MethApp4Mob: A Methodological Approach for Mobile Applications Development

Iris Galeano, Mauricio Merín, Magalí González, Luca Cernuzzi

[1]Department of Electronics and Computer Science Engineering
Catholic University "Nuestra Señora de la Asunción" (UC)
Teniente Cantaluppi y G. Molinas – Asunción – Paraguay

iris.ariana07@gmail.com, mimt80@gmail.com, {mgonzalez,lcernuzz}@uc.edu.py

Abstract. *The aspects of mobile technology have introduced challenges for developers, raising the need to improve the software development methodologies traditionally used. As a result of the analysis of the state of the art, an agile methodological approach for its development is presented, consisting of 8 phases: negotiation, planning, design, implementation and alpha testing, beta testing and feedback analysis, application publication, release and maintenance, promotion, and a series of steps, good practices, principles and tools, to offer mechanisms to accompany and streamline the development process. Some validations were carried out including a exploratory study, seeking to propose a robust methodology but at the same time easy to apply in a company.*

1. Introduction

The fast growth of mobile technology and the peculiarities of this type of applications, introduce challenges on how to approach the complete life cycle development of mobile applications in order to achieve the best acceptance from users. Indeed, the development of mobile applications is usually oriented to horizontal or vertical markets rather than specific projects for a single client, which introduces the need to think about the markets of potential users with their own distribution channel with specific rules that must be fulfilled. At the same time, the scope and complexity of mobile applications are often smaller than large systems running on other technology platforms. Furthermore, the current technological market offers a great diversity of mobile devices, giving rise to diverse challenges for those who develop applications for such devices, as they have particular characteristics that should be considered: they have limited space for visualization; need to adapt the user interface to each type of device; technological park in constant evolution; among others [Wasserman 2010]. This introduces for development companies the need to focus not only on the technical aspects of modeling and development, but also to other complementary aspects of management and time-to-market, application acceptance challenges for different target of end-users, promotion plan for the rapid adoption of the application, etc.

The specialized literature presents several proposals for the field of mobile software production [Scharff and Verma 2010][Usman et al. 2014] [El-Kassas et al. 2014] [Heredia et al. 2014] [Ettifouri et al. 2017] [Khan et al. 2019]. However, most of such proposals lack a global vision of software that covers all the particularities of the life cycle of a mobile application [Wasserman 2010] [D. et al. 2012]. Normally the methodological

proposals focus on technical stages covered by software engineering (requirements, design, development and testing). However, in several projects the results obtained are not the desired ones [Lee 2014].

Hence, to be successful in the industry, a good development approach must also encompass other relevant phases that are more related to the management aspects, including a project vision to monitor agreements and modifications to the final application, negotiation and planning; to the quality assurance, with emphasis in validations with target users; to the market, covering launch and promotion initiatives, among others.

This study is the result of collaboration between researchers and industry professionals to formulate a methodological proposal that covers both the technical aspects and the complementary aspects of the life cycle of mobile applications. A practical approach is proposed to be used in real mobile projects. The proposal is based on different good practices, mainly from agile methods, and it offers general guidelines to work teamsthat can undertake the development project with support tools in each phasepointing to good levels of product quality. A key aspect of the approach is its simplicity, with a low learning curve that promotes the quick understanding of team members and accelerates their incorporation into the company. Finally, an initial validation in academic environment together with other preliminary validations carried out by developers in the industrial sector provide some insights on the strengths and limitations of the proposal.

The rest of the paper is organized as follows. Section 2 presents related work. Section 3 presents the methodological approach for the development of mobile applications. Section 4 describes different validation experiences including a exploratory study and other partial validations that were carried out by developers in the industrial sector. Finally, section 5 presents the conclusions of this work and some outlines for future work.

2. Related Work

The mobile application development is a hot topic with many publications, as showw the systematic mapping study (SMS) carry out by Galeano *et al.* [Galeano et al. 2016]. The study covers the period between 2007 (launch of the first iphone) and 2015 including 28 articles (out of 601 retrieved) from different sources: IEEE, ACM, Springer, ScienceDirect, Wiley, SAGE and Thomson Reuters.

According to the SMS, most proposals for mobile software development focus on ***Model-Driven Development*** , ***Multiplatform*** approaches , and ***Agile Development with Iterative Development*** , showing no preferential inclination for any existing proposal. Subsequent studies between 2016 and 2020 have yielded similar results [Ettifouri et al. 2017] [Rieger 2017] [Khan et al. 2019].

Considering the perspective of the general approach, Model-Driven Development and Multiplatform proposals allow generating applications capable of running on multiple platforms, without having to develop the same application from scratch for each considered platform, objective sought by many software companies that try to reach the largest number of users. On the other hand, agile methodologies are characterized by better capture of changing requirements and risk management; their main objective is to reduce development time.

On the other hand, Agile methods such as Scrum [Schwaber and Beedle 2001],

Extreme Programming [Beck and Andres 2004] or Kanban [Anderson 2010], are widely adopted for the development of mobile applications [Shen et al. 2012], [O'Hagan et al. 2014]. However, despite their popularity, agile methodologies do not consider some aspects of the life cycle of mobile applications as those tasks related to the publication and promotion of the application after its implementation.

Moreover, the review conducted by Hosbond *et al.*[Hosbond and Nielsen 2005] mentions that the traditional systems development focuses on project-level development tasks with very little focus on company-level or inter-organizational development tasks. However, the development of mobile applications is strongly focused on the business perspective, which extends the reach to higher organizational levels.

Considering more technical perspective, in terms of tools, Joorabchi[Joorabchi et al. 2013] points out that current testing tools do not offer the same level of support for different platforms while lacking important features for testing on mobile devices such as mobility, location services, sensors, gesture control, among others, so testing becomes a significant challenge throughout the application development project. On the other hand, better tools are needed to monitor and analyze certain metrics of applications in development such as memory management, battery usage, network performance, among others. The ***Natives*** applications are the most named in the analyzed studies, followed by the ***Hybrids*** applications and finally the ***Web*** applications. In those applications where the user experience and the use of device-specific features are necessary, the best alternative is the native applications.

In addition, an appreciable aspect is the analysis of the usefulness of the methodologies proposed in the industrial field. In this regard, Picco *et al.*[Picco et al. 2014] suggest a fundamental question to the software engineering community. They focus on an in-depth analysis of the processes that are currently being adopted by large companies such as Google, Apple, which may not be totally consistent with traditional practices that are still taught in universities. The theory dictated must be accompanied by elements demanded by real world as a means to offer real solutions to the problems within the development of mobile applications.

One of the aspects that surprises is the low presence of studies referring to other objectives such as the integration of marketing practices with the rest of the phases, which is a key element currently for the success of an application. Another unexploited objective refers to the productivity of the development team, the lack of support for communication and collaboration among team members, as well as a better organization of activities during the development process.

3. MethApp4Mob: A Methodological Approach for Mobile Applications Development

In addition to the review of the literature we considered the experience of the authors and practitioners of the industry in mobile application development projects. The first author is a semi-senior mobile and web application developer. From industry we collaborate with two senior developers from the same company. One of them has extensive experience in enterprise application development using emerging open source web technologies. The other is a senior developer of mobile and web applications. The company is primarily dedicated to enterprise web and mobile development using open source technology. For

the development they mainly use hybrids of agile methodologies. The experience of the company's collaborators enriched the vision of both the technical aspects of development and the project management and interaction with the client. Thus, a methodology based on an incremental iterative model with agile features is designed for tailored native mobile applications development considering its lifecycle. We opted for a focus on the custom mobile applications development because more and more companies contact software enterprises to develop their own mobile application. Likewise, most developers opt for the native applications development due to the user experience, especially in those contexts where accessing the features of the device is essential.

As we seen, the agile methodologies are in agreement with several of the elements present in the ecosystem proposed in this work, which have proven to be appropriate for certain aspects of the mobile application development process. Specifically, the foundations of the Scrum framework have been taken to define the proposed methodology. Scrum seeks early software delivery to the client (between 2 weeks and 2 months) with value to increase the client's satisfaction. Unlike other projects, resistance to change is not encouraged, but this is presented as a strength to take advantage of to increase the customer's competitive advantage and satisfaction. There is a high level of interaction between the team members and the client, which is fundamental considering that the methodology is oriented to the tailored software development, where the client must be an important actor during the execution of the project. On the other hand, it is intended to obtain a constant rhythm of development, where the deliveries are iterative and incremental, providing new features in each iteration or sprint, working in self-organized teams. The choice of Scrum offers two main advantages: on the one hand, it is a method that is easy to learn and put into practice, which is essential for less experienced work teams; and on the other hand, the constant delivery of functional versions provides greater control to the production process because it allows to identify failures in time for immediate correction in future deliveries.

Our experience developing mobile applications allowed also to identify other essential aspects. In the initial negotiations with the client, it is important to analyze and estimate other costs in addition to the development of the application: need for servers, testing infrastructure, payments for the publication channel, among others. Thus, it is paramount to define the scope of application development and services after its publication taking into account that a mobile application can evolve over time. The meetings with the client must be as productive as possible, therefore the exchange of information and the validations that are made must be agile and in a language of quick understanding for the client. An aspect that may be perceived as tedious is the documentation of the specifications and the designs that were agreed and especially the control of changes; however, it is essential to assure the sustainability of the project. This facilitates the incorporation of new people to the work team as well as granting traceability to the requirements. In addition, other exclusive phases for the mobile ecosystem were incorporated, such as: the Mobile Application Publication and Promotion. Depending on the channel chosen to distribute the application, the publication of the mobile application also follows its own process, may incur monetary expenses, time to consider for its release and specific tasks related to the metadata of the application. Likewise, it is well known in the industry that the promotion of an application plays an important role for its quick adoption, that is why we considered to include it in the life cycle, encompassing aspects that the development

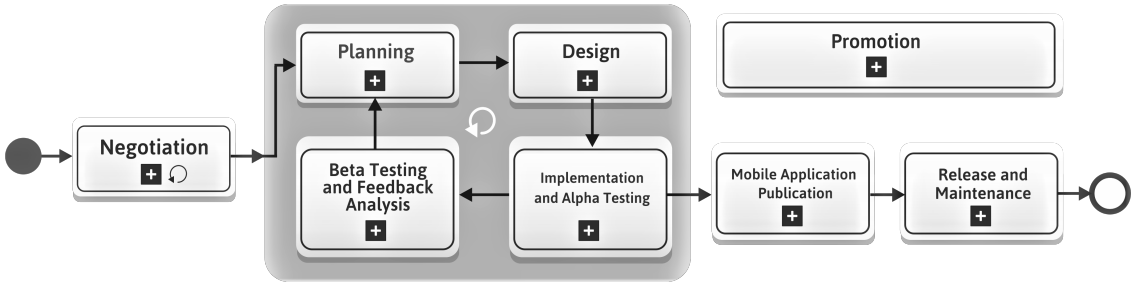

Figure 1. General outline of the phases that make up the methodology.

team can influence.

Finally, the methodological approach includes eight phases: Negotiation, Planning, Design, Implementation and Alpha Testing, Beta Testing and Feedback Analysis, Mobile Application Publication, Release and Maintenance and, lastly, Promotion, as indicated in the Figure 1.

It is important to emphasize that although in Figure 1 the Mobile Application Publication and the Release and Maintenance phases are not within the iterative cycle, its applicability will depend on the client decisions. If the client decides that the deliveries are published and made available to the users, these phases would also be part of an iterative process, otherwise, they would be the last stages once the project is finalized, with its respective connection to the Negotiation phase in case that there were new aspects to incorporate after the publication of the application.

For space limitation, a summary of the main elements that make up each phase of the proposed methodology is presented in Table 1. It includes the output elements of the eight phases along with the main activities and recommendations to obtain the best results according to the needs of the project. In addition, to define a delineation of the tasks to be carried out in each phase, tools are also suggested in each case. In order to access the complete definition of all the aspects that are involved in the described phases, interested readers may access the reference[1].

Table 1. Summary of the main elements and recommendations of the phases of the methodology.

Phase	Output elements	Principles/ Recommendations/ Best practices	Recommended tools
Negotiation	Formal budget	* Constant communication with the customer * Keep the project file updated in the Vision Document (especially when the budget was accepted)	* User stories, Formal Budget, Vision Document templates * Suggested prototyping tools * Effort estimation scheme based on two levels
Planning	Planned tasks	* Take into account the time available for the execution of the project in each incremental planning * Best Scrum practices * Requirements changes control	* XPlanner, Planigle, Assembla, among others * Starfish technique for retrospective meetings * Requirements changes control template (included in the Vision document)

Continued on next page

[1]MethApp4Mob. https://drive.google.com/open?id=0B5GfF1ckQYOmM1hiLTVTOEhuSnM

Table 1 – *Continued from previous page*

Phase	Output elements	Principles/ Recommendations/ Best practices	Recommended tools
Design	* Architecture specification * User interface design * Web services definition * Onboarding design	* Visual representation of the application architecture * MVC design patterns * Heuristics for designing a mobile application interface * Analysis of trends in user interface design according to the chosen platform * Analysis of the most common design errors * Web services with JSON format using a taxonomy for requests/responses between the application and the server * Analysis of Onboarding incorporation for retention	* Rich UML deployment diagram using tools like Gliffy, Microsoft Visio, Dia among others * Prototyping template (included in Vision document) * Spreadsheet proposed by the methodology for the web services definition * Swagger, API Blueprint, Postman as tools for implementing web services * Elasticode, Useronboard, UX Archive and EmptyStat as suggested tools for onboarding design
Implementation and alpha testing	Alpha and beta versions of the application	* Rich project structure for Android applications * Define a naming convention for the development process * Define the criteria to consider "Finished" a task * Consider aspects of the mobile environment for testing * Define the mobile testing infrastructure for tests	* Android Studio * Android project structure proposed by the methodology * Xcode * Visual Studio * Some suggested tools for functional tests are: MITE, MonkeyTal, seeTest Mobile, MobileCloud, among others
Beta testing and feedback analysis	Users feedback	* Choice of distribution channel for beta testing * In-app feedback * Add crash reports within the application * Add questionnaires such as Questionnaire for User Interaction Satisfaction * Categorization and prioritization of users feedback	* Tools like Apptentive and Doorbell.io allow you to implement in-app feedback * Appsee, Apteligent, Buddybuild, Fabric, Fireb, Instabug, among others as some suggested tools for handling crash reports
Mobile Application Publication	Published application	* Consider costs publishing applications in stores * Follow developer policies imposed by app stores	* Google Play and App Store for distribution * Other channels such as websites or emails
Release and Maintenance	Maintenance implementation	* Maintain crash reports for critical error detection * Usage analytics tools * Tracking users feedback left in application stores * Present user surveys within the application at the appropriate time * Add Net Promoter Score * Define with the client the maintenance activities after the launch of the application	* Google Analytics, Mixpanel, Flurry or Localytics are some of the usage analytics available
Promotion	Marketing plan	* Plan on time the marketing activities that make the application known to the users * Consider the App Store Optimization (ASO) suggestions * Provide a landing page for the mobile application	* Some suggested tools for the ASO are: appcodes, appnique, apptweak, AppRankCorner, among others

4. MethApp4Mob Validation

Various validations were carried out under different conditions in which complementary aspects of the methodological proposal were evaluated. For the execution of the validations, it was defined a validation protocol whose activities are summarized in Figure 2. It should be noted that some of the elements that make up the context of a validation did not apply in certain validations, therefore they were not specified.

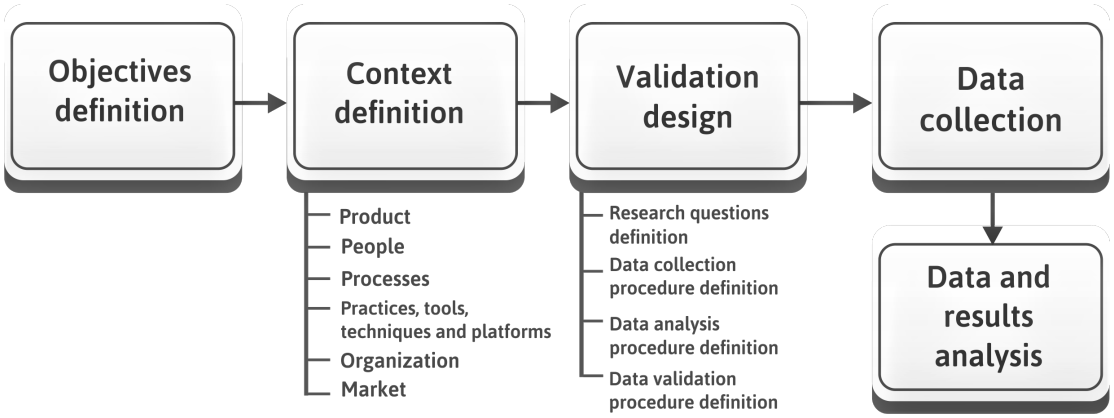

Figure 2. Activities executed in each validation summary.

In order to categorize the different executors of the validations carried out, the years' work experience in the computer science area were used as criteria: Junior, less than 2 years' experience; Semi Senior, between 2 - 6 years' experience; and, Senior, more than 6 years' experience.

Two preliminary validations were executed. The first consisted of a validation of the first 3 phases of the methodology (Negotiation, Planning and Design). The second validation consisted of an evaluation of the user stories usability in the context of the proposed methodology, that is the interest was to evaluate their usefulness according to the characteristics of the defined methodology.

Based on the results, certain adjustments were applied in order to improve some of the proposed activities in the methodological approach, always trying to make a practical proposal and easy to apply but being robust. After that, the developed exploratory study was designed and executed. For reason of space, in this paper we will present the exploratory study developed and its main results.

4.1. Exploratory study

The aim was to apply the proposed methodology to a project in the industrial field to develop a mobile application so that it serves as a guide to approach the project throughout the application development life cycle, and thus determine the impact that the methodology has about the project in order to validate its applicability. Our intent is to explore initial answers to questions such as the following: what benefits offers the methodology for the different actors?; what difficulties has the methodology imposed on the different actors at each stage of the project?; what aspects were not considered during the development of the project; how much time has been invested to understand the methodology?

The product consisted in a native mobile application for Android devices available in the Play Store under the name of ***UCApp***, intended for the students of the UC university. The main functionalities offered by the application were the visualization of the student's academic situation, registration and online payments for exams and career fees. Following the proposed methodology, the application was built iteratively and incrementally. The mobile application communicated with remote servers that simulated the academic, administrative and payment gateway systems which were implemented from scratch for this project. The data communication was made through calls to web services hosted

Table 2. Roles and profiles of the developers in the execution of the exploratory study.

Developer	Role	Profile
Developer E	Developer, Tester , Project configuration manager, Technical writer, System user	* Junior developer * No knowledge about agile methodologies * No experience developing software applications
Developer F	Project manager, System analyst, Designer, Quality assurance, Tester, Technical writer, System user	* Semi-senior developer * No knowledge about agile methodologies * Experience developing web services * Basic knowledge and experience in PHP
Developer G	Designer, Developer, Tester	* Junior developer * No knowledge about agile methodologies * No experience or knowledge developing software applications

on these servers and implemented with PHP. ***UCApp*** had three versions since its launch: Version 1.0, 20/October; Version 1.1, 10/November; and, Version 1.2, 24/November.

The exploratory study was executed by three developers (students of the final years of the Computer Engineering degree) whose roles and profiles are presented in the Table 2. The application was tested in two sessions by a group composed of 12 early-stage students from the same career. Although these students do not possess technical knowledge they were a potential public of interest for the application. The application development process was accompanied by the three researchers mentioned in the previous sections.The three developers participated in three training sessions given by one of the researcher on Android programming (front-end), PHP (back-end) and web hosting administration. The training intention was to provide developers with the necessary tools to face the various activities suggested by the methodology It is important to highlight that although the developers did not have knowledge about several of the programming languages and technologies used, they had enough theoretical knowledge about programming, database engine and other aspects of software engineering that shortened the languages and technologies learning curve. The implementation was carried out over 13 weeks during which the developers performed four iterations of the project whose activities and artifacts generated are described in Table 3.

The practices were carried out with an agile approach such as daily meetings, incremental deliveries, communication with the client and, above all, maintaining a simple, dynamic development framework that is adaptable to changes. The methodology contemplates the use of diverse tools, techniques and platforms throughout the different phases of the mobile application development process. Figure 3 presents the tools, practices, techniques and platforms used in the project specified by phase.

The data collection was carried out using questionnaires and focus group, allowing qualitative and quantitative data to be collected. The main source of data collection were the members of the development team. In order to increase the precision and strengthen the validity of this exploratory study, it was decided to apply two triangulation techniques as part of the data validation procedure: observer triangulation and methodological triangulation. The execution of the focus group was carried out by more than one observer.

Table 3. Executed activities and artifacts generated in the exploratory study.

Iteration	Activity	Artifacts generated
1º	* Functional and non-functional requirements specification through User Stories * Deployment diagram and application prototypes design * Effort estimation using six-dimensional technique	Vision Document
2º	* User stories selection and their decomposition in tasks (academic functionalities) * Web services and MVC architecture specification * Back-end and front-end prototype programming * Mobile Application Releasing in Play Store * Beta testing method and channel for obtaining user feedback definition	Prototype documentation and UCApp 1.0 in the Play Store
3º	* Similar to the previous iteration only that it included the administrative features and the payment gateway * User feedback organization and analysis * Add improvements based on users feedback	Prototype documentation and UCApp 1.1 in the Play Store
4º	* User feedback organization, analysis and implementation * Onboarding implementation * User manual preparation * Acceptance testing definition	Prototype documentation and UCApp 1.2 in the Play Store

In case of the questionnaires, *After-Scenario Questionnaire (ASQ)* was used. The ASQ method has been chosen to measure the degree of users satisfaction with the methodology proposed and executed [Lewis 1991]. Figure 4 presents the questions that make up the ASQ. Three surveys were completed that included the ASQ scenarios regarding the activities carried out in each phase and the use of certain tools and templates recommended in each phase. Different statistical graphs of the scores obtained from the ASQ were elaborated in order to evaluate the results from different perspectives. The different averages calculated are described below:

- The average ASQ score per developer per scenario
- The average ASQ score per developer in each phase
- The average ASQ score per scenario in each phase
- The average ASQ score per phase

The surveys sent to the developers could be found in the following references:

- *Negotiation Phase:* `https://goo.gl/H8pSUq`
- *Planning, Design, Implementation and Alpha Testing phases:* `https://goo.gl/gdEsG9`
- *Beta Testing and Feedback Analysis, Mobile Application Publication, Release and Maintenance phases:* `https://goo.gl/gdMj7K`

Although different statistical graphs were elaborated, due to a limited space, only the graphs that summarize the calculated average ASQ scores will be presented in the following section. At the end of each ASQ survey, a series of exploratory questions were included in order to identify those inconveniences during the phases or elements that were not considered in the proposal if they existed. A question was presented in the format Yes/No waiting for the justification as appropriate. To access the forms used and the full version of the answers obtained, consult the repository of the methodology[1].

Figure 3. Tools, practices, techniques and platforms used in the project.

After the 4^{or} iteration, the three developers and the three researchers carried out a 40 minutes focus group. This, let us identify details that might otherwise be overlooked during data collection and prevent the answers to the questionnaires from being left to the interpretation of the researchers. The researchers asked developers a series of questions related to the User Stories as a requirements gathering technique, the three-level planning, the usefulness of design diagrams, the use of JSON as a communication protocol, the main challenges during alpha and beta testing, experience with the tools used and the learning curve of the methodology.

4.1.1. Data analysis and results

The developers answered the questions based on their experience applying the methodology for the mobile application development. The analysis and data interpretation were carried out by one of the researchers and subsequently verified by two other researchers. The figure 5 describes the summary scores obtained for each developer in the different phases evaluated. It is interesting to note that the developer with the greatest expertise

After-Scenario Questionnarie (ASQ)		Strongly agree	1	2	3	4	5	6	7	Strongly disagree	NA
1	Overall, I am satisfied with the ease of completing the tasks in this scenario		o	o	o	o	o	o	o		o
2	Overall, I am satisfied with the amount of time it took to complete the tasks in this scenario		o	o	o	o	o	o	o		o
3	Overall, I am satisfied with the support information (online help, messages, documentation) when completing the tasks.		o	o	o	o	o	o	o		o

Figure 4. After-Scenario Questionnaire questions.

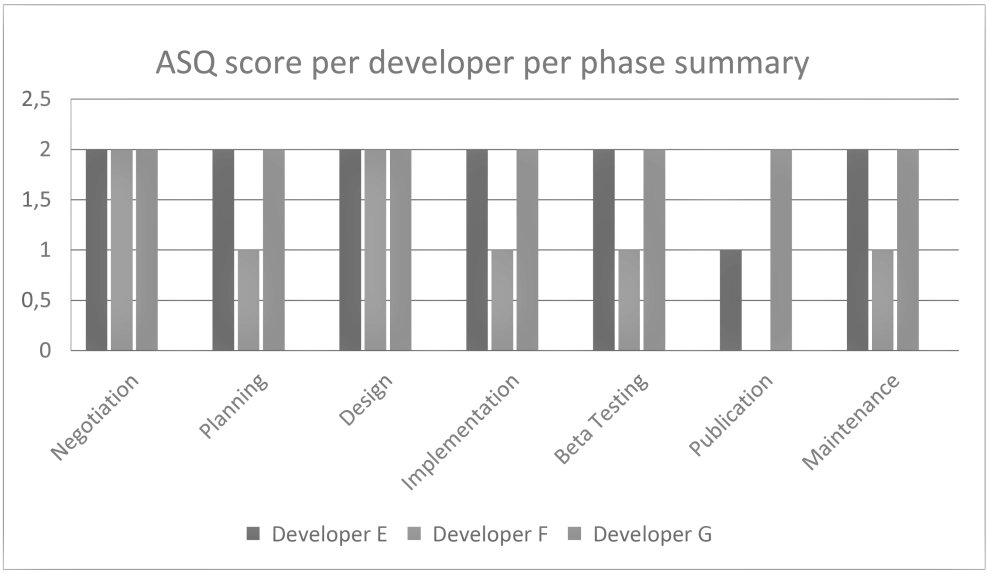

Figure 5. Average ASQ score per developer per phase summary.

(i.e., developer F) was the one who gave the best ASQ scores in most of the evaluated phases.

The usability results of the various activities, tools and techniques proposed in the Negotiation phase, such as the User Stories and the Vision document, were generally quite positive achieving an average score of 2. However, developer E and G commented that elements such as aesthetic requirement and certain fields that should be shown to the end user were not captured by the stories. In this regard it should be noted that both aspects are reflected in the prototypes presented to the client, hence the importance of accompanying the User Stories with the user interface prototypes. Likewise, during the focus group they mentioned that user stories receive significant value as a means of documenting requirements when the work team must work on maintaining the application after the project has finished. Another interesting comment of developer F indicated that it is difficult to identify all the acceptance criteria in the early stages. Also, the developer F mentioned a difficulty in the estimation using story points since they did not have knowledge of the concepts used.

The Planning phase received an average ASQ score of 1,67 (rounded to 2 in the integer scale of ASQ), which is also quite positive. No inconvenience was detected during the execution of the tasks. Despite, the use of a task control tool, suggested by MethApp4Mob, could be a great support to facilitate the project progress and the information exchange anywhere and any time, it is worth noting that the developers did not use it during the project.

The Design phase averaged an ASQ score of 2. The great utility provided by the deployment diagram was rescued for a better understanding of the project context and the prototypes designed for the Implementation phase. The proposed template for the web services definition allowed the independence between the front-end and the back-end team in the Implementation phase where this documentation was sufficient for each team to work on their own. However, developer F expressed the difficulty about the MVC architecture use without having prior knowledge.

The Implementation and Alpha Testing phase resulted in an average ASQ score of 1,67 (rounded to 2). The suggested structure for the Android projects was easy to understand even though the developers lacked prior knowledge of Android. During the focus group it was mentioned that a combination of testing approaches is necessary to cover a greater number of aspects to be controlled. The interaction tests with the interfaces were executed in real devices, since the emulators did not facilitate the user experience. However, the emulators were used instead of those devices that during the validation sessions had problems.

The Beta Testing and Feedback Analysis phase also obtained also very positive ratings with an average ASQ score of 1,67 (rounded to 2). The in-app feedback implementation was quite useful for validation with end users. This technique also facilitated the tasks of feedback organization since all the valuations were centralized in the database During the focus group, the developers indicated the importance of product validation with interested users as a way to constantly improve the quality of the final product.

With an average ASQ score of 1,5 (rounded to 2), the Mobile Application Publication phase was the best evaluated throughout the entire exploratory study. The developer F did not evaluate this phase. The documentation provided by the application store portal, in this case the Google Play Console, facilitated the task to the Applications Store Administrator role who having contact for the first time with this type of platforms.

Finally, the Release and Maintenance phase also averaged an ASQ score of 1,67 (rounded to 2). The developer E expressed the difficulty of evaluating the users' assessments to incorporate them during maintenance tasks, either due to errors in the writing or opinions too generic. In the focus group also a developer mentioned the importance of delimiting the application scope because in many cases the users requests were beyond the project scope. However, at the same time the importance of this process was valued because it allows us to know what real users of the niche market expected from the product.

4.1.2. Threats to validity

Before carrying out the exploratory study, the three developers and the researcher who mainly accompanied the developers, had not worked together. The selection of the developers team was totally arbitrary and it is fundamental to emphasize that no developer had previously knowledge about MethApp4Mob. The group of students who was responsible for testing the application is outside the researchers team. Also, the project specifications definition was made by two researchers and subsequently validated by the third researcher with the aim of ensuring that all the variables that directly affect the methodology evaluation were considered during the project definition process. In addition, several triangulation techniques were used to guarantee reliability or robustness in the validation and to reduce the uncertainty of using only one method during the work.

From the life cycle perspective, the exploratory study does not covered the Promotion phase. However, its exclusion did not affect the project development and at the end the mobile application was available for download in the Play Store.

Considering the artifacts prescribed or recommended by MethApp4Mob, the Vision document generated in the Negotiation phase included the main elements of its structure such as the requirements specification through User Stories, the application prototypes and the effort estimation through story points. Other elements such as positioning and glossary were not included. In addition, beside the methodology does not propose presenting the Gantt diagram to the client, the developers presented such a diagram which introduced some controversial feelings. Also, the methodology recommends many tools for the tasks coordination and tracking but in this case the group was composed of only three developers who were working together the most of the time, therefore the planning was more verbal.

Developers required training sessions of certain technologies and programming languages. We are aware that the training have consequences on the learning curve of the methodology and the better predisposition to evaluate it positively.

The exploratory study was born from a real need to allow academic and administrative processes to students within the UC university. The verification and testing were validated with a group of potential users from the university. Evidently, for the validation of non-functional requirements, such as usability or user experience, it is necessary for further experiences to involve a significant number of end users and adopt specific validation approaches. Moreover, a clear limitation is that the ASQ questionnaires were applied on a small group of participants (three developers). However, considering the complexity of the project, the number of developers involved in the development process was considered reasonable, respecting the established times for the presentation of the deliverables.

Finally, it is worth noting that for more exhaustive and systematic evaluations of MethApp4Mob it is necessary to carry out case studies with other types of projects oriented to mobile applications.

5. Conclusion

The need to implement solutions that support the production line of mobile software, from the creation of the idea to its delivery and maintenance, has led to MethApp4Mob,

a methodological proposal development that seeks to accelerate the mobile applications production processes and reduce operating costs linked to projects of this type of scope. MethApp4Mob covers not only the traditional technical phases of the software development but also other relevant dimensions for an industrial project as negotiation, budget preparation, feedback analysis, application publication, promotion, among others.

Two important aspects of the software production process were analyzed: those elements independent of the tools that have to do with the idea; and, those elements that depend on the tools and technologies used. We recognize the resistance of developers to the adoption of a work methodology that organizes, plans and guides the activities execution in a controlled manner throughout a software development project, for this reason, techniques and tools were recommended in each of the phases. A great effort has been made to try to maintain the balance between proposing a simple and practical approach, which reduce the learning curve of the technical staff and points to good levels of product quality, without losing formality and the need of documenting the work done during the process.

Three preliminary validations were carried out and this study focuses mainly on the exploratory study. The global results of the average ASQ scores range from 1,5 to 2 in each phase of MethApp4Mobile. It is encouraging for an approach that could be applicable to an industrial context to simplify their tasks, streamline their communications and improve the work management. However, like any methodology, we are aware that based on its adoption in industrial contexts it will undergo modifications and adaptations.

Some future works related to MethApp4Mobile are: extension of the methodology to other mobile applications types such as web or hybrid applications as well as specific domains like mobile gaming; interoperability mechanisms could be developed that improve the information exchange conditions between the tools suggested in the phases; and, obviously, to further validate the proposed methodology, we require other systematic and rigorous evaluations applying MethApp4Mobile in different industrial projects.

References

Anderson, D. J. (2010). *Kanban: Successful Evolutionary Change for Your Technology Business*. Blue Hole Press.

Beck, K. and Andres, C. (2004). *Extreme Programming Explained: Embrace Change*. Addison-Wesley, 2 edition.

D., F., S., K., and C., W. (2012). A mobile software quality model. *In: Proceedind of the 12th international conference on quality software. IEEE*, pages 154–157.

El-Kassas, W. S., Abdullah, B. A., Yousef, A. H., and Wahba, A. M. (2014). Taxonomy of cross-platform mobile applications development approaches. ELSEVIER.

Ettifouri, E. H., Rhouati, A., Berrich, J., and Bouchentouf, T. (2017). Toward a merged approach for cross-platform applications (web, mobile and desktop). *ICSDE '17: Proceedings of the 2017 International Conference on Smart Digital Environment*, page 207–213.

Galeano, I., González, M., Merín, M., and Casariego, J. (2016). Mobile application development approaches: A systematic mapping study. *2016 35th International Conference of the Chilean Computer Science Society (SCCC)*, pages 1–12.

Heredia, A., Esteban-Santiago, R., Garcia-Guzman, J., and de Amescua, A. (2014). Agile practices adapted to mass-market application development. *Journal of Software: Evolution and Process*, 26(9):818–828.

Hosbond, J. H. and Nielsen, P. A. (2005). *Designing Ubiquitous Information Environments: Socio-Technical Issues and Challenges*, chapter Mobile Systems Development: A Literature Review, pages 215–232. Springer US.

Joorabchi, M. E., Mesbah, A., and Kruchten", P. (2013). "real challenges in mobile app development". pages 15–24, Baltimore, MD. 2013 ACM / IEEE International Symposium on Empirical Software Engineering and Measurement, IEEE.

Khan, M., Azam, F., Anwar, M. W., Samea, F., and Ahmed, M. A. (2019). A model driven approach for state management in mobile applications. *ICSCA '19: Proceedings of the 2019 8th International Conference on Software and Computer Applications*, page 315–319.

Lee, C. (2014). Gartner Says Traditional Development Practices Will Fail for Mobile Apps. Technical report, Gartner.

Lewis, J. R. (1991). Psychometric evaluation of an after-scenario questionnarie for computer usability studies: The asq. *SIGCHI Bulletin*, 23(1):78–81.

O'Hagan, A. O., Coleman, G., and O'Connor, R. V. (2014). *Systems, Software and Services Process Improvement*, chapter Software Development Processes for Games: A Systematic Literature Review, pages 182–193. Springer, New York, Philadelphia.

Picco, G. P., Julienz, C., Murphy, A. L., Musolesi, M., and Roman, G.-C. (2014). Software engineering for mobility: reflecting on the past, peering into the future. *Proceeding FOSE 2014 Proceedings of the on Future of Software Engineering*, pages 13–28.

Rieger, C. (2017). Business apps with maml: A model-driven approach to process-oriented mobile app development. *SAC '17: Proceedings of the Symposium on Applied Computing*, page 1599–1606.

Scharff, C. and Verma, R. (2010). Scrum to support mobile application development projects in a just-in-time learning context. *Proceeding CHASE '10 Proceedings of the 2010 ICSE Workshop on Cooperative and Human Aspects of Software Engineering*, pages 25–31.

Schwaber, K. and Beedle, M. (2001). *Agile Software Development with Scrum*. Prentice Hall, PTR Upper Saddle River, NJ, USA, 1 edition.

Shen, M., Yang, W., Rong, G., and Shao, D. (2012). Applying agile methods to embedded software development: A systematic review.

Usman, M., Iqbal, M. Z., and Khan, M. U. (2014). A model-driven approach to generate mobile applications for multiple platforms. volume 1, pages 111–118, Jeju. 2014 21st Asia-Pacific Software Engineering Conference, IEEE.

Wasserman, A. I. (2010). Software engineering issues for mobile application development. *Proceeding FoSER '10 Proceedings of the FSE/SDP workshop on Future of software engineering research*, pages 397–400.