An Empirical Investigation of Effort Estimation in Mobile Apps Using Agile Development Process

Abdullah Altaleb[1,2*], Andrew Gravell[1]

[1] School of Electronics and Computer Science, Faculty of Engineering and Physical Sciences, the University of Southampton, Southampton, United Kingdom.
[2] College of Computer and Information Sciences, Imam Mohammad Ibn Saud Is. University, Riyadh, Kingdom of Saudi Arabia.

* Corresponding author. Tel.: +44-7533287832; email: a.altaleb@soton.ac.uk

Manuscript submitted May 4, 2019; accepted July 10, 2019.
doi: 10.17706/jsw.14.8.356-369

Abstract: Effort estimation is essential in order for a project manager and development team members to be able to successfully plan for a software project. The planning and development of mobile applications present many challenges. The aim of this study is to provide and report an overview on the state of the practice of effort estimation techniques that companies use for their mobile app projects. This study focuses on organisations which apply the Agile development process during their projects. We conducted structured and semi-structured interviews with 20 Agile practitioners at 18 different organisations. The results revealed that Planning Poker (PP) and Expert Judgment (EJ) were the most frequently used estimation techniques in mobile app projects.

Key words: Agile, effort estimation, mobile application, planning poker, expert judgment, software development.

1. Introduction

Effort estimation in a software project plays an important role in measuring the expected amount of effort and required resources, and in helping to deliver the project on time. An inaccurate estimation will lead to project failure. A good estimation is not only used to determine an appropriate deadline and schedule, but also helps to reduce risk, uncertainty and support better decisions by raising detailed questions related to the dark parts of the project during the estimation phase [1].

With regard to the term Agile Software Development (ASD), a systematic literature review was aggregated by [2] to describe the state-of-the-art related to estimation techniques. The study found that certain estimation techniques are the most frequently applied in the ASD context, namely: expert-judgment, Planning Poker and use case points. Planning poker is a consensus-based technique for estimation in Agile, and was defined by [3]. This technique has been empirically validated in different forms by [4], [5].

Mobile app development is different from other traditional software due to many special constraints, requirements and other challenges [6]. One of these challenges is the platform multiplity issue, whereby developers are always concerned about the compatibility of their app with all mobile operating systems. Other challenges which have been discussed include the wireless communication issue, portability, performance limitation, responding time, multiple different standards, protocols and network technologies, and strict time-to-market requirements [7], [8].

Anureet and Kulwant presented a systematic literature review concerning effort estimation in mobile application development and testing [9]. The result revealed 15 characteristics from 27 studies that clearly stated how mobile apps development is different from traditional software. One of these characteristics is the limited screen size and user experience UX. The developers are concerned about the limitation of the screen size and how it looks on different mobile devices. Other characteristics which make mobile apps different from other software are as follows: the fact that some apps run in integration with other apps, the diversity of the user interface (keypad, touchscreen, voice, etc.), dealing with interruptions (such as call, messages, notifications, etc.) and limited memory RAM and power [10]–[12]. Other studies provided additional challenges in regards mobile app development which make it different from other traditional software, such as dealing with mobile API, sensors, GPS, camera, voice, etc. [13]. Another systematic literature review was conducted for the effort estimation techniques in mobile apps [14]; as a result, no previous studies were found regarding the state-of-practice to investigate and validate the suitability of existing estimation techniques which are used in the Agile process, such as Planning Poker, for mobile app development.

From the SLR [9], [14], it can be seen that there are many studies which have investigated and validated the COSMIC Function Size Measurement estimation technique in mobile apps [15]–[18]. Moreover, Analogy-based and Regression-based estimation models have been investigated in mobile app development [19]–[21]. However, the previous SLR determine that there is no empirical study which has investigated, validated or examined the usage of Agile estimation models in mobile app development.

Therefore, the authors of the present paper conducted an empirical study that was based on 20 practitioners from multiple different industrial fields to investigate the Agile estimation techniques for mobile app development. The aim of this study was to investigate the estimation techniques and metrics (as well as their accuracy levels) which are used during mobile app development. The present study focused on mobile applications which follow the Agile development process.

The paper is organised as follows: Section 2 presents the related work, while in Section 3 we will describe the research methodology; the results of the survey will be presented in Section 4, following which the analysis of the collected data and results discussion will be put forth in Section 5. Subsequently, the future work of this study and conclusion can be found in Section 6.

2. Related Work

There exists a systematic literature review [14] which has provided an overview of the current state-of-the-art for effort estimation techniques in mobile app development which use the Agile process. 21 papers were found among this literature; 13 of the 21 studies concerned the estimation size metrics and techniques in mobile app development, whereas 8 studies concerned the applicability of the Agile process in the development of mobile apps. The literature revealed that Expert Judgment and regression-based techniques have been the most used tools in past studies, and the function point, such as COSMIC FP, was the most used size metric.

Another study was conducted by [22], and presented a report of an online questionnaire from 60 practitioners on effort estimation techniques in the Agile development process. The result revealed that 63% of the participants used the Planning Poker technique, while 47% used analogy and 38% used Expert Judgment in the practice of estimation in ASD.

[23] put forth an important debate between Magne Jørgensen and Barry Boehm regarding the best approach with which to estimate the effort needed for software projects. The paper found that the formal models were not very frequently used compared with the Expert Judgment. 10 out of 16 studies showed that the effort estimation technique based on Expert Judgment was more accurate than formal models. This debate listed several reasons why the Expert Judgment technique is more accurate and used more frequently

than the formal models. Formal models, such as COCOMO, rely on certain essential inputs, e.g. line of code of function point, which are based on the Expert Judgment technique. As a result, the accuracy of the efforts at the end are also based on the Expert Judgment technique.

3. Research Methodology

3.1. Survey Technique

In this study, we applied the interview technique, and more specifically two types of interview: standardised (questionnaires) and non-standardised (semi-structured). The questions for the former were designed in a way as to enable us to identify and describe the variable of the organisations' practice. In contrast, the questions for the latter (semi-structured) were designed to address the previous questions which needed more clarification and explanation. Ethics procedures were implemented in this research, and the University of Southampton Ethics Committee approved the research method and strategy for the study.

3.2. Interview Approach and Sample type

In this interview, we used the structured and non-structured method within one interview, which is called the mixed-method, and helps to validate and verify the participants' responses to the questions. In an interview, there are several concerns which must be taken into account during the process [24]:

1) Using an appropriate language which interviewees feel more comfortable with.
2) Considering the approach used to ask a question, and allowing the practitioner to talk freely throughout the in-depth discussions.
3) Listening clearly and patiently.
4) Recording data in different forms:
   a) Written; for example: fill-in the form and checklist.
   b) Audio-record.

The sample group members were not selected randomly, with the choice of the sample instead left to the researcher. The practitioner should satisfy the inclusion criteria to make sure that the data we collected were retrieved from the appropriate resources. The criteria were as follows:

- The practitioner should have experience in effort estimation for mobile projects.
- The practitioner should have practised or been involved in the Agile process for any software project.
- The practitioner could be a software developer, mobile project manager, code tester, code reviewer, code quality assurance or software designer for a mobile app.

3.3. Interview Questions

There are four questions which were investigated in this survey:

- RQ1: Which mobile platform type and name are used during your development?
- RQ2: Which effort estimation techniques and measurement types are used in your mobile app development?
- RQ3: How did you use the estimation technique in your organisation?
- RQ4: How accurate are the effort estimates in your organisation?

4. Results

This section presents the results of the interviews. The following sections are organised according to the research question.

4.1. Practitioners' Demographics

During the interviews, certain demographic questions were put to the practitioners, namely:

- Work experience countries.
- Years of work experience.
- Current and previous positions/role in your employment.
- Nature of organisation you work with (Telecom/IT Co./etc.).

The practitioners answered all of the questions above; detailed answers will be presented in the following sections. A total of 20 practitioners were interviewed for this study, 17 of whom agreed to be audio-recorded, while 3 practitioners refused to partake in the audio recording, with notes instead taken.

4.1.1. Work country

As shown in Fig. 1, all 20 practitioners worked in Saudi Arabia, while 3 of them also worked in the US, and 2 experts worked in Bahrain, with only 1 in India. We can observe that many of the practitioners worked in multiple countries.

4.1.2. Job roles

Most of the practitioners worked in multiple roles while building their experience. The interviews revealed that software developer was the most frequently practised job role during the work history of 12 practitioners, followed by project manager, quality assurance (QA), freelancer, sales and IT director with 5, 5, 4, 2 and 1 practitioners respectively. Fig. 2 shows the experts' job roles.

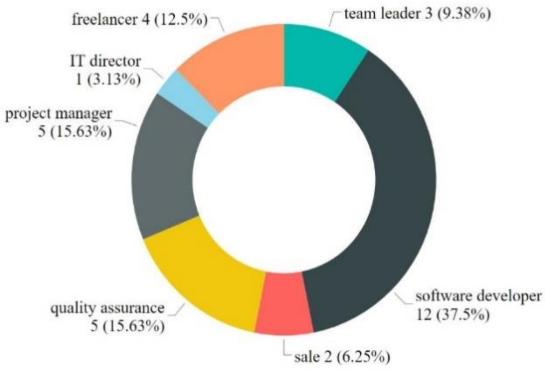

Fig. 1. Practitioners' job roles.

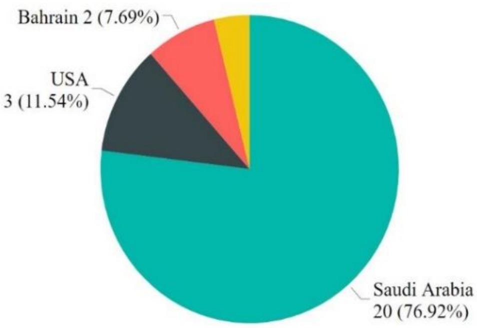

Fig. 2. Practitioners by country.

4.1.3. Years of experience

The work experience years for all of the experts ranged from 2 to 9 years. The work experience mean of the experts was 5.70 years, and the standard deviation of the experience years was 1.867. As we can see from Figure 3, most of the software developers' experience ranged from 4 to 6 years. There was only one IT director and he worked for roughly 4 years.

s, which represent 40% of all company types.

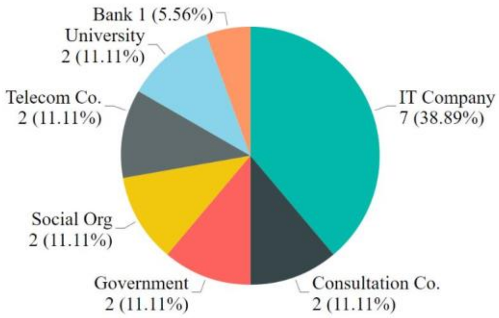

Fig. 3. Company type.

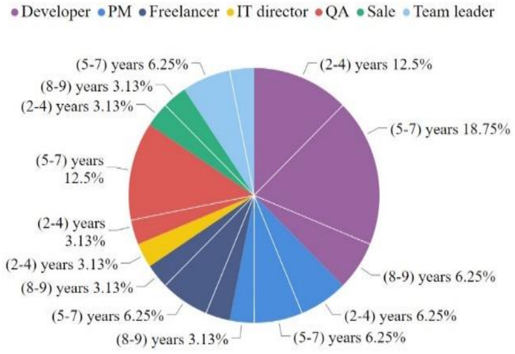

Fig. 4. Years of work experience by job role.

4.1.4. Company type

The practitioners worked at 18 organisations, as can be seen in Fig. 4, which is categorised into 7 groups: Telecommunication companies, IT Solutions companies, Banking, University, Social & Insurance companies, Consultation companies and Government. The majority of the practitioners had worked in seven IT Solutions companie

4.2. RQ1: Mobile App Platform

There are three types of mobile application platform: native, hybrid and mobile web. The native app is developed for a specific platform which uses native APIs. The hybrid app is developed by web technologies, such as HTML, and wrapped in a native container. In contrast, the mobile web app is developed by web technologies and used in a web browser.

As we can observe from Table 1 and Table 2, some practitioners have used multiple frameworks during mobile app development, such as Android Studio and Swift. Most of the practitioners were using a hybrid platform during their development, whereas the mobile web was a less frequently used platform. Since the most frequently used platform type was hybrid, the Ionic was the most used framework for mobile app development by around 50% of the experts, as seen in Fig. 5. For the native platform, the Android Studio framework was also preferable for the practitioners, as well as when it came to developing a native mobile app. The practitioners, who preferred to use the hybrid platform more than the native platform, claimed that to avoid developing two apps for two different platforms. The Android Studio and Ionic frameworks are leads between the other mobile app frameworks.

Table 1. Platform Type Used by Practitioners

		Responses	
		No.	**Percentage**
Platform Type	Hybrid	12	50.0%
	Native	10	41.7%
	Web	2	8.3%
Total		24	100.0%

Table 2. Framework Name Used by Practitioners

		Responses	
		No.	**Percentage**
Platfor m Name	Ionic	10	38.5%
	Android Studio	7	26.9%
	Swift	5	19.2%
	React-native	2	7.7%
	HTML5	2	7.7%
Total		26	100.0%

4.3. RQ2: Effort Estimation Techniques

From the practitioners' answers it can be concluded that only three types of effort estimation techniques were used in mobile app development: Expert Judgment (EJ), Planning Poker (PP) and COCOMO II. Some of the practitioners used a combination of estimation techniques, and we will discuss this in more detail in Section 4.4. More than half of the practitioners stated that they use the Expert Judgment technique to estimate the effort, with around 40% of them using planning poker, as shown in Figure 6. Only one practitioner admitted to using COCOMO to estimate the effort, and he worked as a project manager. He used this technique to have an idea about the required effort prior to the project starting.

Regarding the measurement type of the effort, as we can see from Table 3, when the practitioners employ the Expert Judgment technique, they usually prefer to measure the effort of the user story by ideal hour. In contrast, with regard to the practitioners who tend to use the planning poker technique, all of them prefer to use story point. There is only one practitioner who admitted to using the COCOMO technique, and he used the function point as an input for this method.

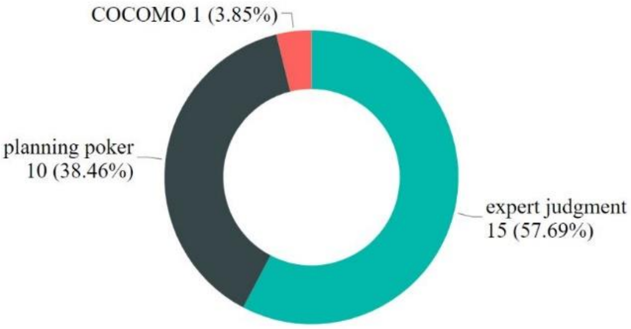

Fig. 5. Estimation techniques used in mobile app development.

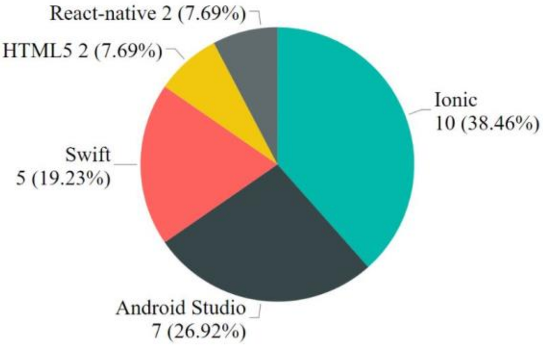

Fig. 6. Name of platforms used by practitioners

Table 3. Measurement Type in Estimation Technique

		Measurement type		Total
		Ideal hour	**Story point**	
Estimation technique	Planning Poker (PP)	0	10	10
	Expert Judgment (EJ)	13	2	15
Total		13	12	25

4.4. RQ3: Estimation Process and Accuracy

RQ3 is considered a semi-structured question. First, we will put forth a summary of the practitioners' answers. Some of the responses shared the same perspective, and therefore we will report some of the practitioners' opinions. Following this, in Section 5, we will analyse the respondents' answers using NVIVO software to integrate the related data from said respondents' answers, and then identify key themes from the data and design a matrix. In the following, we will present a summary of the 20 practitioners' experience related to their estimation process (Practitioners ID: A_T), and categorise their answers based on their main job role.

4.4.1. Project managers' answers

Practitioner A (P_A): the product owner explains a user story, in terms of features, to all members of the development team, and then divides the tasks into subtasks: front-end and back-end. P_A's company employs a mix of estimation techniques, namely PP and EJ. For the initial estimation, the senior developer gives an initial estimation for the user stories using the EJ technique, before applying Planning Poker with all development team members. As P_A worked as IT director, he employed a mixed technique to avoid extremely high estimations from the team. Following this, all development team members, including the scrum master, will be asked to participate and give an effort estimation for all of the user stories. If someone takes this feature and his/her estimation number is close to that of the other development team, the group will take this number, otherwise the team leader will become involved and will ask the developer for more clarification until both of them agree on a specific estimation number. Based on P_A's experience, developers usually assign a high estimation value in a user story to avoid work pressure, and that is why P_A uses a combination method for assigning effort for a task.

Practitioner I (P_I): P_I works as a project manager, and he usually relies on experts from the development team to assign an expected effort value. Usually, P_I asks a senior developer about his expectation for a project.

Practitioners J and R (P_J, P_R): P_J and P_R have similar opinions about their estimation techniques and they usually use Planning Poker to estimate effort for their project. The reason that they use Planning Poker is because they want to hear feedback from each member of the project, and then take an average value for each member regarding each task. P_J and P_R avoid doing second rounds in PP to save time, unless the estimation range values between the team members are high. P_J stated that the risk will be very high when he relies on one person from the team to estimate the effort, since the person, e.g. the developer, could leave the project for any reason, and all of P_J's estimations may have been based on the developer's opinion. P_J and P_R only use the Expert Judgment technique when they are familiar with a project and need quick feedback from the development team. P_J uses COCOMO as an initial estimation for a project, but it is not an accurate technique since it is difficult to measure the complexity of a project, and assign a value for the scaling and cost drivers/factors, especially the capability of the development team factors.

Practitioner K (P_K): P_K asks each developer to assign an estimated effort value for his/her task, and then asks some of the team members about their estimated value. If the difference in value between the team members and the developer is not high, P_K confirms the estimated effort for the developer, otherwise P_K discusses, with the developer, his estimation in order to re-estimate his/her effort and make it more reasonable.

4.4.2. Mobile developers answers

Practitioner B (P_B): P_B stated that the product owner of the project explains the product backlog items to all development team members and answers any questions from the team. Following this, the development team discuss what kind of tasks they could be doing during the sprint, based on their velocity from the previous sprints. P_B told of how, as a whole team, they assign a story point for each user story in one day. They start with a small task to make a base for the story point, and then compare the other users' stories with a base one. They use the Fibonacci sequence, and when a user story has 64 points, it is necessary to divide this story into two or three subtasks as required. P_B stated that they use Planning Poker to simplify the business requirements and make said requirements "user stories" which are clear for all the team members and open a good discussion for the team members.

Practitioner C (P_C): talked about how whomever carries out the task should be responsible for assigning the effort value for his/her task. The problem arises when a manager assigns an estimation effort for a task or allows a back-end developer to assign the effort for a front-end task. As a result, the estimation value will not be accurate, and it could lead to failure of the task. Moreover, in order to estimate the effort, and as stated by P_C, you have to be the person doing the task and understand the big picture of the project.

Practitioner D (P_D): P_D stated that, as a senior developer, he assigns the estimated effort for a task by himself, and lets the team leader know about it. He also stated that, based on his experience, junior developers estimate their effort based on fears and unexpected issues that could happen during their development. For this case, the team leader is involved with the junior developers and makes the task clearer for them while also explaining what kind of tools they need until they reach a fair estimated effort value. Sometimes a senior developer may assign a high estimated value for a task that he/she has not encountered before. For the measurement type of the effort value, P_D does not believe in assigning effort by story point because it does not help him to measure the actual effort. He prefers to use ideal hour as a meaningful measurement for the effort. He stated that when he needs to design a user interface for a mobile app, he knows how long it could take him, e.g. 1 day or 2-3 days, but he is unable to say how many story points he needs to design.

Practitioners E and T (P_E, P_T): both practitioners agreed that, at the beginning, each developer assigns his/her estimated effort value for his/her task. Following this, the developer discusses his/her estimated value with his/her team leader. After that, the developer and team leader come up with a final estimated value following their discussion. Usually, the team leader agrees with what the developer came up with, unless the developer assigns a very high estimated value. No one interferes with the estimated value except for the developer who will complete the task, and his/her team leader.

Practitioner F (P_F): this practitioner stated that everyone on the development team will be assigned to a task, and then all members give their effort estimated value for their task. Following this, each member explains his/her task and effort value to the team members, and then the whole team discuss his/her estimation by applying the Planning Poker technique. The team members will accept the estimated value if the value is close to the effort values from planning poker, otherwise the team ask the developer to re-estimate his/her task.

Practitioner G (P_G): P_G stated that if a developer completes the task alone, then he/she assigns his/her effort value for his/her task and then discusses the value with his/her team leader. P_G also stated that sometimes they have a task that requires group work, and then they apply the Planning Poker technique by having a long discussion with all of the team members (developer, tester, designer, etc.), following which they assign an effort value at the end of the discussion. P_G also claimed that a developer prefers to be assigned to a specific task and to work alone, as in this way there is no need to relate with other team members, which would require more communication and could delay the delivery of a user story.

Practitioners H and M (P_H, P_M): P_H and P_M stated that all the team should participate in assigning the effort value for each task by applying the planning poker technique. P_H stated that they usually take the average of the estimated effort to avoid the second round of planning poker, unless the estimated values between the team members are very large. P_M tells of how the company tries to avoid individual estimation, e.g. Expert Judgment, for a user story and encourages the team to discuss all the tasks together to make a user story clearer.

Practitioner L (P_L): P_L declared that the team leader for a project explains the user story to all team members, and each of the team members participates by giving his/her opinion regarding each task; at the end of the discussion, the whole team assign an effort value for each task. As stated, there is no specific technique which they follow, although the team leader interferes if there is a conflict on a task and gives an estimation value for this task.

4.4.3. Quality assurance team answers

Practitioners N and O (P_N, P_O): P_N and P_O are part of the quality assurance team, and they apply the same method when a task comes to them: they assign an effort estimation value for the task, and let the team leader know how long it could take them. There is no specific method which they use, and they consider their approach to be the Expert Judgment technique.

Practitioner P (P_P): P_P stated that, in his company, they apply two different techniques. The first technique involves the product owner explaining the details of the user story to all development team members on the project. Inside the development team, there are two different groups: front-end and back-end groups. One senior expert from each group assigns an effort estimation (story point) for the user story.

P_P declared that the second estimation technique they use is traditional planning poker, whereby every member gives his/her opinion until a consensus value is reached. Based on these two experiences, P_P concluded that the first technique was more accurate than the second one. The main reason, according to him, is that when they applied the planning poker, the developers' experience levels varied, and the team was new.

Practitioners Q and S (P_Q, P_S): both practitioners mentioned the same experience, namely that there is no specific method which they follow in their company. The task is assigned to them by the team leader with a given estimated effort. If the practitioners reach the deadline and have not finished the task yet, a justification for the delay should be given to the team leader.

4.5. RQ4: Estimation Accuracy

The aim of this question is to identify the respondents' beliefs regarding the accuracy of the effort estimation process and technique in their organisations. From Fig. 7, we can observe that 75% of practitioners tend to underestimate the effort values of a user story rather than overestimating. Around 40% of the respondents stated that they underestimate their effort value by 25% to 45% compared with the actual effort of a user story. The respondents' results show that it is rare to overestimate the effort value by more than 25% of the actual effort.

A discussion was opened with the practitioners regarding the reasons for inaccurate estimation related to a user story. One of the main reasons for high error in accuracy was the effort estimation value being assigned by a team leader or project manager without any discussion with the developer who will do the task. Another reason was when the Expert Judgment or Planning Poker technique is used alone. A combination of these techniques will reduce the error range. Another reason was stated, namely when the estimated value is given by all the team members using the PP technique, and during the sprint the task is assigned to one developer. The developer could be a junior, or the estimated value may be given based on a similar task completed previously by an expert developer. Some practitioners claimed that there is no clear guidance for giving an estimation value. EJ and PP suffer because of formality during the estimation process. More reasons are mentioned in the estimation process section.

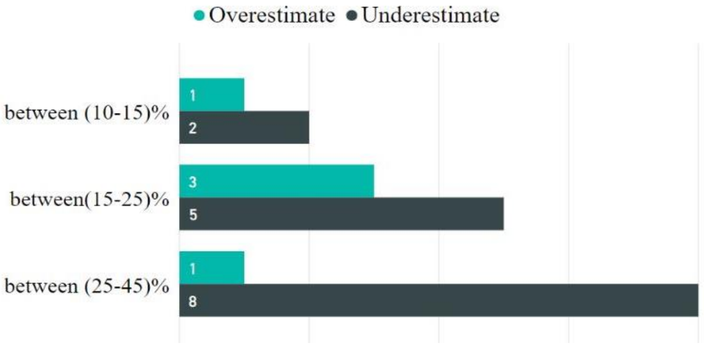

Fig. 7. Effort estimation accuracy.

5. Analysis and Discussion

In this section, we will discuss the estimation technique and its measurement type, following which we will talk about the estimation process.

5.1. Estimation Technique Analysis

The Expert Judgment and Planning Poker techniques were the most frequently used techniques for effort estimation in mobile apps which apply the Agile process. To ascertain the relationship between the two categorical variables, namely estimation techniques and the measurement types, we can use Fisher's Exact test. Fisher's test is a statistical significance method which makes it possible to examine the significance of the association between two types of groups [25]. Since the number of participants in the present study is small, Fisher's Exact test is perfect, as it is designed to deal with small samples and is an alternative to the Chi-square test, which is designed for large samples [25].

As can be seen in Table 4, the p-value of Fisher's test is 0.001 (p<0.05), and we can say there is a statistically significant relationship between the two groups' variables. A Ribbon chart is used in Figure 8 in order to quickly discover which variable has a high usage by each category. As a result, we can conclude that the story point is the best size metric to be used with the Planning Poker technique; on the other hand, the ideal hour metrics can be used with the Expert Judgment technique.

Table 4. Fisher's Exact Test for Effort Estimation Technique and Its Measurement Type

Test name	Value	Significance (2-sided)	Exact Sig. (2-sided)
Pearson Chi-Square	13.33	.0001	--
Fisher's Exact Test	--	--	.001
No. of Valid Cases	20	--	--

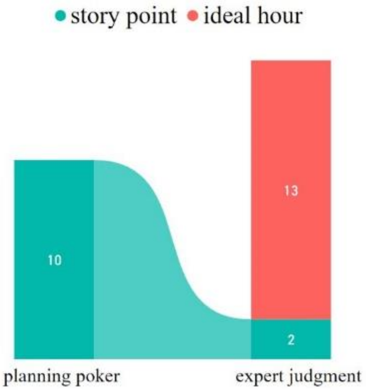

Fig. 8. Ribbon chart to observe the association between the estimation techniques with the measurement type.

5.2. Estimation Process Analysis

With regard to the RQ3 semi-structured question, each of the experts shared his/her experience regarding how to apply the estimation technique in his/her organisation. In this section, we analyse the experts' answers and recognise important themes, patterns and relationships between the data. We integrated the related data from the respondents' answers, and identified key themes from the data. Moreover, we generated categories for the estimation processes from the practitioners' response and designed a matrix to represent said data. In order to achieve this, we used a computer-aided qualitative data analysis software program, NVIVO, to help in categorising the data. From the software, we extracted the coding results and organised the coding as a thematic format. Table 6 presents a summary of a key point that compresses the experts' answers into a coding and a brief statement.

From the analysis results, we observed a relationship between the themes as follows:

1) If a task was to be completed by a developer, then:

- Most of the experts preferred the developer to assign the estimated effort value for the task (Fig. 9, Finding 1).

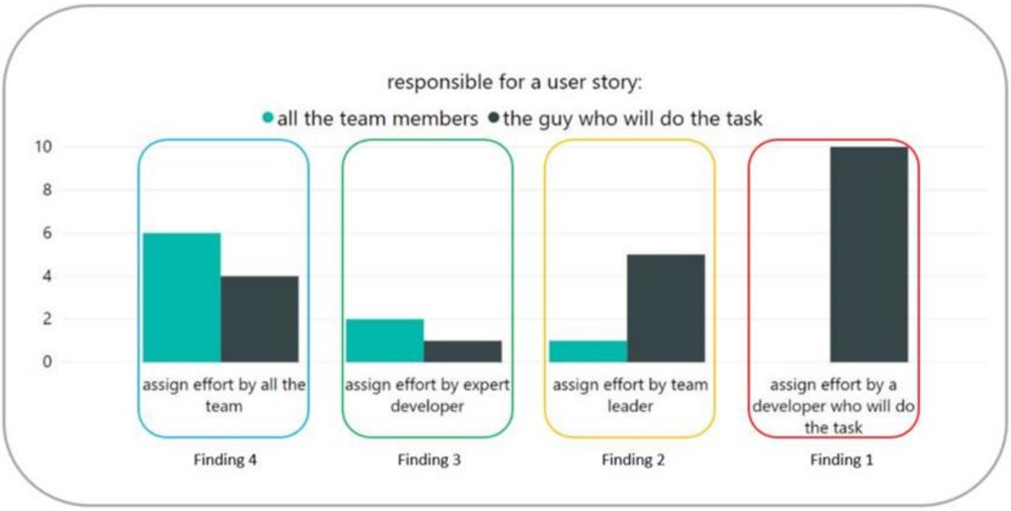

Fig. 9. Practitioners' opinions on the estimation process.

Table 5. Estimation Techniques Used, By Practitioners' Job Role

			Job role							Total
			Developers		Project Managers			QA		
Estimation technique	Planning Poker	count	4	2	1	1	1	0	1	10
	Expert Judgment	count	4		2			4		15
	COCOMO II	count	0		0			0		1
Total		count	10		5			5		20

Table 6. Summary of Practitioners' Estimation Process

Practitioner ID	Explain the task feature to		Who is responsible for a task?		Assign estimated effort to a task by			
	The developer who will do the task	All the developers	All developers	Whomever will do the task	Expert/senior developer	The whole team	Team leader	The developer who does the task
A		✓		✓	✓	✓	✓	✓
B		✓		✓		✓		
C		✓		✓				✓
D		✓		✓			✓	✓
E	✓			✓			✓	✓
F		✓		✓		✓		✓
G	✓	✓		✓			✓	✓
H		✓	✓			✓		
I					✓			
J		✓	✓		✓	✓		
K		✓		✓		✓		✓
L		✓	✓			✓		
M		✓	✓			✓		
N	✓			✓				✓
O	✓			✓				✓
P		✓	✓			✓	✓	
Q							✓	
R		✓	✓		✓	✓		
S							✓	
T	✓			✓			✓	✓
Total	**5**	**13**	**6**	**11**	**4**	**10**	**8**	**10**

- The experts accepted, at a certain level, the team leader assigning the effort value for the task (Fig. 9, Finding 2).
- The experts did not prefer the effort value to be assigned by a different senior developer (Fig. 9, Finding 3).
- The experts did not prefer the effort value to be assigned by all of the team members (Fig. 9, Finding 4).

2) If the user story "task" was to be completed by all the development team members, then:
   - Most of the experts preferred the effort to be assigned to all members of the development team (Fig. 9, Finding 4).
   - The experts, at a certain level, accepted the effort being assigned by an expert senior developer (Fig. 9, Finding 3).
   - The experts did not prefer the team leader to assign the estimated effort value (Fig. 9, Finding 2).
   - The experts did not accept the estimated effort value being assigned by a developer (Fig. 9, Finding 1).

5.3. Discussion

From the data results, we found that Expert Judgment is the most used technique, by 15 practitioners, to estimate the effort in mobile app development. Planning Poker is also a common technique, and is used by 10 experts during mobile application development. In addition, we found that when the practitioner uses the Planning Poker technique, he/she usually employs a story point to measure the effort of the user stories. In contrast, when a practitioner employs the Expert Judgment method, he/she usually tends to employ an ideal hour to measure the user story in the Agile process.

In Table 5 we can see the distribution of the practitioners based on the estimation techniques used. We can derive from the table that four experts prefer to use the mixed technique, involving both Planning Poker and Expert Judgment. They prefer to employ the mixed technique because they want to obtain more accurate estimation values, or use it as a confirmatory approach.

Project managers or developer team leaders tend to use Expert Judgment before Planning Poker, in order to avoid unrealistic estimated values from the latter technique. For the COCOMO model, which is based on the regression technique, none of the developers or QA stated that they use the COCOMO technique. The COCOMO model was only used by the project manager as an initial estimation before the project started.

6. Conclusion and Future Work

This paper presented interviews with 20 experts from 18 different companies. The interviews were focused on five aspects: mobile platform type, estimation techniques, measurement type, estimation process and estimation accuracy. The results revealed that practitioners tend to use a mix of estimation models, Expert Judgment and Planning Poker, to avoid conflicts and obtain a highly accurate estimation value. Some of the practitioners avoid the second round in PP to save more time in order to estimate other user stories. In addition, the results revealed that story point is a common measurement type for effort when the Planning Poker technique is used; in contrast, ideal hour is used when the Expert Judgment technique is chosen. The practitioners recommended that, when a developer is going to implement a user story, it would be better for the developer to assign an effort estimation value to that user story so as to avoid inaccuracy of estimation. Developers tended to give high estimated value to avoid working under pressure, and therefore a mixed estimation technique was recommended to confirm the estimated effort value.

With regard to previous studies, no past research has investigated or validated the Agile estimation techniques in the mobile app context. There is a need to define effort estimation factors and metrics in mobile app development to help the development team in assigning reasonable estimation values for each user story and in avoiding random activity. Moreover, current estimation techniques suffer from low accuracy levels due to lack of formalism (as the survey concluded) and other reasons. Therefore, future work is required to validate the Planning Poker technique in mobile app development. In addition, a proposed framework is required to reduce the ad-hoc activity created by the Expert Judgment and Planning Poker techniques in order to increase the accuracy level of the effort estimation.

References

[1] Cohn, M. (2005). *Agile Estimating and Planning*, 1st ed. Prentice Hall.

[2] Usman, M., Mendes, E., Weidt, F., & Britto, R. (2014). Effort estimation in agile software development: A systematic literature review. *Proceedings of the ACM Int. Conf*.

[3] Grenning, J. (2002). Planning poker or how to avoid analysis paralysis while release planning. *Hawthorn Woods Renaiss. Softw. Consult*.

[4] Mahnič, V., & Hovelja, T. (2012). On using planning poker for estimating user stories. *J. Syst. Softw., 85(9)*, 2086–2095.

[5] Haugen, N. C. (2006). An empirical study of using planning poker for user story estimation. *Proceedings of the Agil. 2006 Conf. IEEE*.

[6] Kumar, N., Ajit, K. T., Krishna, H., & Manjula, R. (2016). Challenges and best practices in mobile application development. *Imp. J. Interdiscip. Res., 2(12)*, 1607–1611, 2016.

[7] Rahimian, V., & Ramsin, R. (2008). Designing an agile methodology for mobile software development: A hybrid method engineering approach. *Second Int. Conf. Res. Challenges Inf. Sci*.

[8] Souza, L. S. D., & Aquino, G. S. D. (2014). Estimating the effort of mobile application development. *CS IT-CSCP*.

[9] Kaur, A., & Kaur, K. (2018). Systematic literature review of mobile application development and testing effort estimation. *J. King Saud Univ. - Comput. Inf. Sci*.

[10] Costa, P., Paiva, A. C. R., & Nabuco, M. (2014). Pattern based GUI testing for mobile applications. *Proceedings of the 2014 9th International Conference on the Quality of Information and Communications Technology*.

[11] Liu, Z., Hu, Y., & Cai, L. (2014). Software quality testing model for mobile application. *Lecture Notes in Computer Science Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics*.

[12] Arzenšek, B., & Heričko, M. (2014). Criteria for selecting mobile application testing tools. *CEUR Workshop Proceedings*.

[13] Santos, A., Kroll, J., Sales, A., Fernandes, P., & Wildt, D. (2016). Investigating the adoption of agile practices in mobile application development. *Proceedings of the 18th Int. Conf. Enterp. Inf. Syst*.

[14] Altaleb, A. R., & Gravell, A. (2018). Effort estimation across mobile app platforms using agile processes: A systematic literature review. *Journal of Software, 13(4)*, 242–259.

[15] Ferrucci, F., Gravino, C., Salza, P., & Sarro, F. (2015). Investigating functional and code size measures for mobile applications. *Proceedings of the 41st Euromicro Conf. Softw. Eng. Adv. Appl. SEAA*.

[16] Heeringen, H. V., & Gorp, E. V. (2014). Measure the functional size of a mobile app: Using the cosmic functional size measurement method. *Proceedings of the 2014 Jt. Conf. Int. Work. Softw. Meas. IWSM 2014 Int. Conf. Softw. Process Prod. Meas. Mensura*.

[17] Avanzo, L. D., Ferrucci, F., Gravino, C., & Salza, P. (2015). COSMIC functional measurement of mobile applications and code size estimation. *Proceedings of the 30th Annu. ACM Symp. Appl. Comput*.

[18] André, N. (2013). Measuring mobile application size using COSMIC FP. *Softw. Metr. Kongress*.

[19] Nitze, A., Schmietendorf, A., & Dumke, R. (2014). An analogy-based effort estimation approach for mobile application development projects. *Proceedings of the 2014 Jt. Conf. Int. Work. Softw. Meas. IWSM 2014 Int. Conf. Softw. Process Prod. Meas. Mensura*.

[20] Francese, R., Gravino, C., Risi, M., Scanniello, G., & Tortora, G. (2015). On the use of requirements measures to

predict software project and product measures in the context of android mobile apps: A preliminary study. *Proceedings of the 41st Euromicro Conf. Softw. Eng. Adv. Appl.*

[21] Shahwaiz, S. A., Malik, A. A., & Sabahat, N. (2016). A parametric effort estimation model for mobile apps. *Proceedings of the 2016 19th Int. Conf. (INMIC)*.

[22] Usman, M., Mendes, E., & Börstler, J. (2015). Effort estimation in Agile software development: A survey on the state of the practice.

[23] Jørgensen, M., Boehm, B., & Rifkin, S. (2009). Software development effort estimation: Formal models or expert judgment? *IEEE Softw.*, *26(2)*, 14–19.

[24] Saunders, M., Lewis, P., & Thornhill, A. (2009). *Research Methods for Business Students*, 5th ed. Pearson Education, 2009.

[25] Field, A. (2009). Discovering statistics using SPSS. *Discovering Statistics Using SPSS*

Abdullah Altaleb is a PhD student specialising in software engineering at the University of Southampton, United Kingdom. He accomplished his master’s degree in software engineering at Southern Methodist University in Dallas, Texas in the US. He worked at the GOSI and NIC organisation as a software and application developer for 4 years, with a focus on the Java EE platform. He also worked with MRN Inc. as a software architect in San Jose, California.

Andrew Gravell was born 1956 in Exeter, UK. He studied mathematics and computer science at Cambridge, and was awarded a PhD by the University of Southampton in 1995. He held the position of software developer for 7 years, most of which was spent at IBM's Hursley laboratory working on a range of projects and writing software for microprocessor-based systems. In 1987 he joined ECS, Southampton, as a lecturer in computer science. He was promoted to senior lecturer/associate professor in 2002. Andrew also held the position of associate dean for education and the student experience, Faculty of Physical and Applied Sciences from 2010 to 2015. He has supervised a number of PhD students and published on a range of approaches to software development, empirical approaches to software development, agile methods, and lean information technology.