INTERNATIONAL JOURNAL of RENEWABLE ENERGY RESEARCH
C. R. Komala et al., Vol.13, No.4, December, 2023

Innovative Cost Estimation for Agile Technology: A Novel Energy Storage Technique Incorporating Modified Planning Poker

C. R. Komala[1 *], H. K. Sowmya[2], R. Aruna[3],
S. Arun Kumar[4], Ramya Maranan[5],
B. Prathusha Laxmi[6], Nageswara Rao Medikondu[7]

[*1]Department of Information Science and Engineering, HKBK College of Engineering, Bangalore, Karnataka 560045, India,
[2]Department of Artificial Intelligence And Machine Learning, New Horizon College of Engineering, Bengaluru, Karnataka 560103, India.
[3]Department of Electronics and Communication Engineering, AMC Engineering College, Bengaluru, Karnataka 560083, India.
[4]Department of Computer Science and Engineering, Nagarjuna College of Engineering and Technology, Bengaluru, Karnataka 562164, India.
[5]Department of Research and Innovation, Saveetha School of Engineering, SIMATS, Chennai, Tamil Nadu 602105, India.
[6]Department of Artificial Intelligence and Data Science, R.M.K. College of Engineering and Technology, Puduvoyal, Tamil Nadu 601206, India.
[7]Department of Mechanical Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh 522302, India.

(crkomala576@gmail.com, somyahk345@gmail.com, arunar789@gmail.com, arunakumar551@gmail.com, ramyamaranan134@gmail.com, prathushalaxmib143@gmail.com, nageswararao778@gmail.com)

*Corresponding Author; Komala C R, Department of Information Science and Engineering, HKBK College of Engineering, Bangalore, Karnataka 560045, India, crkomala576@gmail.com

Received: 09.06.x2023xxx Accepted:30.10.2023

Abstract: Effort estimation is a critical aspect of every software creation project, irrespective of its development discipline, is it traditional or agile. Despite the initial limited project knowledge, accurate estimation of workload and effort holds immense significance for project success. This study addresses the challenges surrounding precise effort estimation and introduces the poker planning method utilizing a modified Fibonacci sequence as a potential solution. The aim of this paper is to examine the behavior and effectiveness of the modified Fibonacci sequence in the context of cost estimation within the agile methodology. By leveraging historical data, expert opinions, and an iterative approach, the proposed technique is evaluated and compared against traditional methods. The results provide insights into the performance of the modified Fibonacci sequence, offering a valuable contribution to enhancing accuracy in cost estimation for agile software development projects.

Keywords: Agile, Software development, Cost estimation, poker planning.

1. Introduction

Since the 1940s, the realm of software development has been fraught with a perennial challenge - the accurate estimation of project costs and efforts. Despite the passage of time, this challenge has endured, and progress in this field has remained notably limited . The quest for precise cost estimation has remained a paramount pursuit for software companies, with far-reaching implications that extend beyond the balance sheets. The ability to discern and calculate project costs accurately can be a pivotal factor in the financial success of software enterprises, positioning them strategically within the competitive software market[1].Within the context of the financial services sector, the practical implementation of agile practices in project management, dissecting the experiences of nine agile teams. Employing grounded theory analysis of semi-structured interviews, we can unveils thematic threads encompassing accountability, team dynamics within the organizational context, human and technical facets influencing agile teams, and the ramifications of agile practice integration. This intricate accountability journey is molded by the convergence of team perceptions, technical implementation encounters, and the interplay of interactions [2]. In response to this long-standing challenge, agile-based project management methodologies have emerged as a beacon of hope. At the core of the agile approach lie a commitment to rapid development and, crucially, a determination to gauge project costs with heightened precision [3].By prioritizing essential elements while discarding extraneous features, agile methodologies chart an adaptable and iterative course. Central to this agile expedition is the agile project manager, a pivotal role within the agile framework, often known by various alternative titles that reflect their multifaceted responsibilities within the project ecosystem. These agile project managers are architects of optimization, orchestrating projects in a manner that aligns with the organization's goals and the team's strengths. These roles encompass dimensions of a protector, program manager, supervisor, coach, mentor, and local project manager, among others (Figure 1).

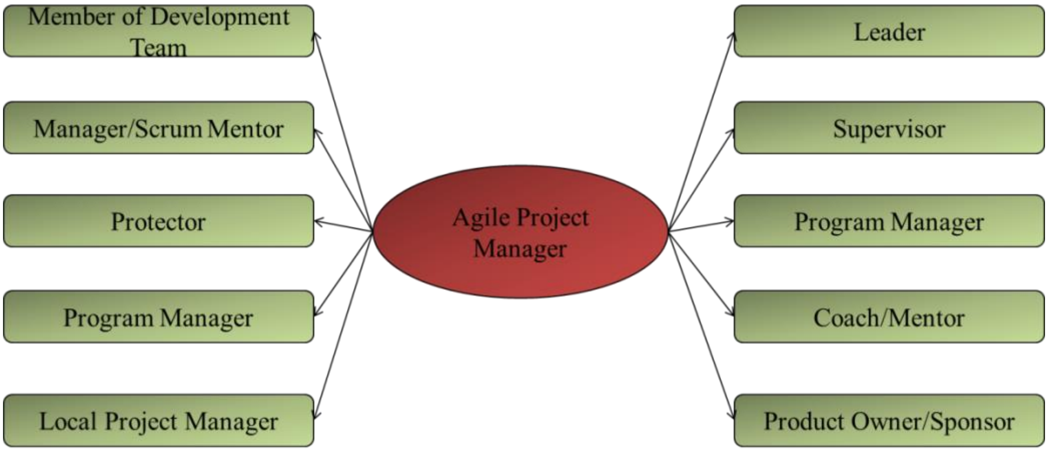

Figure 1: Diverse profile of Agile Project Manager

Agile project managers have a range of cost estimation techniques to choose from based on project scope, duration, and team expertise. These techniques fall into three main categories: Empirical Techniques, which utilize historical data and educated guesses, exemplified by the Delphi method; Heuristic Techniques, which yield optimal decisions using heuristic approaches, like the COCOMO model; and Analytical Techniques, which break tasks into components and solve them systematically, as seen in the Halsted software science approach. Each approach offers distinct insights for accurate cost estimation in agile projects. An illustration of a heuristic technique is the COCOMO model shown in figure 1.

Numerous software cost estimation models, including functional points, COCOMO, and wideband Delphi, are at the disposal of project managers. Among them, the poker planning is relying on the Fibonacci sequence {1, 1, 2, 3, 5, 8, 31, 21, 34, 55….}. Besides these models, the software experts often estimates the cost based on their team's ability and accuracy or historical facts and figures. Additionally, other techniques are influenced by the software development methodology, as estimation techniques are often recommend by these methodologies.

Table1: Comparison of different methods for cost estimation.

Methods	**Structure**	**Anonymity**	**Interaction**	**Overhead**
Delphi	Complex	Confidential	Limited Interaction	Significant
wideband Delphi	Moderate	Limited	Controlled Interaction	Controlled
Planning Poker	Simple	Non-identifiable	Collaborative	Determined
Unstructured Group	Simple	Unidentifiable	Collaborative	Determined
Statistical group	Simple	Confidential	Limited Interaction	Determined
Decision markets	Complex	Confidential	Limited Interaction	Controlled

The paper delves into a comparative exploration of different cost estimation methods, including Delphi, wideband Delphi, Planning Poker, unstructured group, statistical group, and decision markets (Table 1). By evaluating these methods based on structural characteristics, anonymity, interaction, and overhead, the study seeks to shed light on the nuances and advantages of each approach.

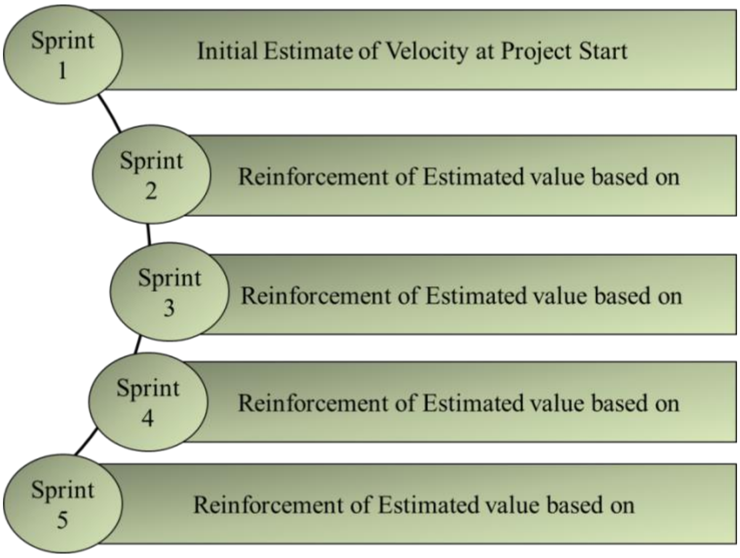

Figure 2: Sprint-wise Cost Estimation Trends

Figure 2 demonstrates the computation of cost during several phases by using agile methodology of the sprint. During the initial stage, an initial cost is estimation is established by using available data and previous experiences, subsequently, the gained knowledge from the prior stage is employed to perform further calculations. Initially, the input of sprint is 1, employed to calculate the modified cost of sprint 2, thereby we can say that the input at x_{i+1} is $x_{i.}$

This paper introduces the poker planning technique with a modified Fibonacci sequence. Section 2 of the study provides a synopsis of previously published works., detailed description of the poker planning estimation technique with a new Fibonacci sequence in section 3. Then section 4 compared the consensus and average of raw data on the actual and estimated size of the project. We surveyed this approach. We have published the results obtained during this survey and compared the actual and relative errors. The section 5 indicated the future work which aims to indicate the plan of combining the poker planning technique with another estimation technique to improve the cost estimation accuracy

2. Literature Review:

In reference [4] explores the management strategies moderate the relationship between project complexity, dynamism, and success. Analyzing data from software professionals, the study highlights the crucial role of managing dynamism for project success. It reveals that an agility-based approach is particularly effective in mitigating the negative effects of dynamism on project outcomes. This research contributes to understanding how project management aligns with project characteristics, offering insights for enhancing success. This study [5] addresses the complex challenges of managing and estimating agile projects in software companies, especially within the Scrum framework. The paper introduces an Intelligent Recommender and Decision Support System (IRDSS) to enhance project estimation by considering cost, time, and resource recommendations. The formal specification of IRDSS using the Z language adds rigor to the approach. Through an experiment involving fifteen web projects, the proposed system demonstrates improved estimation accuracy compared to traditional methods like Delphi and Planning Poker, offering potential directions for more effective software project development in the Scrum community.In reference [6], the authors express concern about the inherent flexibility of the agile methodology, which makes it challenging to accurately predict initial phase timelines and costs. To address this, a method is proposed that assesses project scope by focusing solely on critical elements. In [7], the authors highlight the significance of assembling a highly skilled and self-organizing development team during project development, underscoring the pivotal role of the manager.

This study [8] addresses accurate software project effort prediction, with most prior work focusing on algorithmic models like COCOMO. Introducing an alternative, it suggests using analogies by characterizing projects based on features. Completed projects form a repository, and the method involves finding similar projects for prediction, aided by a PC tool called ANGEL. This approach offers project managers a valuable complement to estimation techniques for improved software project management. In [9] an author focuses on software size estimation's importance in project planning and introduces a stepwise linear regression model for estimating board-based desktop game sizes. The study [10] presents Global Software Development (GSD) and its reliance on Software Project Management (SPM) for success. It identifies the need to understand and address the challenges of SPM in GSD, prompting the development of innovative solutions. Through a Systematic Mapping Study (SMS) analyzing 84 research papers, the paper categorizes SPM approaches for GSD based on six criteria. Findings underscore the growing in SPM for GSD since 2006, with a significant focus on coordination, planning, monitoring, and estimation techniques.

During this research, we have gone through various articles and research papers of IEEE, Springer, ACM, and many more based on that we decided on the research gaps provided by those authors and hence have tried to fill that research gap to the best of our knowledge.

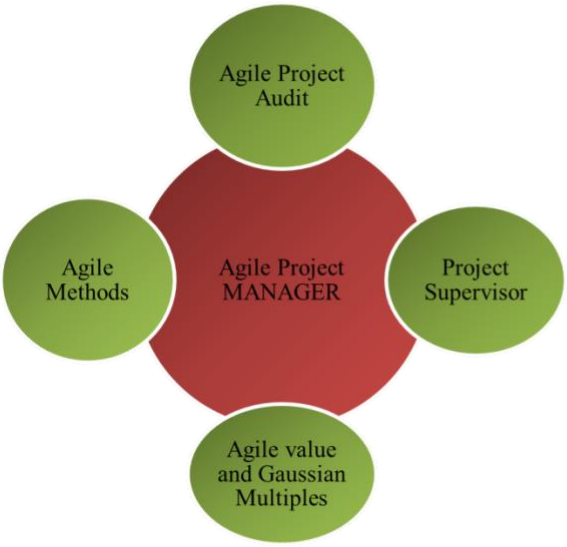

Figure 3: Integration stragies in Agile [8,15]

Figure 3 shows the agile integration concept streamlines interconnected elements, reducing redundancy and evaluation parameters for efficiency. Agile Project Managers play a central role in method selection and project auditing, aligning with agile values of collaboration and adaptability. Gaussian multipliers guide this integration, emphasizing iterative improvement. The Project Supervisor's involvement ensures cohesive decision-making. Agile project audit processes enhance transparency and accountability. This approach resonates with agile principles, optimizing resource allocation and responsiveness. It embodies agile's essence, responding to change over rigid plans. Overall, agile integration promotes synergy, efficiency, and continuous enhancement within the project framework.

Figure 4: Framework for Agile Project Evaluation [9]

Figure 4 depicts the comprehensive conceptual framework for an entire agile project. The process commences with the Product Vision Audit, during which the project's cost is estimated by the auditing team. This audited cost is then compared with existing historical data [10] to gauge its alignment. Subsequently, the framework progresses through, Product Backlog Audit is involves a meticulous examination of the product backlog, ensuring its alignment with the project's objectives and requirements. Release Backlog Audit, verifying that it encompasses the necessary features and functionalities to meet the project's goals. The evaluation extends to the sprint backlog, ensuring that it accurately reflects the planned tasks for the upcoming sprint cycle. The final stage involves refining the product based on the outcomes of the previous audits, ensuring its optimal alignment with project objectives. This sequential process ensures a thorough evaluation at each stage, promoting alignment, transparency, and effective project execution within the agile methodology. The 67% of the developers are using Lean agile concepts to develop their project whereas 55% are using agile scrum methodology which uses processes like and 42% are using Agile XP technique [12].

3. Proposed Methodology

This methodology involves software developers and business experts who act as customer representatives. The process works by reaching a consensus (which is defined as a general agreement between all parties). Since here the case is not binary, so there could be multiple predictions based on the whole number of present on the board at the time of cost estimation [13]. Hence the result with minimum error and least risk would be only accepted.

As we are aware of the Fibonacci sequence (1, 1, 2, 3, 5, 8, 13, 21, 34……) this goes by adding the previous two terms of series to generate a new next term. Now if observe the series we can see that the third term is 100 % more than the last term but the fourth term is only 50% more than the previous term similarly we observe the higher terms of the series we can see that they only grow by 66%.

If we consider an example by taking a brick of 50kg in one hand and 100kg in the other then it is very easy to identify which one is heavier. But if we take a case where we have a brick of 20kg in one hand and 22kg in the other then it becomes difficult to tell the difference. Though the variance is 10%. Similarly, if we have a brick of 1kg in one hand and 1.10kg in the other then it becomes very difficult to state the difference, but here also the variance is 10% [14]. Hence, we can see that it becomes very difficult to identify the difference if the numbers are close. So, the standard Fibonacci sequence fails.

The numbers in the modified sequence go as (0, 0.5, 1,2,3,5,8,13,25,50,125….) We manipulated the numbers from the 9th term the 21 could also have been rounded to 20 but we followed Weber's law (which states the difference between two objects is identified in terms of percentage). Hence the new value 25 is approximately 92.3% greater than the previous one. The next value is 100% greater and after that, the difference is 150%. The modified Fibonacci values represent various instances of the project, including could include the estimated total amount of ideal days, narrative points, and additional units.

While experimenting the cards have been interpreted as 0 for the task already completed. 0.5 indicates the task of smaller size. The initial numbers 5,8,13 indicates a medium-sized task. The value 25, and 50 indicated large tasks (hence the values are so kept 100% greater to identify the difference between medium-sized and large-sized tasks). The value 125 indicates a very large task hence the value is 150% more than the previous one. We have also included various other symbols such as < > for a coffee break so to make the experiment more interesting. The symbol (?) indicates a task about which we do not have an idea of how much time duration it could be completed. Agile software development is a collection of several benefits as it focuses more on time, and it is also customer-oriented, incremental, modular, and iterative . The author proposed a new context where the estimation of software is done
using a heuristically enhanced fusion model. There are so many approaches under the umbrella of Agile and also exist many techniques which have been used by researchers for estimating effort.

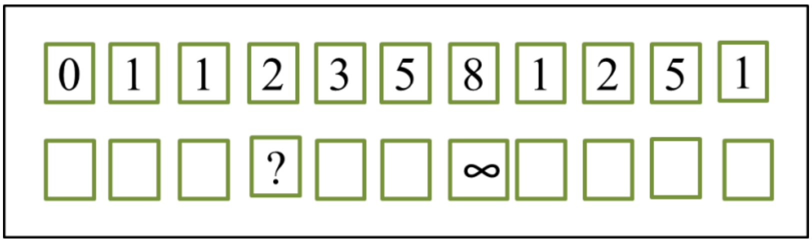

Figure 7: Modified Fibonacci sequence for poker planning

Algorithm Estimate User Story Cost:
Input: List of user stories
Output: Estimated costs for each user story
For each userStory in ListOfUserStories do
    Repeat:
        Moderator selects userStory
        Customer_Representative Explains (userStory)
        Set: Team_Estimates = []
        For each team in Development Teams **do**
            Estimate = Team_Estimate (team, userStory)
            Team_Estimates. Append (estimate)
        End for
        If (Consensus not Reached (Team_Estimates)) then

```
                Outliers = Identify_Outliers (Team_Estimates)
                For outlier in Outliers:
                        Explain And Defend (outlier)
                End for
        End for
    Until Consensus Reached (Team_Estimates)
End for
```

The above algorithm outlines an agile approach for estimating user story development cost. It begins by considering a list of user stories that require estimation. For each user story, the process commences with the selection of a user story by a moderator. Subsequently, a customer representative provides a comprehensive explanation of the chosen user story, addressing any queries raised by the development team. This initial understanding forms the basis for estimating the cost of implementation by each participating team. The estimates are collected and recorded for evaluation. Should the development teams encounter discrepancies and fail to reach a consensus on the estimated cost, the algorithm incorporates a mechanism for resolution. Outliers, which represent estimates substantially differing from the norm, are identified among the team estimates. To ensure transparency and accuracy, the outliers are then requested to elucidate and justify their valuation of the user story. This step encourages a thorough discussion and examination of differing perspectives, contributing to a more informed estimation process. The algorithm iterates through these steps, progressing from story to story, until a consensus is successfully achieved among the development teams regarding the estimated cost of each user story. This iterative nature of the process emphasizes the dynamic and collaborative nature of agile development, allowing for continual refinement of estimates. In essence, this algorithm provides a structured and adaptable framework for estimating user story costs, promoting effective communication, collaboration, and informed decision-making within agile software development practices.

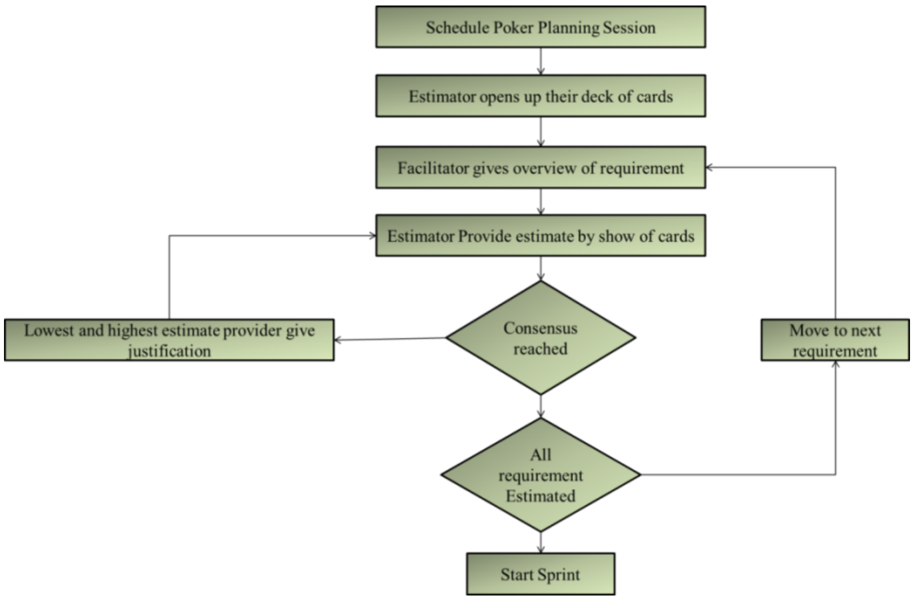

Figure 8: Block Diagram of Agile User Story Cost Estimation Approach

We have taken the PROMISE repository dataset along with ISBSG (International Software Benchmarking Standards Group) dataset along with some publicly available datasets from Kaggle. The Purpose of using these datasets was that they have been already used in the past for conducting various empirical research-related works. The ISBSG dataset has more than 4000 software Projects from across the globe, combining all we have a dataset of more than 7000 projects.

Table 2: Compilation of Software Projects from Different Sources

Serial Number	Source	Number of Software Projects
1	ISBSG	4100
2	PROMISE	2186

3	Kaggle	1678
Total		7964

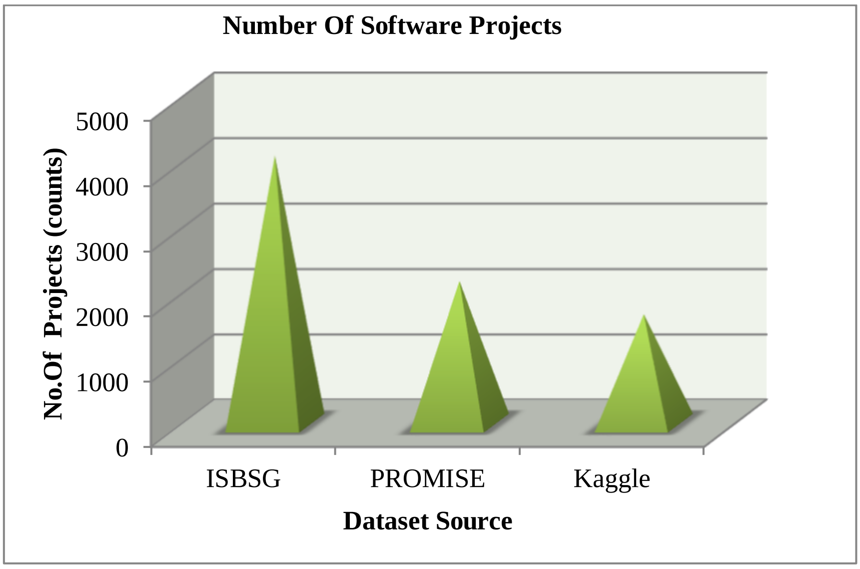

Figure 9: Project Counts Comparison across Data Sources

The above figure 9 is a graphical representation of various sources (ISBSG, PROMISE dataset, Kaggle Online Platform) through which the data has been collected.

4. Results

Parameter of result comparison

The results have been compared based on variance, several estimations round required, duration or time taken during each estimation, the tendency of estimation, and at the end combined estimation tendency has also been used.

(a) Variance: The difference between the final estimated value and every single estimation is termed as a variance in this experiment.
Ω= {1, 2, 3, 5, 8, 13, 25, 50} where X € Ω

We used the assumption that the sample space was normalized by converting each element's value to its location on the estimate scale.

Hence normalized space looks like
Ω= {1, 2, 3, 4, 5, 6, 7, 8} where X€ Ω

$$\text{Variance} = \frac{1}{m} * \sum_{j=1}^{m} (Xi - ev)(xi - ev)$$

Where; m=number of involved estimations
Xi= position of an estimation
ev=final estimated value

Table 3: Variance table according to sample data

Amount of single estimation	Final estimated value	Variance
25,25,25,25,25,25,25	25	0
25,13,8,5,1,5,13	13	≈58.26
13,13,13,8,13,25,13	13	≈38.57

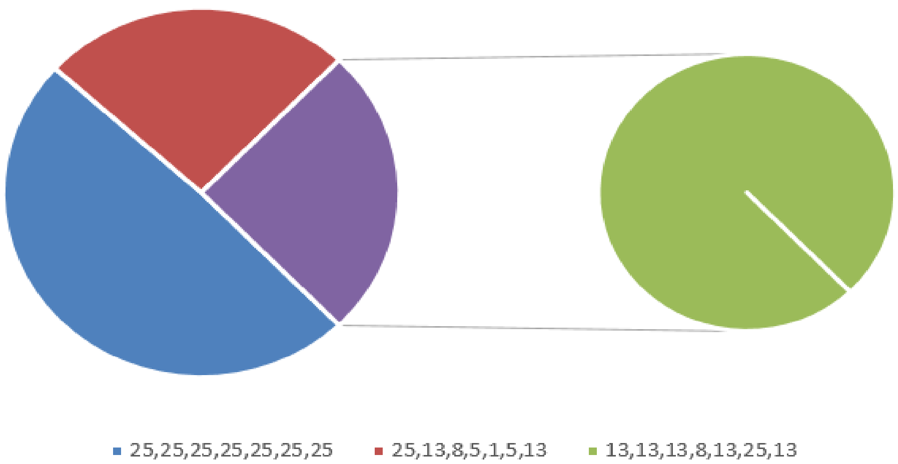

Figure 10: Sample Data Visualization of Estimated Value versus Variance

The above figure is a graphical representation of the sample variance that we have calculated according to the sample data. The green part represents the Final estimated value while the blue, purple, and red part represents the initial value provided as input.

(b) Several estimation Rounds:
The Number of poker played concerning a single requirement is measured by this parameter. A suitable estimation is termed where one estimation round is sufficient.

(c) Duration of estimation:
The length of time between the moderators starts explaining the first story and the end of the final committed estimation value. The mathematical average of the previous estimates of the needs was employed.
It is the measurement of whether the estimation is oversized, undersized, or perfectly suitable. It is also normalized by exchanging all the values by their position on the estimated scale.

$g = \frac{x-ev}{|x-ev|} * (x - sw)^2$ if |x-ev|≠0

where; g is defined as the gap between the suggested and final estimate
x is the position of the estimate on the estimation scale
ev is the final estimated value
sw is the individual suggested estimate.
If |x-ev|=0 then the value of g is o.
The negative value indicates the estimation is undersized, while a positive value is a sign of oversized estimation.

(d) Combined Estimation Tendency:
It has to do with a specific requirement estimate. It is the measurement that indicates whether the actual functional effort is higher or lower than the estimated functional effort of the requirement.

Table 4: Table indicating various variances at different modified Fibonacci instances

Possible Estimated Value	1	2	3	5	8	13	25	50
Variance (≈)	18,7	12	6,5	4,6	2,9	5,1	8,4	13,6

The above table represents the variance that has been calculated using

$$\text{Variance} = \frac{1}{m} * \sum_{j=1}^{m} (X_i - ev)(x_i - ev)$$

Where; m=number of involved estimations
X_i= position of an estimation
ev=final estimated value
against the proposed Fibonacci sequence for cost estimation according to planning poker.

Table 5: Table indicating the absolute and relative error while calculating by modified Fibonacci sequence (Consensus) and average methods

P#	Estimation type	Estimation size	Actual size	Absolute Error	Relative Error
P1	Average	13.5	15	1.5	0.1
P2	Consensus	28	30	2	0.066
P3	Consensus	14	18	4	0.222
P4	Average	12	17	5	0.2941
P5	Consensus	15	18	3	0.1666
P6	Consensus	16.5	19	2.5	0.13157
P7	Average	19.5	25	5.5	0.22
P8	Average	38	41	3	0.0731
P9	Consensus	52	45	7	0.1556
P10	Consensus	61	56	5	0.089

We have calculated the actual and relative error to measure the deviation between actual and estimated values of the estimation.
The actual cost derived from the project using the chosen estimation method (average or consensus), and the absolute error metrics measures the absolute difference between the estimated cost and the actual cost, measuring the accuracy of the estimation.

The substantial disparity among the predicted and actual values is known as the absolute error, while the relative error is the given as RE= $\frac{\text{AbsoluteError}}{\text{ActualSize}}$

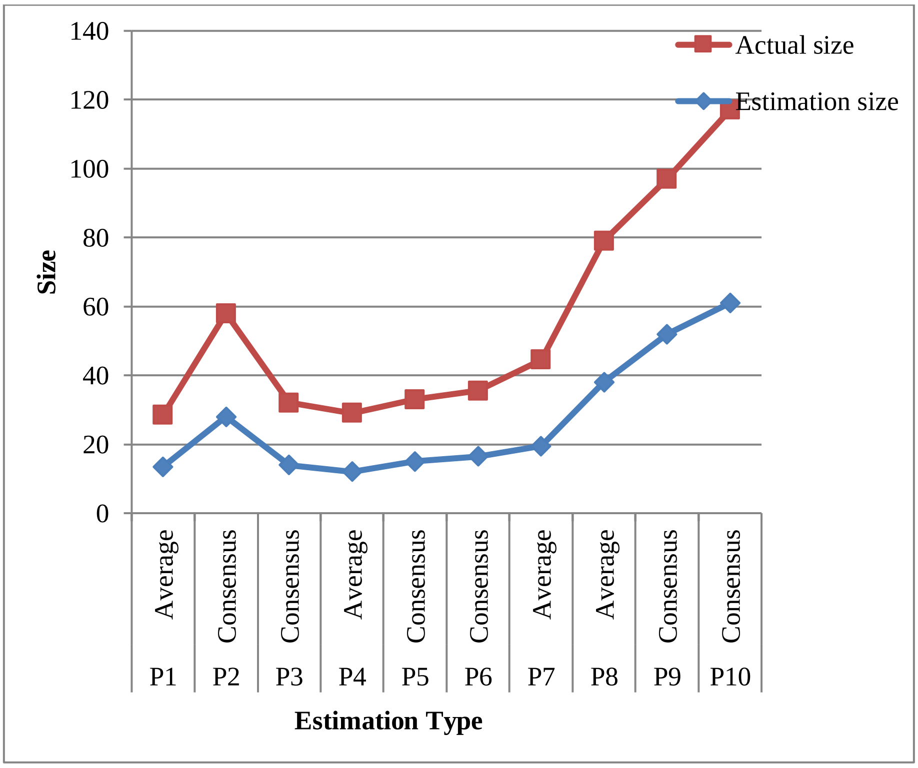

Figure 11a: Estimation vs. Actual Size for Different Estimation Types

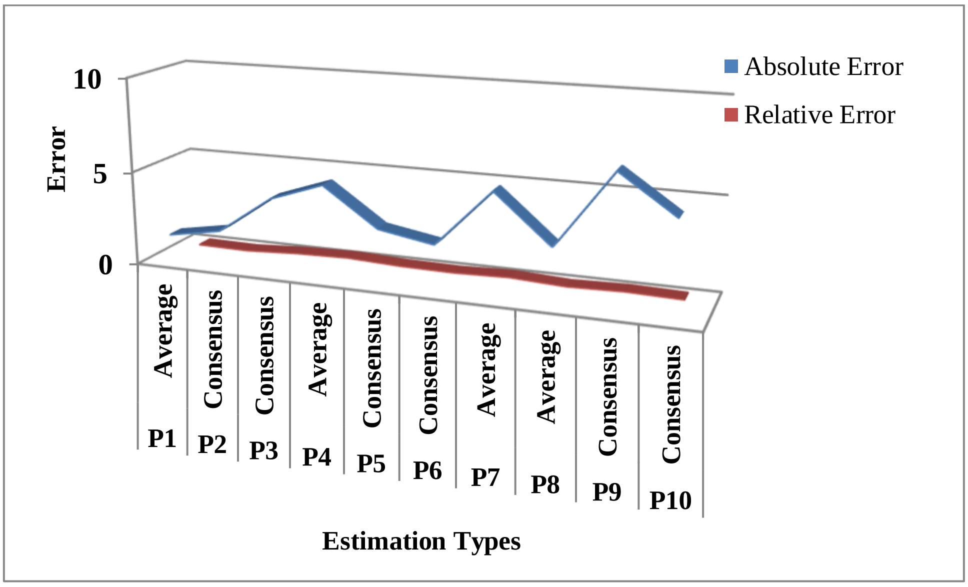

Figure 11b: Absolute and Relative Error of Different Estimation Types

Figure 11c: Variance Comparison at Various Fibonacci Levels

The above graph 11 a) and b) is a graphical representation of the estimation made for variance due to variation in the Fibonacci level. The estimation depends on Estimation cost size of the project, and the actual cost that has been derived from various methods such as average or consensus. Then we plotted the actual and relative error between estimated and actual cost. The scope of the undertaking and the team's level of experience are two other elements that affect project pricing.

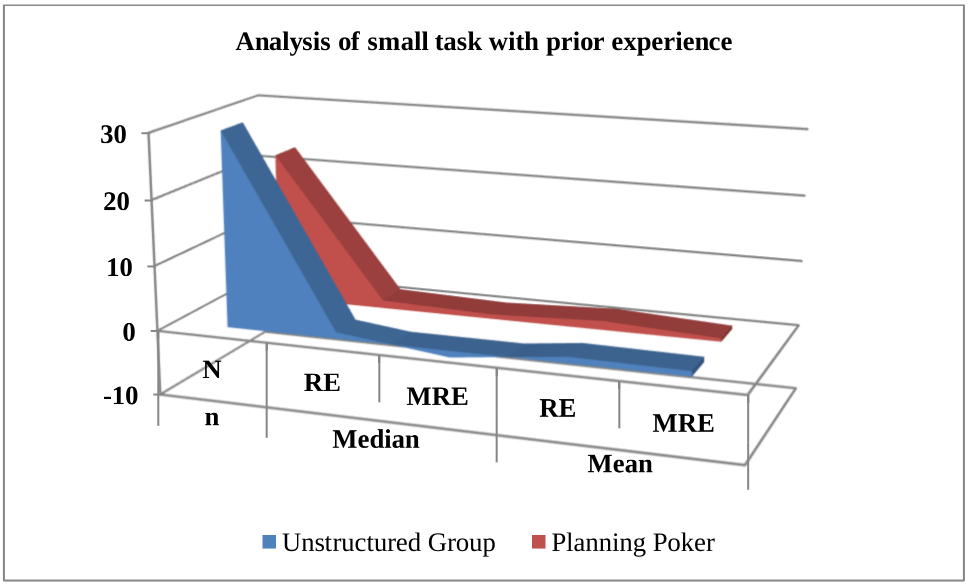

Figure 12: Small Task with Prior Experience - Comparative Analysis

The above graph 12 highlights a comparative analysis of mean and median relative errors (RE) and mean relative magnitude errors (MRE) in two distinct scenarios: Unstructured Group and Planning Poker, the comparison of mean and median when the size of the task is small (in the unstructured group the size is 30 and in planning poker the size is 21) and the team has prior experience then we can see the median magnitude relative error as 0.42 and 0.25 with unstructured group and planning poker methodology respectively. The average relative magnitude error is 0.39.

and 0.50 for unstructured group and planning poker methodology respectively. The graph demonstrates that in this situation, the strategy for and method of estimation performs better.

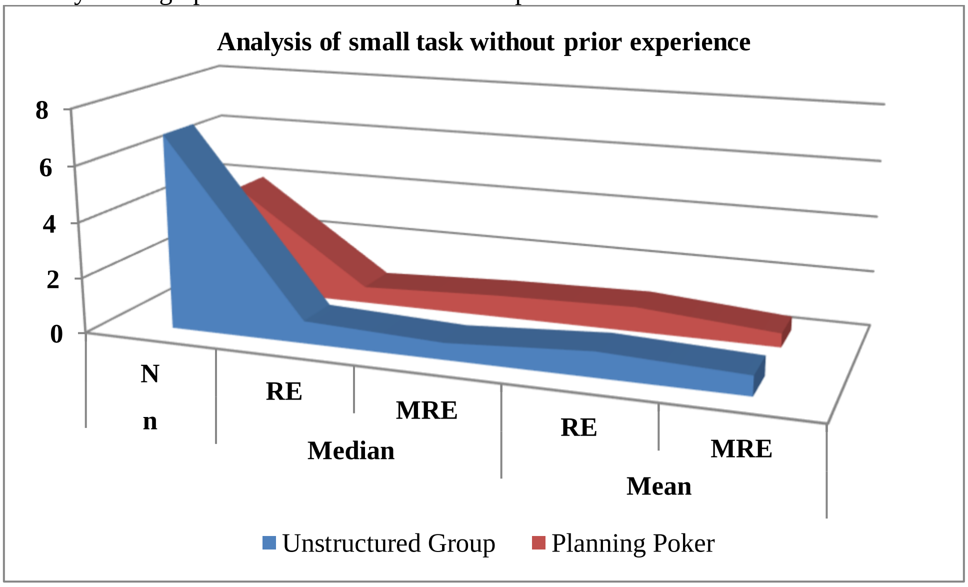

Figure 13: Comparative Analysis of small task without prior experiences

The above graph 13 shows the comparison of mean and median when the size of the task is small (in the unstructured group the size is 4 and in planning poker, the size is 7) In the case of the Unstructured Group, where the task size is 7 and the team possesses prior experience, the calculated median relative error (MRE) and mean relative error (RE) are found to be 0.60 and 0.80, respectively. Similarly, for the same group and task size, the MRE and RE are observed as 0.73 and 0.91, respectively. Conversely, in the Planning Poker scenario with a task size of 4 and prior team experience, the MRE is 0.83, and the RE is 0.61. Similarly, for the same Planning Poker scenario but with the smaller task size of 41, the MRE is 0.53, and the RE remains at 0.94. A comprehensive evaluation of the average relative magnitude errors (MRE) for the two scenarios reveals that the Unstructured Group exhibits an MRE of 0.7, while the Planning Poker approach demonstrates a slightly superior MRE of 0.58. Moreover, examining the average relative errors (RE) further underscores the advantage of Planning Poker, as it boasts an RE of 0.65 compared to the Unstructured Group's RE of 0.75. Ultimately, this analysis underscores that, under conditions of smaller task sizes and previous team experience, the Planning Poker methodology offers enhanced accuracy in estimation, making it a more effective approach than the Unstructured Group method.

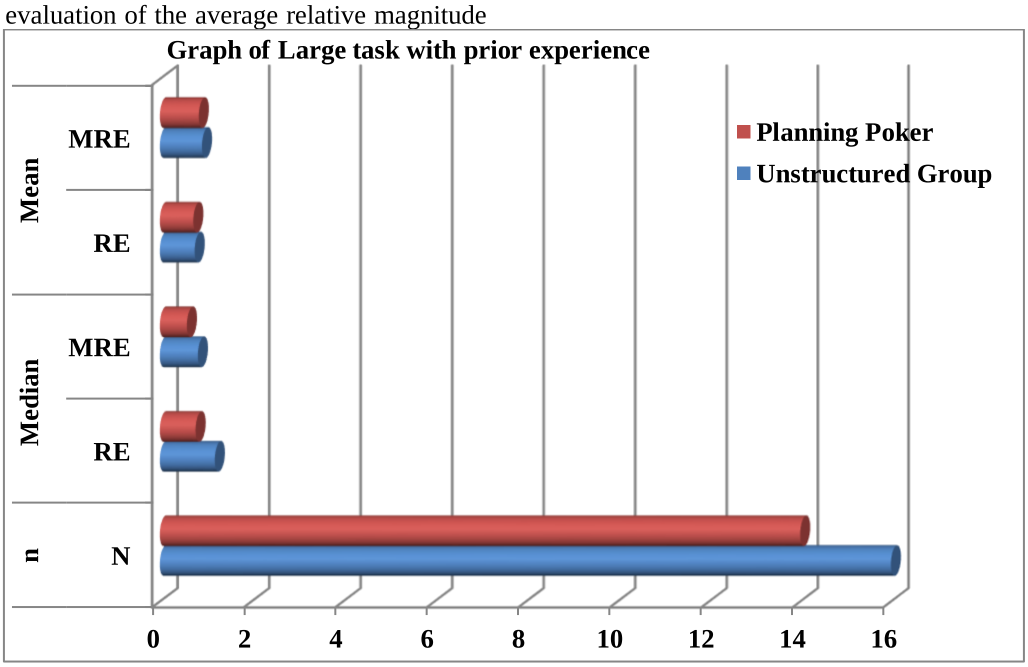

Figure 14: Comparison between Unstructured Group, and Planning Poker based on Large task with prior experiences

The above graph 14 shows the comparison of mean and median of unstructured group and planning poker. This analysis is conducted for larger task sizes – specifically, a size of 14 for the Unstructured Group and a size of 16 for the Planning Poker approach – while considering prior team experience. In the case of the Unstructured Group with a task size of 30 and previous experience, the median relative error (MRE) is recorded as -1, indicating a slight underestimation, and the mean relative error (RE) is 1. The corresponding MRE and RE for the Planning Poker approach with the same task size are 0.7 and 1.2, respectively. Similarly, for a smaller task size of 23 and prior team experience in the Planning Poker scenario, the MRE is found to be 0.5, while the RE is 1.4. On the other hand, the Unstructured Group exhibits an MRE of 0.8 and an RE of 1 for the same conditions. Additionally, examining the average relative magnitude errors (MRE) for the two approaches reveals that the Unstructured Group has an MRE of 0.23, whereas the Planning Poker method boasts a slightly improved MRE of 0.21. Ultimately, this comparison demonstrates that, in the context of larger task sizes and with prior team experience, the Planning Poker methodology proves to be more accurate in estimation, showcasing its superior performance over the Unstructured Group method.

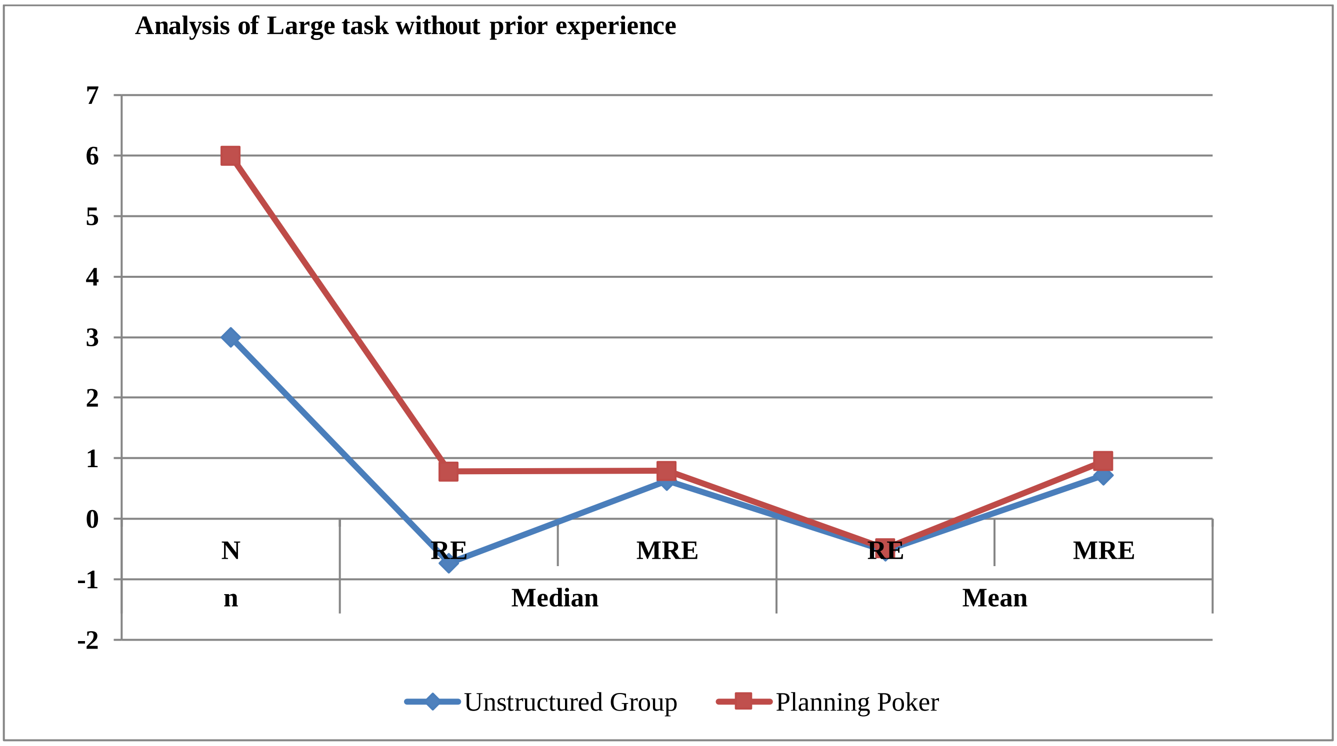

Figure 15: Comparison between Unstructured Group, and Planning Poker based on large task without prior experiences

The above graph 15 shows the comparison of mean and median when the size of the task is large (in the unstructured group the size is 3 and in planning poker, the size is 6) and the team has prior experience then we can see the median magnitude relative error as 0.40 and 0.58 with unstructured group and planning poker methodology respectively. The mean magnitude relative error is 0.30 and 0.87 for unstructured group and planning poker methodology respectively. The graph shows that the unstructured estimation process works better in this case

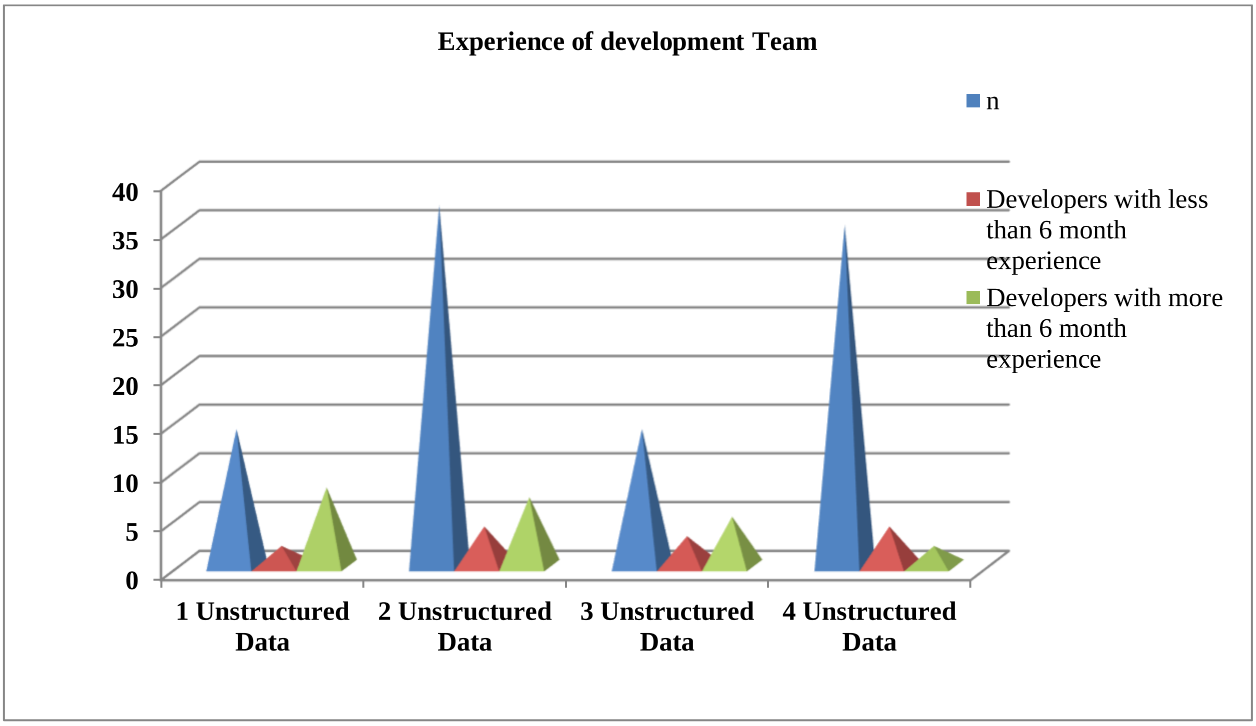

Figure 16: Team Experience Comparison Analysis

As we have discussed earlier team experience matters when we develop a software project. The above graph 15 is a comparison between two teams one having experience of (0-6 months) and the other having an experience of more than 6 months. Each instance highlights the count of developers falling within these experience brackets who participated in tasks related to unstructured data. The estimation process used is an unstructured group with several projects 14 and 37 and planning poker with several projects 12 and 38. Additionally, two developers with more than six months of experience were also part of this endeavor. This pattern extends across the subsequent data, providing a snapshot of the distribution of developer experience levels for different instances involving unstructured data in software development.

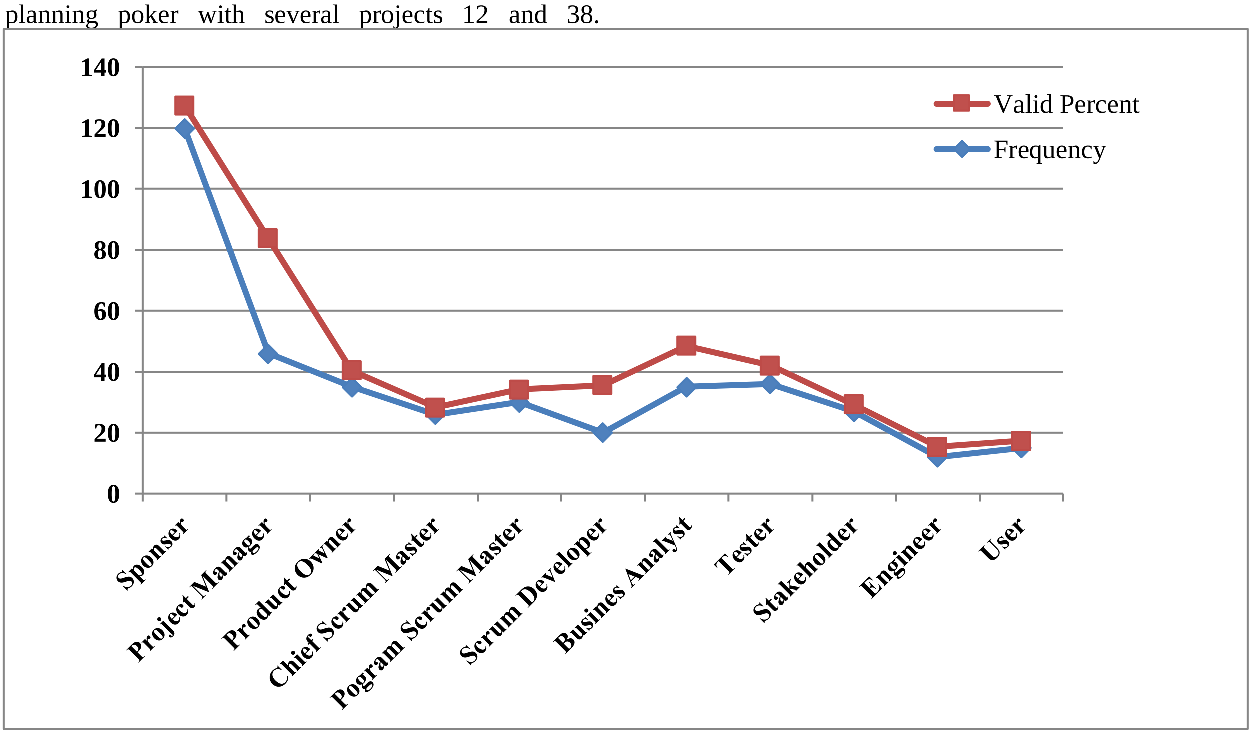

Figure 17: Analysis of IT Professional Respondent Profile

The above graph 17 is distribution of roles and corresponding frequencies among a sample of 77 IT professionals who participated in a study. The study aimed to understand the roles and responsibilities of these professionals in software projects. Each role's frequency and valid percentage are provided to offer insights into the composition of the participant group. Among the participants, the role with the highest frequency was Project Manager with 46 respondents, accounting for approximately 37.7% of the total participants. Following closely, Scrum Developer represented 15.6% of the participants with 20 individuals. Business Analyst and Tester roles accounted for 13.5% (35 respondents) and 6% (36 respondents), respectively. Roles like Product Owner, Chief Scrum Master, and Program Scrum Master were also present, comprising 5.4%, 2.3%, and 4.2% of the participants, respectively. Additionally, roles such as Sponsor, Stakeholder, Engineer, and User had relatively lower representation, each accounting for 7.4%, 2.2%, 3.4%, and 2.3%, respectively. These percentages provide a clear picture of the various roles held by the IT professionals in the study, offering insights into their distribution and highlighting the prevalence of certain roles, such as Project Manager and Scrum Developer, within the surveyed group and also their frequency of handling the software project has been represented in the above graphs.

Furthermore, our analysis of the impact of the modified Fibonacci sequence on variance, estimation accuracy, and the efficiency of cost estimation within agile projects holds significant implications. Project managers and teams can leverage these insights to make more informed decisions, thereby enhancing project planning, optimizing resource allocation, and ultimately contributing to overall project success. This approach to estimation, tailored to the unique characteristics of individual projects, nurtures a more adaptable and effective estimation process. By aligning estimations with project intricacies, practitioners can refine accuracy and responsiveness, ushering in a more agile and efficient estimation paradigm. In essence, our study not only illuminates the intricate relationship between the modified Fibonacci sequence and estimation outcomes but also furnishes a practical roadmap for its application. This empowerment equips stakeholders to harness these insights, propelling improved project outcomes, resource utilization, and the overarching achievement of project objectives.

5. Conclusion and Future Work

In summary, our study provides compelling evidence in favor of the Planning Poker estimation technique, highlighting its superior performance. Furthermore, our investigation demonstrates that the incorporation of the modified Fibonacci sequence yields a significant enhancement in the Planning Poker methodology's effectiveness. Through a comprehensive analysis presented in the results section, we calculated variances and determined actual and relative errors across various instances, juxtaposing these outcomes with existing data. The notably lower relative error substantiates the efficacy of our proposed algorithm, underpinning its successful implementation. Notably, our dedicated efforts in algorithm execution have revealed the potential of this agile cost estimation technique to expedite project workflows. By minimizing estimation discrepancies, this technique holds promise in streamlining project execution. In future endeavors, we intend to synergize the Planning Poker method with complementary techniques, such as affinity grouping. This integration aims to achieve more efficient forecasting of budgets and schedules, harnessing the strengths of both approaches to drive heightened accuracy and productivity.

Declaration:

Ethics Approval and Consent to Participate:

No participation of humans takes place in this implementation process

Human and Animal Rights:

No violation of Human and Animal Rights is involved.

Funding:

No funding is involved in this work.

Data availability statement:

Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study

Conflict of Interest:

Conflict of Interest is not applicable in this work.

Authorship contributions:

All authors contributed equally to this work

Acknowledgment:

There is no acknowledgment involved in this work.

References

[1] S. Kolesnikov, N. Siegmund, C. Kästner, A. Grebhahn, and S. Apel, 2019. “Tradeoffs in modeling performance of highly configurable software systems”, *Software & Systems Modeling*, *18*(3), pp.2265-2283.

[2] R. Burga, C. Spraakman, C.Balestreri, and D.Rezania, 2022, “Examining the transition to agile practices with information technology projects: Agile teams and their experience of accountability”, International Journal of Project Management, 40(1), pp.76-87.

[3] M.E. Helander, M. Zhao, and N. Ohlsson, 1998, “Planning models for software reliability and cost”, IEEE Transactions on Software Engineering, *24*(6), pp.420-434.

[4]C.W.Butler,L.R.Vijayasarathy, and N.Roberts, 2020, “Managing software development projects for success: Aligning plan-and agility-based approaches to project complexity and project dynamism”, Project Management Journal, 51(3), pp.262-277.

[5]M.Hamid, F.Zeshan, A.Ahmad, F.Ahmad, M.A.Hamza, Z.A.Khan, S. Munawar, and H.Aljuaid, 2020, “An intelligent recommender and decision support system (IRDSS) for effective management of software projects”, IEEE Access, 8, pp.140752-140766.

[6] Amjad, S., N. Ahmad, T. Saba, A. Anjum, U.Manzoor, M.A. Balubaid, and S.U.R. Malik, 2017, “Calculating completeness of agile scope in scaled agile development, IEEE Access, *6*, pp.5822-5847.

[7] T.J. Gandomani, Z. Tavakoli, H. Zulzalil, and H.K. Farsani, 2020, “The Role of Project Manager in Agile Software Teams: A Systematic Literature Review”, IEEE Access, *8*, pp.117109-117121.

[8] P. Letelier, and M.C. Penades, 2017, “Agileroadmap: An approach to implement agile practices in teams, IEEE Latin America Transactions, *15*(7), pp.1295-1300.

[9]M.Shepperd, and C. Schofield, 1997. “Estimating software project effort using analogies,” IEEE Transactions on software engineering, *23*(11), pp.736-743.

[10] A.A. Sabahat, Malik, and F. Azam, 2017. “A size estimation model for board-based desktop games, “ IEEE Access, *5*, pp.4980-4990.

[11] M.El Bajta, A.Idri, J.N. Ros, J.L Fernández-Alemán, J.M.C de Gea, F. García, and A. Toval, 2018, “Software project management approaches for global software development: a systematic mapping study,” Tsinghua Science and Technology, *23*(6), pp.690-714.

[12] Ciampa, Pier Davide, and Björn Nagel, "AGILE Paradigm: the next generation collaborative MDO for the development of aeronautical systems," *Progress in Aerospace Sciences* 119 (2020): 100643.

[13] A.Zaitsev, U. Gal, and B. Tan, 2020, “Coordination artifacts in Agile Software Development,” Informationand Organization, 30(2), p.100288.

[14] C.Tam, Moura da Costa, E.J. Oliveira, and J. Varajão, 2020, “The factors influencing the success of ongoing agile software development projects, International Journal of Project Management, 3*8*(3), pp.165-176.

[15] M.Fernández-Diego, E.R.Méndez, F.González-Ladrón-De-Guevara, S. Abrahão, and E.Insfran, 2020, “An update on effort estimation in agile software development: A systematic literature review”, IEEE Access, 8, pp.166768-166800.

[16] M. Grossi, "Energy Harvesting Strategies for Wireless Sensor Networks and Mobile Devices: A Review", *Electronics.*, vol. 10, no. 6, pp. 661, 2021.

[17] P. Livreri, "A 3.3 V Output Voltage Optical Plasmonic Solar Energy Harvester", *2021 10th International Conference on Renewable Energy Research and Application (ICRERA)*, pp. 310-31, 2021.

[18] Ramakrishna Nuvvula, Elangovan Devaraj & Kishore Teegala Srinivasa (2021): Optimal Sizing of BESS- WtE Supported Hybrid Renewable Energy Sources to Improve Sustainability of Smart Cities with Ramp-Rate Limitations on Thermal Power Stations.. International Transactions on Electrical Energy Systems. DOI: 10.1002/2050-7038.13241

[19] T. S. L. V. Ayyarao Ramakrishna S.S. Nuvvula Baseem Khan 'Extended State Observer-based Primary Load Frequency Controller for Power Systems with Ultra-High Wind-Energy Penetration Journal of Electrical Engineering & Technology – Springer -2022 .

[20] M. Colak, I. Cetinbas and M. Demirtas, "Fuzzy Logic and Artificial Neural Network Based Grid-Interactive Systems for Renewable Energy Sources: A Review", *2021 9th International Conference on Smart Grid (icSmartGrid)*, pp. 186-191, 2021.