2017 IEEE/ACM 4th International Conference on Mobile Software Engineering and Systems (MOBILESoft)

A Set of Metrics for the Effort Estimation of Mobile Apps

Gemma Catolino, Pasquale Salza, Carmine Gravino, Filomena Ferrucci
University of Salerno, Italy
E-mail: {gcatolino, psalza, gravino, fferrucci}@unisa.it

***Abstract*—In this work, we report a study carried out to identify a set of metrics to early estimate the development effort of mobile apps. The applied methodology was inspired by the work of Mendes et al. who addressed a similar problem in the field of web apps. In particular, we extracted an initial set of metrics by analyzing the online quotes forms that companies made available on their websites. Afterward, a Delphi approach with four project managers was employed to identify the proposed set of 41 relevant factors.**

***Index Terms*—Effort Estimation; Mobile Applications; Metrics**

I. Introduction

In the recent years, the complexity and size of mobile apps have been growing and the development of high quality mobile apps requires a systematic engineering approach and the identification of specific management tools. Effort estimation is a key project management activity needed for project planning, staff resources estimation, cost estimation, quality control and benchmarking [1]. For traditional software several approaches have been defined to support this task that can be divided into two main categories, namely the non-model- and model-based methods [2]. Broadly speaking, non-model-based methods involve the judgment of human experts, who provide a prediction based on their previous experience [2]. On the other hand, model-based approaches rely on the definition of a set of cost drivers used as independent variables in prediction models aimed at estimating a numerical variable, e.g., the number of man/hours or the effort required to develop/maintain a software [2]. One of the advantages of model-based approaches is that they are more replicable. Nevertheless, they critically depend on the identification and evaluation of cost drivers. Different approaches can be devised based on the employed cost drivers and each one can be applied in a different phase of the development process, once the information to evaluate the required cost drivers is available.

Currently, no much work has been devoted to identify suitable approaches for effort estimation of mobile apps, that can be particularly challenging for project managers due to the new development and programming approaches adopted [3]. The aim of this work is to fill the gap by proposing a set of cost drivers to be employed for model-based effort estimation of mobile apps. The methodology adopted to get the proposal was inspired by a similar work carried out by Mendes et al. in the context of web applications [4]. The cost drivers gathered by Mendes et al. determined the creation of a dataset, named TUKUTUKU, that has been employed in several investigations [5]–[7].

Similarly to Mendes et al., the first step of our methodology consisted in analyzing online quotes forms made available by software companies in order to extract an initial set of metrics. Then, we involved four project managers having a good experience in managing and developing mobile apps with the goal of validating the initial set of metrics derived during the first phase. In particular, starting from a set of 48 metrics, 36 of them were confirmed as the ones having a higher impact on the effort of a project, while 12 were discarded. Furthermore, the managers also suggested the introduction of new 5 metrics not extracted in the previous step.

The rest of the paper is organized as follows. Section II presents the related work. The study design and the resulting final set of metrics are presented in Sections III and IV, respectively. Section V concludes the paper with future work.

II. Related Work

Most of the work in the contest of mobile effort estimation has been concerned with the study of the size estimation for mobile apps. Indeed, software size is recognized as one of the most important cost drivers and is employed in many effort estimation models. In particular, the applicability of the *Functional Size Measurement* (*FSM*) as *Function Point Analysis* (*FPA*) [8] and *COSMIC* [9] method to mobile apps was investigated [10]–[16]. As for the use of FPA, the *IFPUG* proposed a guide explaining how to adopt this method in the context of mobile apps [14], while recent studies proposed a set of guidelines for an approximate and quick sizing of mobile apps in terms of COSMIC [10], [13], [15]. In particular Sellami et al. [15] defined the "action type", a way to simplify the counting of the number of COSMIC Function Points by assigning the types of actions which characterize the data movements to be determined when applying COSMIC [9]. van Heeringen and van Gorp [13] introduced a set of assumptions (e.g., considering an app just as a presentation layer, without any persistent storage) related to the characteristics of the app to be measured, which allow the estimation of the size in terms of COSMIC Function Points. Successively, Ferrucci et al. proposed and assessed [10], [17] a new set of guidelines able to help in measuring mobile business apps, containing persistent storage as an internal database (i.e., it is accessible by Read and Write data movements). Other methods focused their attention on the measurement of game apps [16] or the

978-1-5386-2669-6/17 $31.00 © 2017 IEEE
DOI 10.1109/MOBILESoft.2017.31

application of the standard COSMIC method [11]. The use of COSMIC for measuring complex applications composed by both mobile and cloud-based architectures was also considerer by Cruz et al. [12] and Ferrucci et al. [18].

The limitation of the above approaches is that they can be employed once the functional user requirements have been well documented, thus only after the requirements engineering phase has been completed. Differently, our goal is to identify and investigate relevant factors that can be estimated in the early phases of software development, in a way similar to the TUKUTUKU [4] cost drivers.

III. STUDY DESIGN

In this section, we present the methodology that we employed to extract relevant metrics that can be found in the early phase of development of mobile apps. Although such methodology is inspired by the work of Mendes et al. [4], there are some differences between the two approaches. The main difference lies in the considered object of interest, i.e., the mobile apps compared to the web applications. Other differences concern the steps performed to extract the metrics. Indeed, even though we share the first step with Mendes et al. (i.e., the metrics collection through the analysis of the online quotes), in the second step we did not interview a single expert, but we rather decided to survey four experts in order to take into account more than a single point of view.

A. 1st Step: Online Quotes Mining

The *goal* of the first step of the study was the analysis of the online quotes made available by companies on the web, with the *purpose* of extracting an initial set of metrics based on the information requested by the companies. The *context* of the study consisted of every company having a website and providing an online form for requesting a quote about the development of a mobile apps.

We used an automatic search tool, named GOOGLE-SCRAPER[1], which is publicly available and open source on *GitHub*. In particular, the tool behaves as a *Google* search engine and receives as input the query to search and a maximum number of links to mine. When we designed the query, we firstly included all the terms possibly referring to mobile apps (e.g., "Android application" or simply "app"). Then, we included the terms relating to the presence of an online quotes, such as the "estimated price" or simply "quote". Finally, we considered synonyms and abbreviations. The final query was the following:

(*"quote"* OR *"quote form"* OR *"price estimate"*)
(*"mobile application"* OR *"mobile app"* OR
"smartphone app" OR *"smartphone application"* OR
"android app" OR *"android application"* OR *"ios app"*
OR *"ios application"* OR *"windows phone app"* OR
"windows phone application")

The query result consisted of a list of links that we manually validated. The goal of the validation was to filter out all the links that were not related to online quotes, to have an initial set of metrics. To this aim, we firstly discarded the links that were devoid of any form of quote within the web page, while in a second step we discarded the links that presented generic quotes that did not give any useful information in our context (e.g., the information of the customer or generic description of the project to be developed). The validation process was performed and cross-checked by two of the authors. The output of this phase consisted of a set of metrics that were employed in the validation phase described in the next subsection.

[1]https://github.com/NikolaiT/GoogleScraper

B. 2nd Step: Survey with Experts

The *goal* of the second step of the study was to validate the initial set of metrics by experts having a good knowledge of both effort estimation methods and mobile apps. The *purpose* was to exploit the involved experts in order to (i) confirm/refute the usefulness of the metrics/cost drivers to estimate the effort in the early phase of development of mobile apps, defined during the first step of the process and (ii) possibly discover new factors that were not revealed after the first analysis. The *context* of the study was composed by four project managers with more than 4 years of experience in managing mobile development and effort estimation.

The selection of the types of participants involved in the study was not random. In fact, the selected project managers are responsible for leading the projects in their companies, in addition to managing the people, resources and the effort needed to complete the project. Two of them work for large companies, while the other work in local companies.

With the goal of bringing together the opinions of the participants and providing a joint solution, we adopted the *Delphi* method [19]. The Delphi method is a structured communication technique, originally developed as a systematic, interactive forecasting method which relies on a panel of experts. The experts answer questionnaires in two or more rounds. After each round, a facilitator provides an anonymous summary of all judgments. As a consequence, experts are encouraged to revise their earlier answers in the light of the replies of other members. The process is stopped after a predefined stop criterion (e.g., number of rounds, achievement of consensus, stability of results).

In our case, we firstly proceeded with the design of an online questionnaire using the *Google Form* platform. Once all the participants completed the questionnaire, the different opinions have been collected and grouped into a single document. Finally, such document was sent to the experts, who had the opportunity to express newer opinions based on the answers provided by the other participants. If a common solution was found in this stage, the process ends. Otherwise, the process would restart until a common solution would have been found. The first author of this paper has played the role of facilitator. Specifically, the steps are detailed in the following:

Phase 1: Each participant was initially contacted by e-mail that summarized the purpose of the work with an explanatory document which included: (i) a brief explanation of the goal of the work and (ii) the list of metrics with a description. The instructions to fill in the questionnaire were also included.
The questionnaire was composed of three parts:

1) **Pre-questionnaire:** a pre-questionnaire aimed at collecting general information on the background of the participants;
2) **Metrics evaluation:** for each metric, the participants were asked to evaluate the level of importance for early effort estimation by using a *Likert* scale intensity [20] from 1 ("not at all") to 5 ("very much");
3) **Suggestions:** the participants were asked to suggest possible additions, removals, or changes to the metrics.

Phase 2: Once received the responses from the questionnaire, the facilitator analyzed the emerged opinions. As for the second part of the questionnaire, i.e., the evaluation of the metrics, we calculated the mean, median, minimum, maximum, and standard deviation of the scores assigned by the participants to each metric. Regarding the third part of the questionnaire, we aggregated the answers about the project managers opinions considering the percentage of participants who believed the metrics contained in a given category (e.g., the "Features") were meaningful. We also collected any further opinions about the addition/modification/removal of metrics in the form of open questions, in order to understand the rationale behind the choices of the participants;

Phase 3: Once analyzed the questionnaire, the facilitator aggregated the results into a single document, and sent it to the experts, who had the opportunity to express newer opinions based on the judgment provided by the other participants. This step was the most sensitive part of the study because we had to collect and merge all the different opinions to create a common solution. For this reason, we provided participants with a new online questionnaire containing all the opinions collected. The goal was to push experts to focus on the opinions of the other participants in order to reach a consensus on the final set of metrics.

In particular, the questionnaire contained two parts:

1) **Scores assigned:** the scores assigned by experts during the 1st phase are revealed. We showed to the participants the mean, median, minimum, maximum, and standard deviation of the metrics for which we found high variability, asking each expert if they agreed with the score;
2) **Discussion:** the discussion and detailed opinions of the experts about possible additions, removals, and changes to the set of metrics are presented.

If all the participants approved the evaluation of metrics and the given suggestions, then the common solution was considered as found.

C. Threats to Validity

In the context of this study, the threats to *construct validity* are mainly related to how we measured the usefulness of the metrics by the project managers. As explained in Section III-B, we asked the project managers to tell us whether they perceived as useful the set of metrics. In addition, we asked if they wanted to change something or not, giving some reasons. For the assessment of each metric, we used a Likert scale [20] that permitted the comparison of responses from multiple respondents. It was very useful for the comprehension in the 2nd phase of the questionnaire, when it was needed to converge the different opinions in a shared solution.

Threats to *internal validity* can be related to the use of the GOOGLESCRAPER tool. Its use mitigated the possibility of human error in the online quote search phase. However, we needed to calibrate the parameters of the tool, i.e., the query and the maximum number of links. While the query contained all the words related to the mobile apps, we set the maximum possible number of links in order to overestimate the results and, therefore, trying to gather all the web pages of interest. Moreover, these results were manually cross-checked between two of the authors.

In this study, threats to *external validity* can be connected with the pool of the participants to the study. We chose them having at least 4 years of experience in managing mobile development and with a good knowledge of effort estimation. Moreover, all participants were Italian. For this reason, the opinions may reflect a limited company reality. Further investigations should be performed.

IV. RESULTS

The following section describes the results of the two steps which led to the final set of metrics.

A. 1st Step: Online Quotes Analysis

During the first phase, we found a total of 377 links. The list of links was placed in a CSV file containing: (i) the link and title of the web page, and (ii) a brief description of the content of the page provided by *Google*. Then, we started with the two phases of validation. From the initial set of 377 discovered quote forms, we selected 28 real quote forms. The other 349 were false positives, not presenting any actual quote form (237) or having forms too general to induce any useful information (112).

Starting from the set of 28 links, we extracted the draft of all the factors taken into account in the online quotes. The list was composed by 48 metrics and for each of them, there is a measuring scale (e.g., a Likert scale intensity from 1 to 5, measuring to what extent the presence of a factor is relevant). Moreover, we grouped the metrics into seven categories, i.e., *Features*, *Application GUI*, *Cost Driver*, *Project's Metrics*, *Application Functionality*, *Application Size*, and *Other Metrics*. Due to space limitation, the complete table named `Draft_Set_Metrics` reporting an initial draft of the set of metrics with a description is available in the online appendix [21].

B. 2nd Step: Analysis of survey with Experts

In this stage we analyzed the opinions of the experts with regard to the initial set of metrics. For each category, Table I reports the percentage of participants who considered as useful the metrics in that category, as well as the percentage of them who would suggested to add, modify, or remove metrics. The first thing that is evident is that the experts perceived almost all the metrics as "useful" for estimating the effort in the early phase of development of mobile apps.

Table I: Opinions of the participants regarding the usefulness and need of modifying the set of metrics.

Categories	Usefulness	Addition	Removal	Modification
Features	100 %	50 %	0 %	0 %
Application GUI	67 %	17 %	0 %	17 %
Cost Drivers	84 %	50 %	17 %	17 %
Project's Metrics	67 %	0 %	17 %	17 %
Application Functionality	84 %	17 %	0 %	0 %
Application Size	100 %	0 %	0 %	0 %
Other Metrics	84 %	34 %	0 %	0 %

The case of the category *Features* and *Application Size* is particularly interesting since all the participants affirmed that all the metrics were needed, highlighting the potential of the metrics. In the other cases, we can still see that the experts expressed a consensus about the usefulness of the metrics. At the same time, the experts helped us in the definition of new metrics to consider, especially in the cases of the *Features* and *Cost Drivers* categories. For instance, Expert #4 reported the need of considering the security support of mobile apps. According to his/her opinion, this was needed because setting up a security platform may be a time-consuming and effort-prone activity. As for the removal of metrics, the participants were less prone to suggest modifications. Expert #2 gave us the rationale behind this decision. Specifically, he said that *"All the shown metrics have a potential impact on the effort. Thus, I would not remove things, but rather I would add further metrics"*.

On the other hand, we found several cases where a joint solution was not found. It is visible in the table named `Score_metrics` in appendix [21] that reports the mean, median, minimum, maximum, and standard deviation of the scores assigned by the participants to each metric. It is particularly interesting the discussion of the *Social sharing* feature. Here the participants expressed divergent opinions because two of them believed that this feature is quite easy to implement due to the possibility of using external APIs, such as SHARETHIS[2]. Instead, the remaining two participants, even considering the possibility of using existing APIs, stated that their integration might require a lot of effort. Another interesting example regards the metric *Project estimated start date*, which obtained different scores (e.g., Expert #1 estimated its importance by assigning a score of 4, while Expert #3 assigned 1). When explaining the motivations using the open questions in the last part of the questionnaire, Expert #3 proposed to group together the metrics *Project estimated start date* and *Project estimated end date*, and evaluate the resulting metric using the time between the kick-off meeting and end of the project. Therefore, stating that having a value reporting the total time available rather two dates may give a more clear idea of the time and the costs of the project.

C. The Final Set of Metrics

Based on the opinions expressed by the experts, we aggregated the results into a single document and sent it to all participants as a support material for carrying out the final phase. Once received the answers from the experts, we manually analyzed the results to provide the final set of metrics. First of all, it is important to note that the participants reached a shared solution and the disagreements were all resolved. Thus, no more analysis involving the experts was needed. The complete table reporting the final set of metrics is available in the online appendix [21].

Looking at the metrics in the online appendix, we can notice that the categories have been reorganized to better reflect the opinions of the experts. In particular, the category *Generalities* includes the metrics characterizing all the apps to develop and the category *Features* includes all the functionality that need to be developed.

Moreover, comparing to the initial draft of the metrics, we changed the name of some of them without make them lose their semantics. For instance, a metric called *New application or enhancement* has been replaced by *Development type*.

Regarding the previously mentioned *Social sharing* and *Project estimated start date* metrics, the experts decided to keep the former. The *Project estimated end date* metric, renamed to *Defined deadline*, was considered as more important than the *Project estimated start date* one since it declares the deadline of the project. Moreover, the metrics are not able to give a clear indication of the cost of the project.

As for the size, in the preliminary classification we measured it using the number of features but the total number of features would have been equal to the number of true values assigned to the features to be implemented. Thus, we changed the count using the number of static and dynamic views of the app.

Moreover, most of the metrics previously contained in the category *Other metrics* were removed since considered as poor indicators of the effort by the participants. For the same reason, the metrics *Payment information collection*, *What type of business owns the app idea?* and *Mobile application type* were removed.

On the other hand, several metrics were added after the collection of participants' opinions. In particular, metrics as *Backward compatibility*, *Analysis security support*, *Platform type* and *User target* were considered relevant by all the experts involved in the study.

V. CONCLUSIONS AND FUTURE WORK

This paper presented a new set of 41 metrics to estimate the effort in the early phase of development of mobile apps, which were validated by four project managers. To deeper understand the importance and validity of the proposed metrics, we plan to conduct a further evaluation with industrial companies that would also give us a clear indication of the most influential metrics. Furthermore, we plan to evaluate the effectiveness of these metrics as dependent variables in a prediction model able to estimate the effort needed for the development of mobile apps. To this aim, we are starting the preparation of mobile project data entry forms to gather data on mobile projects worldwide, that will be used to train and test the effort prediction model.

[2]https://developer.sharethis.com

REFERENCES

[1] I. Sommerville, *Software Engineering: (Update) (8th Edition) (International Computer Science)*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2006.

[2] L. C. Briand and I. Wieczorek, "Resource estimation in software engineering," *Encyclopedia of software engineering*, 2002.

[3] A. I. Wasserman, "Software engineering issues for mobile application development," in *Proceedings of the FSE/SDP workshop on Future of software engineering research*. ACM, 2010, pp. 397–400.

[4] E. Mendes, N. Mosley, and S. Counsell, "Investigating early web size measures for web cost estimation," in *Proceedings of EASE'2003 Conference, Keele*, 2003, pp. 1–22.

[5] B. A. Kitchenham and E. Mendes, "Software productivity measurement using multiple size measures," *IEEE Trans. Software Eng.*, vol. 30, no. 12, pp. 1023–1035, 2004. [Online]. Available: http://dx.doi.org/10.1109/TSE.2004.104

[6] E. Mendes, S. D. Martino, F. Ferrucci, and C. Gravino, "Cross-company vs. single-company web effort models using the tukutuku database: An extended study," *Journal of Systems and Software*, vol. 81, no. 5, pp. 673–690, 2008.

[7] A. Corazza, S. D. Martino, F. Ferrucci, C. Gravino, F. Sarro, and E. Mendes, "Using tabu search to configure support vector regression for effort estimation," *Empirical Software Engineering*, vol. 18, no. 3, pp. 506–546, 2013. [Online]. Available: http://dx.doi.org/10.1007/s10664-011-9187-3

[8] A. J. Albrecht, "Measuring application development productivity," in *Proceedings of the joint SHARE/GUIDE/IBM application development symposium*, vol. 10, 1979, pp. 83–92.

[9] R. Dumke and A. Abran, *COSMIC Function Points: Theory and Advanced Practices*. CRC Press, 2016.

[10] L. D'Avanzo, F. Ferrucci, C. Gravino, and P. Salza, "Cosmic functional measurement of mobile applications and code size estimation," in *Proceedings of the 30th Annual ACM Symposium on Applied Computing*. ACM, 2015, pp. 1631–1636.

[11] A. Nitze, "Measuring mobile application size using cosmic fp," in *DASMA Metrik Kongress*, vol. 11, 2013.

[12] A. M. Rosado da Cruz and S. Paiva, "Modern software engineering methodologies for mobile and cloud environments," *Information Science Reference*, pp. 60–87, 2016.

[13] H. van Heeringen and E. Van Gorp, "Measure the functional size of a mobile app: Using the cosmic functional size measurement method," in *Software Measurement and the International Conference on Software Process and Product Measurement (IWSM-MENSURA), 2014 Joint Conference of the International Workshop on*. IEEE, 2014, pp. 11–16.

[14] T. Preuss, "Mobile applications, function points and cost estimating," in *International Conference on Cost Estimation and Analysis Association (ICEAA)*, 2013.

[15] A. Sellami, M. Haoues, H. Ben-Abdallah, A. Abran, A. Lesterhuis, C. Symons, and S. Trudel, "Sizing natural language/uml requirements for web and mobile applications using cosmic fsm," Tech. Rep., 2016.

[16] N. A. S. Abdullah, N. I. A. Rusli, and M. F. Ibrahim, "Mobile game size estimation: Cosmic fsm rules, uml mapping model and unity3d game engine," in *Open Systems (ICOS), 2014 IEEE Conference on*. IEEE, 2014, pp. 42–47.

[17] F. Ferrucci, C. Gravino, P. Salza, and F. Sarro, "Investigating functional and code size measures for mobile applications: A replicated study," in *International Conference on Product-Focused Software Process Improvement*. Springer, 2015, pp. 271–287.

[18] F. Ferrucci, C. Gravino, and S. Pasquale, "Using cosmic for the functional size measurement of distributed applications in cloud environments," in *Software Project Management for Distributed Computing: Life-Cycle Methods for Developing Scalable and Reliable Tools*. Springer, 2017.

[19] H. A. Linstone, M. Turoff *et al.*, *The Delphi method: Techniques and applications*. Addison-Wesley Reading, MA, 1975, vol. 29.

[20] R. Likert, "A technique for the measurement of attitudes," *Archives of Psychology*, vol. 22, no. 140, 1932.

[21] G. Catolino, P. Salza, C. Gravino, and F. Ferrucci, "A set of metrics for effort estimation of mobile apps - online appendix - http://tinyurl.com/joqcgnf."