

An Agile and Lean Process Model for Mobile App Development: Case Study into Austrian Industry

Raoul Vallon*, Lukas Wenzel, Martin E. Brüggemann, Thomas Grechenig
Research Group for Industrial Software, Vienna University of Technology, 1040 Vienna, Austria.

* Corresponding author. Tel.: +43 1 587 21 97; email: raoul.vallon@inso.tuwien.ac.at
Manuscript submitted November 15, 2014; accepted August 17, 2015.
doi: 10.17706/jsw.10.11.1245-1264

Abstract: With recent advances in mobile technology, mobile devices have emerged from a luxury to a necessity in our daily lives. App development has been growing into a fast-paced market with volatile requirements. From a software developing point of view, responsive and adaptable software process models are advised to cope with such competitive markets. Over the last decade agile and lean methodologies have become very popular for software development in general, but there has only been limited research about the applicability of these methods in the area of mobile applications. This study aims to address the identified research gap and drafts a lean-agile process model that can compete with the requirements of today's market. Our model ALP-mobile, an agile and lean process for mobile app development, combines elements of Scrum, Kanban and eXtreme Programming (XP). We drafted our model based on previous work and pilot interviews. We then presented it to multiple mobile-app-centric companies in Austria for an initial validation of strengths and weaknesses by industry experts. Our study shows large support and potential for our model among the interviewed experts calling for further increments and practical evaluation in future research.

Key words: Agile software development, case study, extreme programming, kanban, lean, mobile app development, scrum.

1. Introduction

In 2012 the number of active smartphone users has topped one billion for the first time ever [1]. This huge number is only a small indicator for the rapid growth the mobile phone market is currently undergoing. Experts say that this number is likely to double within the next three years [1]. But not only is the mobile phone market expanding very fast at the moment. Tablets have become the new must-have gadget. Just in the 4th quarter of 2012, more than 50 million tablets were sold worldwide [2]. With the high quantity of Internet-connected mobile devices, the demand for suitable software is becoming bigger and bigger. From a software developing point of view these numbers, combined with the future outlook about the further growth of the mobile market, show the importance of mobile application development in the near future. In order to be successful in this highly competitive market, the process of developing software needs to be standardized [3]. This can be accomplished through the use of software process models and the proper execution of software development methodologies. Agile and lean software development has become very popular in recent years [4]-[6]. The principles behind those two methodologies come from agile and lean manufacturing [7].

The term *agile* was introduced in the 1990s to refer to flexible production systems [7]. With the creation

of the Agile Manifesto [8] in 2001 the principles of agile manufacturing have been adapted to software development [7]. Lean manufacturing has its origins in the 1940s, when Toyota produced automobiles with roughly half the labor hours as automakers in the US [7]. Even before Toyota [9], the concepts of Henry Ford's work [10] also relate to lean thinking but were later regarded as a special case rather than a robust lean solution [11]. The term *lean* was only introduced in the mid 1980s at MIT corresponding to production management processes and product development [11].

Although agile and lean software development is nowadays widely used, there has been comparatively little research about the applicability of agile and lean specific to the area of mobile applications. By working out the characteristics of mobile apps the effectiveness of those two software development methodologies in this area can be illustrated.

This paper offers the following contributions:

- 1) Current insight into agile and lean software development specific to the context of mobile applications. Specifically the challenges in this field of software engineering will be discussed and analyzed. Possible solutions for those challenges will be presented and recommendations will be given.
- 2) Development of an agile and lean software process model to be used to evaluate the applicability of those techniques in the field of mobile application development.
- 3) A peek into the Austrian mobile app development industry is provided.

The remainder of the paper is organized as follows. Section 2 discusses related work. The detailed research design is presented in Section 3. Section 4 analyzes experiences from our pilot case. Our agile and lean process model *ALP-mobile* is described in Section 5. Section 6 offers the full case study analysis. Section 7 discusses results with regard to related work and limitations of the study. Section 8 presents our conclusion.

2. Related Work

Lean and agile. This study aims to analyze if agile and lean software development methodologies are suited for the area of mobile application development. Perera and Fernando [12] focus on the combination of those two process models in general. Agile software development is very popular nowadays, but there are also downsides to this approach which Perera and Fernando try to overcome by combining lean practices with the agile method. They call this approach the *hybrid paradigm*. Wang [5] also investigates how agile and lean approaches have been combined in software development. For her study Wang analyzed 23 experience reports containing real-world experiences of those two methodologies. Her conclusion was that there is no "one-type-fits-all solution", but companies have the choice to combine agile and lean software development to better tailor the process to their needs.

Characteristics of mobile applications. With recent advances in mobile technology, smartphones have emerged from a luxury to a necessity in our lives, which has increased the demand for high-quality software in this area [13]-[15]. In general, there are three factors which characterize mobile app development [13]:

- Maturity of the mobile network infrastructures: Developers of mobile apps have recognized this evolution and as a result innovative software is produced to fully utilize the increasing network bandwidth which comes with new telephony communication protocols, e.g. 4G.
- Advanced mobile hardware: Mobile phone manufactures constantly try to improve the specs of the hardware they build into their devices. The main focus is hereby set on installing new displays with a higher resolution, increasing the computing power and making lighter devices along with a better battery life.

- Increasing demand for mobile applications/services: In the first quarter of 2013 smartphones outshipped feature phones for the first time ever. 216.2 million smartphones were shipped in this time period worldwide, which marks a 51.6% share of total phone shipments [16]. According to Canalys this percentage will further increase to an estimated share of 73% (1.5 billion smartphone shipments) in 2017 [17].

With the advances in technology the mobile application landscape is rapidly evolving leading to behavior described by Scharff and Verma: “The industry follows a strict time to market requirement orchestrated by a fierce competition where standards are not stable.” [18]. To survive in the mobile market a suitable process model is needed.

In 2005, Abrahamsson considered agile methods to be the best fit for mobile applications [19]. Abrahamsson regarded the following agile characteristics as applicable to mobile software: high environment volatility, small development teams, object-oriented development environment, non-safety critical software, application level software, small systems and short development cycles. Only the agile characteristic identifiable customer does not match with the corresponding characteristic of mobile software, since the end-users in the distribution of mobile applications are potentially unlimited. Nevertheless there are also areas in mobile application development in which the developers are able to identify their customer, e.g. a netbanking app will only be interesting for customers of the respective bank.

In [20] Corral *et al.* focus on changed characteristics in mobile software compared to the characteristics Abrahamsson stated in his publication nearly a decade ago. They identified the following changes:

- Abrahamsson writes about the high uncertainty of mobile software which comes from the fact that hundreds of new smartphones are introduced each year. Today the amount of new smartphones each year is still very high, but the uncertainty which comes with this fact is decreased due to the fact that only three main operating systems dominate the market, i.e. Android, iOS and Windows Phone. Those systems all have a solid development kit, which makes it easier to develop software for new devices. Nevertheless in contrast to the opinion of Corral *et al.*, we consider the high heterogeneity of mobile devices a problem in mobile app development nowadays, because the operating system cannot be optimized for the device itself. Especially Android features different versions of their operating system which run on all kinds of different devices manufactured by different companies. All these factors still bring an uncertainty to mobile app development which cannot be ignored.
- Regarding the team size not that much has changed in mobile application development. Mobile software is still produced by small- to mid-sized teams, but today mobile software is often part of large development projects by large companies and organizations.
- Mobile applications have become more complex over the years. This results in the fact that mobile software is not necessarily a stand-alone application any more. Mobile applications nowadays interact with other systems, use network and hardware resources more heavily, apply collaboration tools and so forth. This makes mobile applications by definition not small any more.
- The agile characteristic for non-safety critical software collides with mobile software produced these days. According to [20] there are applications like “healthcare monitors, mobile banking or earthquake alerts that are required to meet strict standards to enter into service and cannot be categorized as noncritical software”. In general, due to the fact that mobile applications have become more and more complex the software produced for mobile devices faces new requirements like, for example safety issues.

New processes for mobile systems must therefore adapt to the changes in the field of software development. The first approach to apply agile methodology to mobile application development was proposed in 2004. Abrahamsson *et al.* introduced an agile development approach called *Mobile-D* which tries to overcome the challenges in mobile app development [21]. Four years later Jeong *et al.* presented *MASAM* (Mobile Application Software Development based on Agile Methodology) in [22] which is very

similar to Mobile-D. Also in 2008, Rahimian and Ramsin based their research on the combination of plan-based and agile methodologies for the production of mobile software systems in [23]. The outcome was the *Hybrid Methodology Design Process*. In 2010, Scharff and Verma [18] analyzed the use of Scrum for the development of mobile applications in a class setting at Pace University involving a professional Scrum Master. In 2011, Cunha et al. presented in [24] *Scrum Lean Six Sigma (SLeSS)* which is an integration approach of Scrum and lean six sigma. They claimed that this approach “enables the achievement of performance and quality targets, progressively improving the development process and the outcome of projects” [24].

Corral et al. focus in [20] their research on the agile development models for mobile application development described within this section. They conclude that mobile software engineering still faces extensive work to determine which processes and practices are the best fit for this field of software development.

Abrahamsson discusses the applicability of agile software development in the area of mobile information systems in [25]. Due to the rapid growth of the mobile market and the constant advance of technologies in this field, the mobile application market is highly competitive, uncertain and dynamic. Restrictions in screen size, memory and battery power are not that big anymore and the processing power of smartphones and tablets is already as good as the power of regular computers. So the limitations in the mobile market are getting smaller, which opens the door for new advances in mobile application development.

In [26] Wasserman articulates software engineering issues related to the development of mobile applications. He analyzes the characteristics of mobile applications in general and further raises possible research topics related to mobile app development. He concludes that there are still a large number of complex issues in this area of software engineering which need to be addressed in further work.

3. Research Design

Following our research of related work regarding the usage of lean and agile processes in mobile application development, we were able to identify the following research gap that we define as the objective of this study:

As mobile application development is a rapidly changing field of software engineering, the objective of this research is to improve and update the knowledge of the execution of agile and lean methodologies in the field of mobile app development. To this end, this study aims to explore the current usage of agile and lean within the Austrian mobile application development industry.

Following the recommendation of Verner *et al.* [27], we defined the objective of the study before choosing a suitable research method. Based on our objective and the literature review we classified this research as an exploratory case study [28] as it aims to explore a subject area where only little current research exists. The research question (RQ) of this study is:

RQ: How can agile and lean processes be implemented to cope with currently evolving problems in the development of mobile applications?

Based on this research question, research propositions are defined to direct attention to topics which should be examined within the scope of this study [28]:

PR1: Organizations use agile methodology because it is a good fit for small projects.

The first proposition is concerned with the reason why organizations use agile processes in their projects. As pointed out by Abrahamsson in [19], ideal agile characteristics are, among others, small development teams and small systems. Although mobile software has become more and more complex over the years, app projects are still small compared to big, traditional software projects. Therefore with this proposition we strive to analyze if organizations focusing on mobile application development use agile or lean

processes at all, and what the reasons for implementing these particular processes are.

PR2: *Mobile application development works well with short iterations due to potentially smaller scope and cycle times.*

Short iterations are one practice of lean software development defined in [29]. We are interested if organizations in the field of mobile application development agree with this practice.

PR3: *Automated testing and continuous integration are not common in mobile testing.*

The third proposition targets the area of testing in mobile app development. Based on the literature research performed in the beginning of this study the topics of automated testing and continuous integration are not popular in this field of software development. In [26], Wasserman points out that testing is an important area for mobile software engineering research. Kim states in [30] that practices like test-driven development and continuous integration are either unknown or difficult to use. As a consequence the majority of apps are tested manually.

PR4: *Early prototyping is very important in mobile app development.*

Ben Morris *et al.* define in [31] early prototyping as an important activity in the development of a mobile application. The purpose of this proposition is to examine the opinion of the experts regarding this topic.

PR5: *Mobile application development requires a high degree of collaboration with the customer.*

The last proposition concerns the collaboration with the customer in mobile app development. Customer collaboration is an important concept in both agile and lean software development. One of the four values proposed in the Agile Manifesto [8] is “Customer Collaboration over Contract Negotiation”. In [29] Hibbs *et al.* define customer participation as one practice of lean software development. By defining this proposition we want to investigate the importance of this concept in the field of mobile software development.

In our research, the subjects of analysis are organizations which focus their services on the development of mobile applications. Choosing such organizations offers great insights into the processes associated with the development of mobile applications. Furthermore through the opinion of experts the applicability of our developed process model (Section 5) in this field of software development can be determined.

The data collection performed in the course of this case study is based on conducting qualitative focused interviews [28]. First-hand experience with the development of mobile software was the prerequisite for the selection of the participating experts of the interviews. This approach allowed us to ask open questions about many topics based on our propositions, but still follow a declared-in-advance protocol. The interviews lasted one hour on average.

In order to find experts for the interview we performed an Internet search about organizations located in Austria which focus their services on the development of mobile software.

Verner *et al.* recommend in [27] to undertake a pilot study before running the real cases. Concerning the selection of the pilot case Yin states in [28] that convenience, access and geographic proximity are the main criteria for selecting the pilot case. We conducted two pilot cases to develop a lean-agile process model called *ALP-mobile* (an Agile and Lean Process for mobile application development) tailored to the needs of mobile application development using input from literature review and the pilot cases. The goal was to present the model to the participants of the full case study and investigate which parts of the model they can relate to and to which they cannot. This should serve to stimulate discussion in the interview and also be a first practical validation of the model that is subject to further • incremental improvement. Figure 1 shows the phases of this study.

4. Analysis of Pilot Cases

In general pilot case studies help researchers to check if the defined research propositions are meaningful and if the questions asked during the interviews focus on these propositions [27].

The organizations Alpha and Beta were selected as pilot cases since no cold-calling approach was necessary due to previous contact. These pilot interviews were used as additional input to ALP-mobile, i.e. the process model was not completely finished by the time the pilot interviews were conducted. Therefore, no references to ALP-mobile were made during the course of these interviews.

The organization Alpha is specialized on the development of native mobile applications for Android and iOS. Alpha develops both projects for specific customers and implements its own products. The interview was conducted with the CEO of Alpha. Besides his activity as CEO, the expert is responsible for a clean and structured project management within the organization. Alpha has ten employees, nine of which work in the development department. Concerning processes in the organization, there are differences between projects for customers and the development of own products. This is due to the fact that within customer projects the budget is clearly defined, which changes the development process of Alpha, because new features must be communicated to the customer, which decreases the agility in the development.

The organization Beta works in the health sector and they develop apps for diabetics. They focus on developing their own products, so there is no customer role available. The representative of Beta, who attended the interview, was their CTO. In this role, the person is responsible for the successful development of the products. The organization has seven developers and 25 employees in total, some of which only work part-time. Due to the fact that Beta is ISO 13485¹ certified, which is needed to develop products for the health sector, they must document their processes to prove the quality standards that are needed for this specific sector. Therefore software processes are a very important concern for Beta.

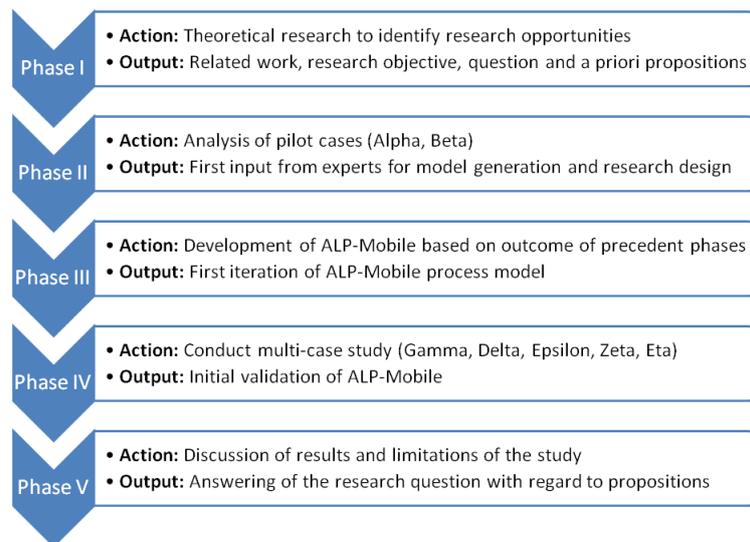


Fig. 1. The five phases of this study.

4.1. Implications of Pilot Study

In the following section we present the main results of the two pilot interviews and its implications on the refinement of ALP-mobile.

The first topic in the analysis of the pilot study concerned agile processes in mobile app development in general. The CEO of Alpha pointed out that there is a difference between customer projects and the development of own products. According to the CEO it is a lot harder to implement agile processes in customer projects, because the development highly depends on the customer. Because of this fact ALP-mobile is designed for the use in customer projects. Since this approach is more difficult, ALP-mobile

¹ http://www.iso.org/iso/catalogue_detail?csnumber=36786

can also be used for the development of own products with only minor adaptations.

The experts further stated that agile processes in general align very well with the development of mobile applications, but because of certain specific restrictions, such as the obligatory review process by app stores during the deployment procedure, it is sometimes hard to follow agile methodologies in this field of software development. The deployment procedure can be addressed by ALP-mobile by adding “queue” columns to the value stream board. The following quotes summarize the attitude towards agile software development in the two pilot organizations, respectively:

“Let’s take a retail app which is located and sold in the app store: After the first release I get a lot of support requests that features are missing or that errors occur. Then I resolve those issues in sprints or iterations as in classical agile software projects. Totally makes sense.” (CEO, Alpha)

“[Using agile] Makes perfect sense. However, I have had the experience that one could get lost using the agile approach when developing a product where the result is not entirely clear. Where I do not make a product that simply copies another product or where I do not go a specific path that is already clearly foreseeable, but where I’m more in an experimental stage and I don’t exactly know how the final product will look like. You just start to develop, implement some stuff and then you discover that it should have been different. Especially in the mobile area, in particular with iOS apps. With web services I can make an update at any time in principle and thus deploy minimal changes several times a day if I want to. This means that I can be agile, not only in development but also in the deployment. This is different with mobile apps. With iOS for example I have a weeklong review process. Therefore I have to plan accordingly. The idea of small improvements in agile development is a lot harder to implement in such an environment, compared to web applications for example.” (CTO, Beta)

Regarding the topic of prototyping the interviewees described that they use low-fidelity and high-fidelity prototypes. Especially high-fidelity prototypes, e.g. click-through mock-ups, provide a very rich user experience, which is great to demonstrate functionalities of the future application. Concerning ALP-mobile we therefore pick up on the activity of early prototyping in the process model. According to the two experts intra-team feedback mechanisms are an important part of their process implementations, which is reflected in ALP-mobile.

Table 1. Interview Structure Developed after Conduction of Pilot Studies

Topics	Description
1. Process models in use	- which process models are used in the organizations - what are the reasons for the use of these models
2. Mobile app development	- in what way is the development of mobile apps unique in the field of software development
3. ALP-mobile	Short presentation of ALP-mobile process model
4. Roles	
5. Meetings	
6. Requirements Engineering	- what is the approach to [topic 4-9] in your current process
7. Development Practices	- what is your opinion on the approach to [topic 4-9] in ALP-mobile
8. Testing	
9. Distribution & Maintenance	

The last topic in the analysis of the pilot study concerned the testing approaches executed in the organizations. Our a priori proposition perfectly matched with the current situation in mobile app development in the two pilot cases. Often apps are tested solely manually, but concepts like test automation and continuous integration are desirable. The implication on ALP-mobile is the introduced testing concept

in which manual testing is prescribed and the concepts of automated testing and continuous integration are preferable but not mandatory.

The following interview structure was developed for the actual multi-case study based on experiences from the two pilot case interviews (cf. Table 1).

5. An Agile and Lean Process Model: ALP-Mobile

In order to stimulate discussion for the interviews, we developed a process model designed to meet the characteristics identified during literature review and pilot interviews. The workflow of the model is illustrated in Fig. 2. The model is described by running through the whole workflow as follows.

ALP-mobile focuses on mobile software projects which are developed for a specific customer. Therefore details concerning the project are discussed with the customer in a preliminary meeting. After this preliminary meeting the prescribed kick-off meeting is held. The kick-off meeting includes the determination of the technology stack which is associated with the activity of defining the development approach.

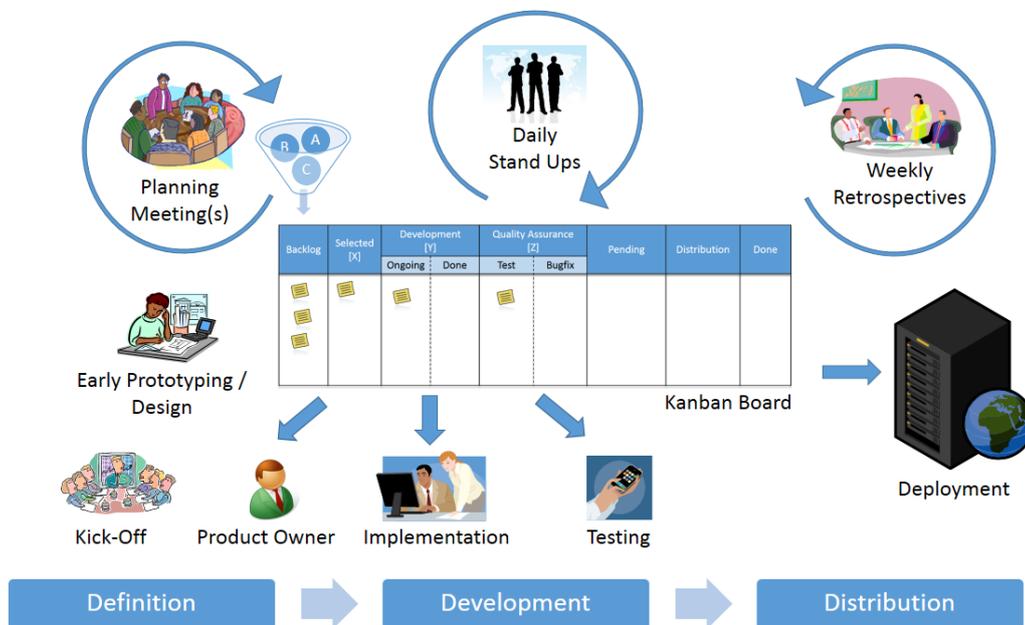


Fig. 2. The ALP-mobile process workflow.

After the kick-off meeting the main focus is put on requirements engineering to lay out the initial groundwork for the development of ALP-mobile. The product owner defines user stories according to the wishes of the customer and furthermore schedules the planning meeting in which the user stories are estimated. Early prototyping is strongly encouraged in ALP-mobile, since this task leads to valuable feedback from the customer early in the project, as has also been noted in the pilot study. Since ALP-mobile is an iterative agile and lean based process, not all user stories need to be, or even can be, defined in the beginning of the project. This flexible approach is necessary in mobile application development because it is a quickly evolving fast-paced field of software engineering. At least the initial set of user stories should be defined so that the team is ready to start with the implementation of software. Communication is essential in all activities of ALP-mobile. The close cooperation with the customer fits the lean practice of customer participation.

For development, a lean-style kanban board in ALP-mobile provides a transparent mapping and status report of the value stream. Hence it is a central element in the process. We propose the following steps on

the board which can be tailored to each specific project environment. The *Backlog* column is filled with the (estimated) user stories and the development team is ready to implement the first work items from the *Selected* column on the kanban board. The *Selected* column is maintained by the product owner according to the priority of the user stories, which is based on the business value and the technical feasibility of the stories. The process is now controlled via the WiP (Work in Progress) limits on the kanban board, which limit how much work packages can be taken on at the same time for a step. This lean concept enforces a pull-approach and a “go see” mentality, i.e. the work item gets pulled to the next step once resources are free and bottlenecks should be fixed by all parties. The overall goal is to establish a continuous flow. After a work item is implemented and tested, it can get pulled into the *Pending* column of the board. The sum of the stories in that column represents the current version of the software prior to the first distribution. The product owner is responsible for keeping the user stories up to date and to request a new planning meeting if new user stories need to be estimated. The iteration of the planning meeting and the ongoing changes in the backlog represent a clear distinction to traditional software process models, in which completed phases are not executed again.

In agreement with the customer a version of the software is then distributed. After the distribution the work items can get pulled into the *Done* column. As stated, the process is not finished with the first distribution of the software. The enhanced flexibility which is established through the execution of a process without fixed iterations makes ALP-mobile very efficient and productive in the very short-lived mobile market.

As illustrated in Fig. 2, ALP-mobile suggests ongoing retrospective meetings in order to review the development of the product and the process itself, which should encourage making changes to the process, such as adjusting WiP limits. This meeting is scheduled weekly. Also a Scrum-style daily standup-meeting is advised with an upper limit of 15 minutes to facilitate intra-team coordination.

To sum up, ALP-mobile has iterative elements (such as agile plannings, daily standups and retrospectives) without the need for fixed iterations as in lean software development. New stories can be added to the product at any given time. The activities in each phase provide a guideline of tasks to be executed in order to finish a mobile project successfully. Concerning the current nature of mobile app deployment and distribution in the app markets (which may dictate review phases), we argue that agility is decreased in the distribution phase in comparison to the precedent ones, where the process can make full use of both agile and lean elements.

6. Case Analysis

This section presents the multi-case study analysis. Table 2 gives an overview of the respective interview partner of each case.

Table 2. Overview of Interview Partners: Alpha and Beta have been Addressed in Section 4 for the Pilot Study

Topics	case type	Size	Expert's Role
Alpha	pilot case	10	CEO
Beta	pilot case	25	CTO
Gamma	multi-case study	26	CEO
Delta	multi-case study	9	Solution Manager
Epsilon	multi-case study	15	CTO
Zeta	multi-case study	5	Android Tech Lead
Eta	multi-case study	8	CEO

The organization Gamma focuses on the development of apps and on the realization of customer projects,

but also offers consulting services. At the moment they have 26 employees, 14 of which are developers. The remaining staff works in quality assurance, design and project management. Gamma develops apps for iOS, Android and Windows Phone. Furthermore they support the following platforms: HTML5, Blackberry and Unity. The interview was conducted with the CEO of Gamma. The expert has 10 years of experience in the IT sector, six years of which in the area of mobile development. Gamma applies a lightweight version of Scrum.

Delta is an IT provider in the area of Banking and Finance. The interviewed expert has eight years of experience in software development. As solution manager he focuses on requirements engineering and is further responsible for the coordination and the lead in the mobile projects. Delta currently uses Scrum as a process model. Regarding the team size, the mobile department currently has three solution managers, five developers and one tester.

Epsilon focuses its services on digital communication and mobile app development. As a full-service agency they offer services in consulting, design, implementation and marketing of digital solutions. The interview was conducted with the CTO of Epsilon. As technical operator the expert is responsible for everything associated to the development of software in the organization. At the time of the interview the organization had 15 employees with two thirds being developers (some of which freelancers). The rest of the personnel are involved with sales and project management. Epsilon does not follow any strict process models, but focuses on the independence of their developers, as they are only involved in small-scale projects and regard a full-blown process model as an overhead.

Zeta develops software for medium and large enterprises. The interview was held with the Android tech lead in the mobile department of the organization. The mobile department has five full-time employees and if necessary additionally up to three part-time freelancers. Zeta started with Scrum „by the book“ and then adapted the process to their needs. Especially the iteration length of 2-4 weeks recommended in Scrum was regarded as too long, which is why they work with shorter iterations.

Eta is a full-service IT agency and was founded in 2014. Their product portfolio covers the whole lifecycle of developing app- and web-based projects. At the time of the interview, the organization was comprised of two permanent employees and six freelancers. The interview was conducted with the CEO of the organization. Eta does not follow any strict agile process model, but they try to go an agile way during the implementation phase.

6.1. Cross-Case Analysis

Table 3 provides a compact overview of the five cases. A cross-case analysis of each of the studied categories based on the interview protocol (cf. Table 1) follows.

Process Models. Most of the experts stated that they use agile processes within their organization. According to the CEO of Gamma they make use of a lightweight version of Scrum. This approach is similar to the approach executed at Zeta. The expert at Zeta pointed out that they have difficulties to follow Scrum in the complete form. The solution manager of Delta stated that they made use of the waterfall process model in the beginning but now they use Scrum as a process model. The CEO of Eta explained that they do not make use of any particular process model. They follow the classic process, but in the phases they try to work relatively agile. A complete different approach was proposed by the CTO of Epsilon who said that they do not have any real processes in the organization due to small-size projects for which an implementation of a specific process model would be an unnecessary overhead.

Concerning the use of process models in general we conclude that there is no perfect solution which satisfies every organization. The interviews showed that agile respectively lean processes are a good fit for mobile application development, which is the first indicator that ALP-mobile can bring real value to this field of software development. In the end it always depends on the organization respectively on the projects

which processes are the best solution. Another interesting fact was that the process models used were not implemented as prescribed, but rather adapted to the specific needs of the project teams, which aligns with ALP-mobile.

Table 3. Overview of the Cases Studied in the Multiple-Case Study

	Gamma	Delta	Epsilon	Zeta	Eta
Process Models	Agile (Scrum)	Agile (Scrum)	No defined process; Projects too small	Agile (Scrum)	Traditional Agile in phases
Differences in Mobile App Development	Technological environment and markets	Increased need of flexibility	Small tasks for which it is hard to define standardized processes	Fast pace of the mobile market	Decreased cycle time of mobile applications
Roles	Sales, Project Manager, Designer, Developer, Tester	Solution Manager, Project Manager, Developer, Tester	Project Leader, Developer	Product Owner, Scrum Master, Dev Team	PO / Coach, Dev Team
Meetings	Kick-off, Dailies, Weeklies	As in Scrum without Dailies (only on demand)	Kick-off, otherwise on demand	As in ALP-mobile, but planning before kick-off	Weeklies
Requirements Engineering	Highly depends on the experiences of the customer	Based on early prototyping	Highly depends on the customer	Requirements are clarified with the customer prior to creation of user stories and estimation	Essential to clearly determine the desired functionalities
Development	Development similar to ALP-mobile	Kanban board important for process execution	Company uses framework for hybrid apps	Development similar to ALP-mobile	No iterations in the development is essential
Testing Distribution	Manually Review process for iOS is hard to plan	Manually Provided Betas of the OSs are very helpful to release updates soon	Manually Review process with iOS is tedious, one needs to plan accordingly	Manually Distribution for iOS much more complex than for Android	Manually 2 weeks for app store- review need to be planned for an iOS app

Mobile App Development. In general, the information gathered on the differences between mobile application development and software development shows that the interviewees especially see the increased need of flexibility in the development as a major distinction. The mobile market is very fast-paced and therefore mobile software must constantly be improved. The ongoing development regarding mobile technology also brings changes to the implementation of mobile apps, since the products must be updated to new environment conditions like changing screen sizes or new versions of the operating system.

Roles. The Android tech lead of Zeta stated that the roles in ALP-mobile are very similar to the roles they have in their organization. Besides the fact that they use other names for the roles, the approach regarding the roles in the organization Gamma is also comparable to ALP-mobile. According to the CEO of Eta they combine the roles of the product owner and the agile coach into one role because the organization was just newly founded. The CTO of Epsilon described that they have the role of the project leader which is comparable to the product owner prescribed in ALP-mobile. Besides that role they have developers who implement the mobile applications. The situation in the organization Delta is a different one: They have the role of the solution manager which is mainly responsible for the activity of requirements engineering. In

most projects they also have the role of the project manager which is responsible for the budget of the project. Besides those roles they have developers and testers.

The usage of such process models including all artifacts highly depends on the organizations and on the projects they carry out. The investigation of the roles implemented in the organizations show exactly that result. Limited resources respectively process overhead are reasons why organizations customize process models as they see fit.

Meetings. Similar to the roles in the organizations the established meetings also depend on the available resources. According to the CEO of Gamma their approach to meetings is very similar to the approach in ALP-mobile. They have a kick-off meeting, daily stand ups and a weekly meeting which combines estimation, review and retrospective. The interviewee of Zeta pointed out that the concept regarding the meetings prescribed in ALP-mobile is almost the same as in the mobile department of Zeta. The only difference is that the first planning meeting is executed before the actual kick-off meeting of the project. The organization Delta tries to implement the meetings as advised in Scrum. A difference to the meetings in Scrum is the fact that they do not perform daily meetings. Instead those meetings are held on an on-demand basis. The on-demand approach is also the approach executed in the organization Epsilon. The CTO of Epsilon prefers that developers directly ask if they have problems instead of waiting for the next meeting to discuss this issue. The CEO of Eta stated that for them daily stand ups would be an unnecessary overhead since the size of the organization is not appropriate. They have a weekly meeting which is the only scheduled meeting at the moment.

The conclusion regarding the roles in the organizations can be mapped to the meetings without further adjustments. It depends on the organizations and on the projects which meetings they establish respectively which meetings are reasonable.

Requirements Engineering. The Android tech lead of Zeta stated that the requirements engineering already happens before the actual kick-off meeting of the project. According to the interviewed expert of Delta the requirements engineering is based on an early prototype of the application. In both cases, as in ALP-mobile, the outcome of the requirements engineering is a set of user stories which describe the software. Concerning user stories the organization Gamma takes another approach. The CEO of Gamma stated that they do not write user stories in their projects, because he sees the creation as unreasonable since the developers already know the target group as well as the functionalities of the app. Therefore their focus is set directly on the features. The CTO of Epsilon pointed out that the knowledge of the customer regarding mobile applications defines this activity. The result of this activity is a requirements document. According to the CEO of Eta the task of requirements engineering is really important, because only with a profound requirements engineering the customer satisfaction at the end of the project can be ensured.

Concerning the activity of requirements engineering, all experts shared the same view. Each of the interviewees pointed out that this activity is an important part of their project lifecycle. Another similarity which was encountered was that this activity highly depends on the customer's knowledge.

Development. Regarding the concept of no fixed iterations in the development in ALP-mobile, the CEO of Eta pointed out that this concept is inevitable in mobile development. For him the iteration length proposed in Scrum is not applicable in this field of software development, because usually the projects are small and therefore the flexibility is lost with long iterations. The Android tech lead of Zeta stated that due to the fact that they have weekly iterations in their process, which is an adaptation to the usual sprint iterations advised in Scrum, they are already more flexible in the development. A similar view regarding this subject was expressed by the CEO of Gamma. He described that the process in the organization Gamma is very similar to the process executed at Zeta. They have weekly iterations, which highly increases the flexibility. In essence they have almost the same approach as proposed in ALP-mobile since even the goals for the

weekly iterations are not fixed, which means that work items can be added to the sprint backlog on demand. The solution manager of Delta expressed a different view, noting that if changes to sprint backlogs with iterations of one week are necessary then there are probably problems in the sprint planning respectively the requirements engineering. Because of the fact that the organization Epsilon does not make use of any particular process models, we did not ask the CTO of Epsilon about his opinion regarding the concept of no fixed iterations in the development.

The experts basically shared the same opinion: Iterations of two to four weeks as prescribed in Scrum are too long in mobile application development, since the flexibility in the development is lost. Therefore processes must be adapted to the fast pace of the mobile market.

Testing. Concerning the testing of mobile applications the organization Gamma has a dedicated testing team which is responsible for testing. Regarding automated testing the CEO of Gamma said that due to the fact that they implement a lot of customized solutions the overhead for the creation of automated tests would be unreasonable. The testing process executed at the organization Eta starts with unit tests on the bottom and user tests at the top of the application. Concerning the size of the project the CEO of Eta also mentioned that in small mobile projects the concepts of automated testing and continuous integration are probably an unnecessary overhead. The solution manager of Delta stated that testing is performed manually. According to the CTO of Epsilon the testing in the organization Epsilon is mainly performed by developers who are assigned to test a specific application. Besides testing performed by the developers and the customer, the product owner tests the application in course of the weekly meetings at Zeta. The Android tech lead of Zeta confirmed that concepts like automated testing and continuous integration are very interesting especially with applications which have a longer lifetime than small mobile apps.

All the cases clearly displayed the current situation in mobile application testing: Concepts like automated testing and continuous integration are not common. Nevertheless with the increase of complexity of mobile software those concepts will become more interesting in the future.

Distribution. The last topic in the cross-case analysis of the focused interviews concerns the distribution and the maintenance of the software. Regarding the distribution of the mobile application the Android tech lead of Zeta pointed out that iOS is much more complex in the distribution than Android. According to the CEO of Gamma the necessary review process with iOS is hard to plan in the distribution of the software. He pointed out that it is important to communicate this fact to the customer in order to avoid any problems with the schedule of the project. The CEO of Eta also mentioned that for iOS you have to plan one up to two weeks where you cannot do anything, because you have to wait for the deployment of the application. Therefore you need to be very careful in the development, since updates take very long. Regarding the review process for iOS prior to the distribution in the app store the CTO of Epsilon pointed out that it is a bit tedious, but in the end you just have to plan the project accordingly. Concerning new versions of the operating systems the solution manager of Delta noted that beta versions of the operating system offered by the operators are very helpful in order to release an update of the application soon after the release of the new operating system.

Based on the operating system, all experts shared the same opinion. iOS is much more complex compared to Android when it comes to the distribution of the application. The necessary review process with iOS must also be considered in the schedule of a project. Nevertheless the impact is not that high, because the organizations are used to this situation. The experts also stated that it is important to communicate these dependences with the customer.

7. Discussion

Our a priori research propositions presented in Section 3 will be examined in detail, which leads to

answering the research question. Subsequently we also investigate related work in order to point out differences of ALP-mobile to other process models. For that purpose, we focus on the feedback regarding ALP-mobile gathered in the course of the interviews in our multi-case study. By analyzing the feedback gathered, an initial validation of strengths and weaknesses of ALP-mobile can be derived. The end of the section will address limitations of the study.

We stated five research propositions which are related to the research question of this study to direct attention to specific areas within the scope of this study:

PR1: Organizations use agile methodology because it is a good fit for small projects.

As a result of the focused interviews we especially see an increased need for flexibility in mobile application development as the main argument for the use of agile processes in this field of software development. As experienced in the case study sometimes the mobile projects are even too small for the use of such processes and therefore the implementation of a proper process model is seen as unreasonable. Nevertheless, as noted in the literature, smaller projects are in theory a good fit for agile processes, but the main reason why organizations use agile methodologies in mobile app development is the increased need for flexibility in the development.

PR2: Mobile application development works well with short iterations due to potentially smaller scope and cycle times.

This proposition was confirmed by the majority of the interviewees. As pointed out in the data analysis of the case study, the experts noted that short iterations help to adapt to the wishes of the customer. This fact also relates to the aforementioned need of increased flexibility in mobile app development.

PR3: Automated testing and continuous integration are not common in mobile testing.

In the course of the focused interviews the perception that automation in mobile software testing is not common was corroborated. As the main reason for this situation the interviewees stated that the overhead for the creation of an automated test system is unreasonable in most of the cases. This is due to the fact that a lot of apps have a strong focus on the user interface which is hard to test with automated test cases. Additionally the experts pointed out that features are often highly customized, which increases the effort of automated testing since the tests cannot be reused within other projects. Therefore mobile software is still mainly tested manually.

Nevertheless some of the interviewees said that it would be great to have an automated testing process within their projects. Especially for changes late in the project, such tests are very useful in order to check all functionalities of the software automatically.

Based on the interviews conducted within the case study we conclude that concepts concerning automated testing of mobile applications will become more important as the complexity of mobile software increases. For simple apps, which mainly focus on the visualization of information, the overhead of such concepts may be unreasonable, but, similar to software development in general, more complex applications depend on activities like automated testing and continuous integration in order to ensure the high quality of the software which is necessary to be successful in the highly competitive mobile market.

PR4: Early prototyping is very important in mobile app development.

The interviewees stated that they make heavy use of prototypes early in the development. The solution manager of Delta explained that a prototype of the application is the basis for every project proposal. That means that the prototyping of the app already happens before the actual kick-off in this organization. Other experts pointed out that they made good experiences with clickable mockups of the application. Such prototypes help to visualize the vision of an application and therefore feedback can be gathered early in the project. Especially the design and the functionality of the software can be addressed at a very early stage.

A disadvantage of this activity is the fact that clickable mockups already provide such a great user

experience so that customers sometimes do not understand why it takes quite a long time to actually develop the software:

“We put strong focus on hi-fi mockup prototype development. That means that the final prototypes are very close to the final product. The problem then is that many believe that the product is already finished after they have seen the prototype. We struggle with that, for example if we present a nice-looking prototype and we still have another half a year until we can go live.” (Solution Manager, Delta)

Besides this disadvantage the experts did not mention any problems with that concept. To the contrary, the interviewees confirmed that early prototyping is very important in mobile app development.

PR5: *Mobile application development requires a high degree of collaboration with the customer.*

The opinion of the interviewees regarding this topic was very clear. Customer collaboration is very important in mobile application development. We see the importance on the user interface in mobile software as the main reason for the importance of customer collaboration. The experts also pointed out that especially in mobile app development the customer is often closely integrated in the development of the software which means that he can define the functionalities respectively the design of the app over relatively long periods in the project lifecycle. This fact implies high communication with the customer, as it is the only way to ensure the high agility in the development.

Following our case study analysis we strive to answer our research question:

RQ: *How can agile and lean processes be implemented to cope with currently evolving problems in the development of mobile applications?*

In order to answer this question we proposed an agile and lean process model for mobile app development called ALP-mobile. This model is on the one hand based on an extensive literature research in relevant areas and on the other hand on practical experiences gathered in course of the pilot interviews performed within the case study. In addition to this pilot interview we conducted focused interviews with Austrian experts in the field of mobile application development. By presenting ALP-mobile to the interviewees we were able to derive strengths and weaknesses of this process model. As a part of the interviews the interviewees were asked about their opinion regarding ALP-mobile. In the following we present quotes concerning ALP-mobile.

The CEO of Eta especially emphasizes the importance of the kanban board since it scales well. Nevertheless he pointed out that because of the current size of the organization the idea of an extra coach is not reasonable at the moment:

“I think the model is quite good. We probably will not afford a coach besides the product owner for a long time, because it makes no sense. But I think especially the board is a pretty good thing, because it is very flexible. It probably will scale easily. There is no difference if you have limits of one and two on the board, because you only have two developers and for example only five stories in the backlog or if there are limits of seven or eight and you have 25 stories in the backlog. The concept remains the same.” (CEO, Eta)

The solution manager of Delta also had a positive impression of ALP-mobile:

“It gives the impression that you were trying to combine Scrum and kanban - I think it's good. Concerning the role of the coach I do not see the big difference to the Scrum master. What I really liked is your approach of the dailies. That you concentrate on the perspective of the stories and not on the perspective of the people who implement it.” (Solution Manager, Delta)

The size of the organization or the projects, respectively, is very important for the decision which process model to use. The CTO of Epsilon pointed out that his organization mainly develops marketing apps which have a strong focus on data visualization. Therefore he sees the execution of a proper process model as an unnecessary overhead for their projects, but he further states that for more complex projects ALP-mobile would be a good fit:

“The purpose of our apps is to draw attention to the customer. I would rather see your process in another type of app such as an app for employees of insurance companies [...]. These are more complex things, where such a process model, I think, would be optimal.” (CTO, Epsilon)

The Android tech lead of Zeta already sees a lot of similarities in ALP-mobile to the process model executed within their organization. The only aspect to which he did not agree with the concept of ALP-mobile is the fact that the first estimation of the user stories happens after the initial kick-off meeting. Besides that he did not mention any problems to be expected when using ALP-mobile in practice:

“We use a lot of Scrum. The kanban approach we follow implicitly. [...] Concerning the roles we are also much the same. There's nothing unfamiliar. Parts were really very similar and therefore good. The only point that would not work for us is that the story points are estimated after the kick-off. For us the kick-off is only after the project has been confirmed and for that we already need the estimation of the effort for the proposal. But this is probably the only point that does not work in the real world - at least for us. It's different when the project is internally and simply uses its own manpower. Since I do not need to estimate in advance, that works better.” (Android Tech Lead, Zeta)

The CEO of Gamma also liked the concept of ALP-mobile in general. A good point for him is that there are no iterations, which increases the flexibility in the development of the software. He pointed out that, although they have weekly iterations, their process model is very similar compared to ALP-mobile regarding this issue.

To conclude this section we present ALP-mobile as the answer to the initial research question on how agile and lean processes can be adapted to mobile app software development. ALP-mobile combines concepts from the agile respectively lean world and furthermore focuses on the fast pace of the mobile market. The focused interviews in course of the case study provided positive initial feedback, which can be seen as an indicator that ALP-mobile is capable of bringing real benefits to organizations working in the field of mobile application development. Nevertheless practical experiences with ALP-mobile are needed in the future to evaluate whether this process model can be executed successfully in industry.

7.1. Strengths and Weaknesses

Based on the displayed quotes in this section the first impression of the experts regarding ALP-mobile was very positive. The interviewees in particular liked the increased flexibility which is a result of the use of kanban as a tool for the process execution in which fixed iterations in the development are waived in order to create a steady flow of work. The kanban board as the main source of information was another positive concept introduced in ALP-mobile. Especially the ability to use the board for small as well as for large projects makes it very interesting for a variety of teams. Due to the fact that ALP-mobile is a combination of popular process models the learning curve should not be too high, which helps to decrease the overhead of the initial implementation of ALP-mobile.

Concerning the weaknesses of ALP-mobile, the experts did not mention any big disadvantages of the model at this point. Nevertheless they pointed out that some concepts would not work at their organization. A reason for that is the fact that ALP-mobile is mainly based on an extensive literature research and therefore limitations in the real world are possible. Another reason for this situation is that process models always need to be adapted to the settings of the organizations which implement a particular model. This is also true for the execution of ALP-mobile.

7.2. Comparison to Related Work

Corral *et al.* [20] concentrated on process models for mobile application development and investigated their impact in real environments. Since the introduction of *Mobile-D* [21] in 2004 the characteristics of mobile software have changed. Nevertheless this approach is the only model which has received

considerable support in real world settings [20]. Table 4 presents a summary of documented implementations of each methodology as reported in [20]. The presented numbers show a need for more empirical evidence concerning those methodologies in order to answer the question whether they are applicable in practice or not.

Table 4. Agile-Based Mobile Software Development Processes and their Reported Case Count [20]

Methodology	Year	Case Studies	Cited by
Mobile-D	2004	16	17
MASAM	2008	0	3
Hybrid	2008	0	9
Scrum	2010	1	4
SLeSS	2011	1	1

In the following we compare each of the methodologies to ALP-mobile. The first model in Table 4 is *Mobile-D*. As stated, Mobile-D was first introduced nearly a decade ago. Mobile software has changed since then and therefore Mobile-D does not focus on those changed characteristics. Nevertheless this model has already been used in large industry settings [20] and therefore Mobile-D is probably the most prominent agile representative in mobile application development. In contrast to ALP-mobile, Mobile-D does not make use of lean concepts such as a kanban board. A possible reason for that could be that lean software development was not popular yet.

The next methodology presented in the table is *MASAM* [22] which has a strong tie to Mobile-D. There is no case study with evidence of the execution of this model in a real environment. Similar to Mobile-D, MASAM also does not introduce any lean principles into the process.

Rahimian and Ramsin take a different attempt compared to ALP-mobile by combining plan-based and agile methodologies into a framework for the development of mobile applications. Again this hybrid approach [23] does not make use of any lean concepts and there is also no case study evidence of the implementation of this approach in a real world setting.

In [18] Scharff and Verma make use of Scrum for the development of mobile applications in a student course setting. Since ALP-mobile makes use of concepts of Scrum, this approach shows similarities to the concepts introduced into ALP-mobile. Nevertheless it also lacks the input of lean principles.

SLeSS [24] is also a combination of agile and plan-based methodologies. In case of SLeSS it is a combination of Scrum and lean six sigma. The case study involves a real world project but the model has not been very popular [20].

The discussion with regard to related models shows that ALP-mobile takes a unique and different approach and is the latest installment in the legacy of process models for mobile app development by combining agile and lean elements.

7.3. Limitations

Our case study provided valuable insights into real projects in the field of mobile application development. As an empirical study, several limitations must be noted. A limitation is that focused interviews were used as the single source of evidence and results may thus be subject to interviewee bias. It is advised to use multiple sources of evidence in case study research in order to corroborate the findings [28]. Our approach was exploratory trying and aimed at several small to medium-sized Austrian app development organizations. Future research should involve further iterations of ALP-mobile and include a practical evaluation of the model if feasible. As a case study, generalizability of results is limited in light of its limitations. Our results should be regarded as an initial validation and a basis for further research.

8. Conclusion

The highly competitive mobile market requires standardized processes in order to be successful in this fast-paced environment. Therefore we proposed a new process model called ALP-mobile which is a combination of agile and lean methodologies to overcome the challenges in today's mobile application development. ALP-mobile covers the complete project lifecycle in mobile application development including (incremental) definition as well as implementation and testing of the product, bringing the app to its desired market and providing proper maintenance of the software in order to ensure high quality.

In order to overcome the challenges in mobile application development we combined concepts from Scrum, XP and kanban. To increase the flexibility in the development kanban was used as a tool for the process execution. The kanban board was introduced as the main source of information in ALP-mobile. It visualizes the workflow and provides transparency by means of WIP limits directly attached to each stage in the process. ALP-mobile furthermore makes use of four different types of meetings: a kick-off (once), planning (on demand), standups (daily) and retrospective (weekly).

In our case study, ALP-mobile was introduced to Austrian experts in the field of mobile application development. The overall feedback regarding the presented concepts was very positive. Especially the increased flexibility in the development was regarded as essential in the field of mobile application development. Nevertheless the focused interviews showed that process models in general and also ALP-mobile in particular need to be tailored to an organization's or even project's individual constraints. This procedure is facilitated by implementing regular retrospective meetings and the introduction of an agile coach.

Based on the initial feedback from experts in Austrian industry, the concepts introduced in ALP-mobile have the potential to bring great benefits to organizations working in the field of mobile application development. In order to corroborate the findings and the results of this study ALP-mobile must be the subject of future research in which empirical evidence of the practical usage of this process model is collected and analyzed.

Acknowledgements

This research was partly funded by the Austrian Marshall Plan Foundation (<http://www.marshallplan.at>) by providing a scholarship to the first author. The report has been finalized during a research visit at the Center for Design Research at Stanford University. The first author thanks Professor Larry Leifer for hosting the research visit.

References

- [1] Dover, S. Study: Number of smartphone users tops 1 billion. Retrieved, 2012, from <http://www.cbsnews.com/news/study-number-of-smartphone-users-tops-1-billion>
- [2] Johnson, J. 52.5 million tablets sold in Q4 2012, record sales included 22.9 million ipads. Retrieved, 2012, from <http://www.inquisitr.com/503134/52-5-million-tablets-sold-in-q4-2012-record-sales-included-22-9-million-ipads/>, last accessed: 11/14/2014
- [3] Gao, Y., & Yao, X. (2010). The two approaches to sustainable development of the theory of software process models. *Proceedings of the 3rd International Conference on Information Management, Innovation Management and Industrial Engineering* (pp. 537-540).
- [4] Mishra, A., & Mishra, D. (2011). A curriculum for agile software development methodologies. *ACM SIGSOFT Softw. Eng. Notes*, 36(3), 1-2.
- [5] Wang, X. (2010). The combination of agile and lean in software development: An experience report analysis. *Proceedings of the IEEE Agile Conference* (pp. 1-9).

- [6] Leffingwell, D. (2010). *Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise*. Addison-Wesley Professional.
- [7] Poppendieck, M., & Cusumano, M. (2012). Lean software development: A tutorial. *IEEE Softw.*, 29(5), 26-32.
- [8] Beck, K. *et al.* Agile manifesto. Retrieved, 2011, from <http://agilemanifesto.org/>
- [9] Ohno, T. (1988). *Toyota Production System*. Productivity Press.
- [10] Ford, H., & Crowther, S. (1922). *My Life and Work*. Garden City Publishing Company.
- [11] Womack, J. P., Jones, D. T., & Roos, D. (1990). *The Machine That Changed the World: The Story of Lean Production*. Rawson Associates.
- [12] Perera, G. I. U. S., & Fernando, M. S. D. (2007). Enhanced agile software development — Hybrid paradigm with LEAN practice. *Proceedings of the International Conference on Industrial and Information Systems* (pp. 239-244).
- [13] Gasimov, A., Tan, C. H., Phang, C. W., & Sutanto, J. (2010). Visiting mobile application development: What, how and where. *Proceedings of the 2010 Ninth International Conference on Mobile Business and 2010 Ninth Global Mobility Roundtable* (pp. 74-81).
- [14] Nosseir, A., Flood, D., Harrison, R., & Ibrahim, O. (2012). Mobile development process spiral. *Proceedings of the 7th International Conference on Computer Engineering and Systems* (pp. 281-286).
- [15] Palmieri, M., Singh, I., & Cicchetti, A. (2012). Comparison of cross-platform mobile development tools. *Proceedings of the 16th International Conference on Intelligence in Next Generation Networks* (pp. 179-186).
- [16] Restivo, K., Llamas, R. T., & Shirer, M. *More Smartphones Were Shipped in Q1 2013 Than Feature Phones, An Industry First According to IDC*. Retrieved, 2013, from <http://www.businesswire.com/news/home/20130425006953/en/Smartphones-Shipped-Q1-2013-Feature-Phones-Industry>
- [17] Canals. Over 1 billion android-based smart phones to ship in 2017. Retrieved, 2013, from <http://www.canalys.com/newsroom/over-1-billion-android-based-smart-phones-ship-2017>
- [18] Scharff, C., & Verma, R. (2010). Scrum to support mobile application development projects in a just-in-time learning context. *Proceedings of the ACM 2010 ICSE Workshop on Cooperative and Human Aspects of Software Engineering* (pp. 25-31).
- [19] Abrahamsson, P. (2005). Keynote: Mobile software development — The business opportunity of today. *Proceedings of the International Conference on Software Development* (pp. 20-23).
- [20] Corral, L., Sillitti, A., & Succi, G. (2013). Agile software development processes for mobile systems: Accomplishment, evidence and evolution. *Proceedings of the Mobile Web Information Systems* (pp. 90-106).
- [21] Abrahamsson, P. *et al.* (2004). Mobile-D: An agile approach for mobile application development. *Proceedings of the 19th Annual ACM SIGPLAN Conference on Object-Oriented Programming Systems, Languages, and Applications* (pp. 174-175).
- [22] Jeong, Y. J., Lee, J. H., & Shin, G. S. (2008). Development process of mobile application SW based on agile methodology. *Proceedings of the 10th IEEE International Conference on Advanced Communication Technology* (pp. 362-366).
- [23] Rahimian, V., & Ramsin, R. (2008). Designing an agile methodology for mobile software development: A hybrid method engineering approach. *Proceedings of the 2nd IEEE International Conference on Research Challenges in Information Science* (pp. 337-342).
- [24] Da, C. T. F. V., Dantas, V. L. L. Andrade, & R. M. (2011). SLeSS: A scrum and lean six sigma integration approach for the development of software customization for mobile phones. *Proceedings of the 25th*

Brazilian Symposium on Software Engineering (pp. 283-292).

- [25] Abrahamsson, P. (2007). Agile software development of mobile information systems. *Advanced Information Systems Engineering* (pp. 1-4).
- [26] Wasserman, A. I. (2010). Software engineering issues for mobile application development. *Proceedings of the Workshop on Future of Software Engineering Research* (pp. 397-400).
- [27] Verner, J. M., Sampson, J., Tomic, V., Bakar, N. A. A., & Kitchenham, B. A. (2009). Guidelines for industrially-based multiple case studies in software engineering. *Proceedings of the 3rd IEEE International Conference on Research Challenges in Information Science* (pp. 313-324).
- [28] Yin, R. K. (2009). *Case Study Research: Design and Methods*. Sage publications.
- [29] Hibbs, C., Jewett, S., & Sullivan, M. (2009). *The Art of Lean Software Development: A Practical and Incremental Approach*. O'Reilly.
- [30] Kim, H. K. (2013). Test driven mobile applications development. *World Congress on Engineering and Computer Science*.
- [31] Morris, B., Bortenschlager, M., Lansdell, J., Luo, C., & Somerville, M. (2010). *An Introduction to Bada: A Developer's Guide*. Wiley.

Raoul Vallon received his M.Sc. in business informatics in 2011 and the M.Sc. in software engineering in 2012 (both with honors) from Vienna University of Technology, where he currently serves as the operational head of AMMA, the Work Group on empirical software engineering, at the Research Group for Industrial Software (INSO). In 2010 he participated in the European master in software engineering program at Universidad Politécnica de Madrid. In 2014 during his PhD studies, he was a visiting researcher at professor Larry Leifer's center for Design Research of Stanford University. He has several years experience working in industry in both technical and management positions.

Lukas Wenzel received his master's degree in business informatics at the Vienna University of Technology in 2014. During his studies he served as a teaching assistant for several computer science courses. His research interests currently largely focus on mobile application development and process management. Besides his research at the Vienna University of Technology he gathered extensive web programming experience working for technical organizations in Austria. Additionally, Mr. Wenzel has in-depth knowledge in the conceptualization and programming of innovative information visualization platforms and smart web applications.

Martin E. Brüggemann received his M.A. in 2010 from RWTH Aachen University, Germany, and is currently pursuing his PhD studies at the Research Group for Industrial Software (INSO) at the Vienna University of Technology. His research focus is system dynamics in agile and lean processes. He has several years of working experience in project management and consulting in Germany, the Netherlands and Austria.

Thomas Grechenig is a senior architect in large IT systems and nation-wide IT-infrastructure. He is a professor for industrial software engineering and the head of the Research Group for Industrial Software (INSO) at the Vienna University of Technology. He and his teams have planned, designed, and built several large scale IT solutions, e.g. in health, railway systems or public transport applications. In science and research the focus of interest goes towards enhancing the stability and fine-tuning of the IT-mass-concept in all its critical sectors (usability, security, IT-infrastructure, performance, integration and interoperability)