Requirement Gathering Technique

Artificial Intelligence (AI) has been suggested for the majority of software development project management stages, such as evaluating risks, organizing, price and resources calculation, and viability evaluation. The author (Meziane & Vadera, 2010) outlines various suggested applications of evolutionary algorithms, case-specific deduction, systems that utilize knowledge, and artificial neural networks in the development of projects and assesses their efficacy (Budake et al., 2023). AI has the potential to completely transform requirement gathering in the creation of applications by improving and optimizing the whole workflow. Algorithms for “natural language processing” (NLP) allow artificial intelligence (AI) systems to interpret and evaluate data provided by users, as well as identify relevant needs from unorganized information like user comments, papers, and conversations. Artificial intelligence (AI)-powered robots and AI-powered assistants may lead lively discussions and help consumers express their wants and requirements. Early scheduling can be aided by machine learning algorithms that use previous data to forecast probable foreseeable requirements (Dafaalla et al., 2022). AI also helps to prioritize and classify needs so that development activities are in line with strategic goals (Ahmad et al., 2020). Evaluation of sentiment technologies provides continuous refinement of requirements by measuring consumer happiness and inclinations. Self-driving technologies can recognize contradictions and uncertainties, which can lead to more accurate and dependable specifications. Artificial Intelligence expedites the requirement collecting procedure via automation of monotonous operations, freeing up project managers to concentrate on creative and inventive solving issues. In the end, using AI for requirements collecting methods improves application development's
effectiveness, precision, and flexibility. AI may simplify the process of producing proposals and determine the boundaries of the project by helping comprehend the demands of consumers and company stakeholders (Ali & Lai, 2017; Inayat et al., 2015; Zowghi & Coulin, n.d.). AI technologies that reduce uncertainty regarding intended performance, such as ChatGPT and Bard, can be used to design software concepts. The criteria may be determined by the written content through the application of Machine Learning (ML) or Natural Language Processing (NLP) to analyze the usage patterns of users. The effort that researchers need to spend crafting user stories and interpreting business requirements can be reduced by using AI solutions. (Sofian et al., 2022). While generative AI techniques can develop specifications that satisfy the project's magnitude, business researchers and product administrators can effectively summarize the task (Feldt et al., 2018) (Barenkamp et al., 2020).

CONCLUSION

Modern requirements collecting practices in software creation emphasize the use of modern technologies, agile processes, and user interaction as a top priority. Iterative feedback processes are emphasized because they enable flexible adaptations to changing user requirements. Reliability is ensured by using based on artificial intelligence analytics to help understand complicated requirements (Kramer, 2018; Krishna & Gopinath, 2021; Walia & Carver, 2009). Working from home presents issues that are addressed by interactive and working environments. Reducing uncertainty and improving transparency are achieved by including participatory meetings, designs, and user stories. To promote mutual comprehension among engineers and stakeholders, there must be regular interaction and comprehensive record-keeping. Using state-of-the-art technologies in conjunction with an agile, centered user strategy improves the requirement collecting method in the rapidly changing software engineering environment of tomorrow (Makridakis, 2017).

REFERENCES


