



Agile methods for mobile application development: A comparative analysis

Mudasir M Kirmani

Assistant Professor, Computer Science

Sher-e-Kashmir University of Agricultural Sciences & Technology of Kashmir
Srinagar, J&K, India

Abstract: Agile software development methods have caught the attention of software engineers and researchers worldwide. The comparative analysis is performed using the methods life-cycle coverage and empirical evidence as the analytical lenses. The result show that the agile software development methods cover certain/different phases of the software development life cycle and most of them don't offer adequate support for project universal solution and empirical evidences is still very limited. Based on the results new directions are needed and to place emphasis on methodological quality not method quantity. The aim of this paper is to organize, analyze and make sense out of the dispersed field of agile software development methods. Based on the results of the analysis, practitioners are in a better position to understand the various properties of each method and make their judgment in a more informed way. For this purpose an analytical framework is constructed which guides the analysis of the different existing methods.

INTRODUCTION

The development speed in mobile software development has become a key factor due to developer's possibility of submitting applications (app's) directly to the market. Thus, it is necessary to identify best practices to implement mobile applications as well to provide a good learning experience.

As mobile platforms continue to improve in performance, the users need for a wide variety of mobile applications increases. The mobile application development market has been dramatically growing in the last few years the complexity of its application and speed of software development process. These changes in the mobile development market require a re-thinking on the way the software development should be performed by teams. Mobile software development is the process of creating software which can be used by small, low power handheld devices such as mobile phones. This software is either factory pre-installed applications on the mobile phones or downloaded from app stores and mobile software distribution platforms. The development of mobile application is challenging due to rapidly changing business requirements. The development of mobile applications and system should satisfy special requirements and constraints and thus differs from traditional software development in many aspects. Thus as the use of methodologies is recognized in theory as starting point in dealing with stated challenges, issues and specifics and as existing methodologies don't provide enough merits for use in this field.

According to Wasserman [1], mobile devices have been adopted in different ways for desktop or laptop computers. Mobile applications development can be similar to software engineering for other embedded applications. However, mobile applications development present some additional requirements that are less commonly found if compared to traditional software applications

The use of methodology is recognized in theory as starting point in dealing with stated challenges, issues and specifics. When discussing the development of mobile applications. The first issue that should be addressed is the usage of

methodology (Rahimian and Ramsin,2008; spataru 2010)[12-13] Classic or agile software development methodologies should be adapted for the development of mobile applications as the existing ones don't cover the specific mobile targeted requirements. There are several attempts from different authors to create new methodologies in order to cover the gaps in the domain of mobile applications. Some of them are Agile Risk-based methodology (Rahimian and Ramsin,2008)[12], MASAM (Jeong et al., 2008) , and Mobile -D (Abrahamsson et al., 2004)[14].

Process model selection for developing a software depends on the software project characteristics. For mobile application development, we need to understand the characteristics of mobile application. Main characteristics of mobile application are:

- 1- User requirements are changing with time, change is frequent.
- 2- Development time is short.
- 3- Emphasis is more on user interface.

Keeping these requirements in mind, a flexible software process models that can adjust to frequently changing system requirements in short duration with low budget is what is needed for mobile application development. The uncertainty in mobile application development thus requires an iterative and incremental model. Agile is identified as the most suitable process model.

RELATED RESEARCH

A new development methodology , specifically tailored for mobile application development called Mobile-D. is presented in (Abrahamsson,et al.,2004)[2]. The method is based on agile practices, drawing elements from well established agile methods such as XP and Crystal methodologies but also from the heavier Rational Unified Process.

According to (Beydeda, Book and Gruhn, 2005)[3], Model Driven development involves using models not to document code but to serve as a basis for application development.

In (Balagtas Fernandez and Hussman, 2008)[4], the authors propose a development approach that combines both Mobile Driven Development(MDD) and Human-computer interaction(HCI) by making process end –users to create their own applications and by making the process user-friendly through user centered design.

In (carton,et al.,2007)[5], propose a development approach that combines Aspect-oriented Software Development(AOSD) techniques with Mobile –Driven Development ones.

In (Braun and Eckhaus,2008)[6], MDD is used to develop an architecture that supports the provision of mobile service both as a web service and mobile application. The goal was to allow the provided services to be accessed both via built ion XHTML browser and pre installed Java application.

In (Khambati,el al.,2008)[7], MDD approach has been documented for the development of mobile personal health care applications.

A different approach is presented in (Rahimian and Ramsin,2008)[8]. The author use a methodology engineering approach called Hybrid Method Engineering to generate method suitable for mobile application development.

AGILE DEVELOPMENT

Agile methodology follows an iterative approach to build software rapidly where the entire software development lifecycle is broken down into smaller iterations which helps to minimize overall risks, allows the project to adapt to modifications rapidly, doesn't demand a requirements freeze upfront and maintains the project schedule and budget. Some of the well known existing agile software – development methods are : XP, SCRUM, LEAN, CRYSTAL, Feature Driven Development(FDD), AUP, ASD, Dynamic system development method(DSDM), KANBAN, AM.

Agile development or adaptive development are aimed to rapidly adapt to the changing reality. The adaptation of agile makes software process more flexible helps in continue learning and incremental delivery, quickly and easily adapting to requirements and technologies changes. Agile development is recommended to small to medium sized projects.

Even though agile methodologies offer a good solution for mobile application development, different approaches exist in a literature

In literature Extreme programming (XP) and SCRUM are the most common agile methods for Mobile application development.

Scrum: It is an iterative and incremental agile software development approach. It offers a framework and set of practices that keep engineering visible, allowing

practitioners to know exactly what is going on and to make adjustments in order to have the project moving towards desired goals. The scrum workflow is a sequence of iterations called sprints which have duration between one and four weeks each. Each sprint has daily meetings where each team member answers what he/she has done on the previous day, what is going to be in the current day and if there is any roadblock to move forward on development activities. At the end of each sprint there is a product demo called Sprint Review and after that it is handled a lessons learned session called Sprint Retrospective (Reichlmayr,2011)[8].As per majority of survey participants scrum is found to be the most effective mobile application because it offers daily stand-up meetings. This meeting is daily for maximum 15 minutes and provides all time project overview with details of impediments, to-do and completed task list. Main strength f scrum model is high project management capability. Scrum is a combination of generic project management practices and lacks system development cycle (SDLC) phases about engineering of a software. For this purpose, practicing XP engineering practices is essential to support scrum principles.

Extreme Programming (XP): It is the most suitable for the development of mobile application mainly due to Behavior-Driven Development (BDD), Test Driven Development (TDD) and Continuous Delivery (CI) because one won't be agile without them.

TDD is important for designing before coding and test automation, very important when delivery every sprint and wanting tp break what was working in previous sprints and CI makes sure that nothing breaks.

-Pair Programming: XP programmers write all production in pairs, two programmers working together at one machine. One is coding and thinking of the tactics of implementing the method strategically fits into other parts of the project.

-Test Driven Development: XP team focuses on validation of the software at all times. Programmers develop software by writing test first then checks its correctness, and then software that fulfills the requirements reflected in the tests. Customers provide acceptance test that enable them to be certain that the features they need are provided. Pair programming makes it easier for developers to adapt TDD.

According to Moore and Flannery (Moore and Flannery, 2007)[9]. XP implements a groupware style development where feedback is obtained by daily testing the software where developers deliver the system to the customers as early as possible, allowing a rapid response for requirements and technologies changes. It is a discipline of software development which emphasizes productivity, flexibility, informality, team work and the limited use of technology outside of programming working in short cycles and every cycle starts by choosing a sub set of requirements from a larger set (Macias et al.,2003)[10]. The main strength of XP are fast development, cost saving, high satisfaction of client, test driven development resulting in less errors and acceptance of changing requirements.

Comparison of key Methodologies in Agile:

Documentation	-XP, Scrum , ASD(Agile software development) and crystal don't put a lot of emphasis on documentation and minimum documents are created. -Teams following FDD spend sufficient amount of time in documentation. -DSDM requires some documents to be created and degree of documentation is less than that of FDD and more than that of XP.
End-User Involvement	-End user is actively involved in XP. -Product owners represent end-users in scrum. -In ASD, DSDM and Crystal end-users participate in all of the incremental releases. -In FDD, end users participate through reports.
Team meetings	-Informal daily stand up meetings happen in Scrum. And XP. -Information sharing is through documents in FDD and DSDM. -Face to Face meetings happen in Crystal and ASD.
Size of projects	-While Scrum, DSDM ,FDD and Crystal can be followed for projects of any size, methods like ASD and XP are only followed for smaller projects.
Sprint cycle	-DSDM advocates producing 80% of solution in 20% of the time. -XP has a sprint cycle varying from 1 week to 6 weeks. -Scrum has 2 to 4 weeks. -ASD has 4 o 5 weeks. - FDD can have a smaller sprint cycle too (2 days)and it can vary up to 2 weeks(based on the project requirements.

Basic characteristic features, advantages and gaps of agile methods:

1. Method Title: Extreme programming

Model overview: It is the most suitable for the development of mobile application mainly due to Behavior-Driven Development (BDD), Test Driven Development (TDD) and Continuous Delivery(CD) because one won't be agile without them. It is collection of well known software engineering practices [5,19,20]. Some of the main characteristics of XP are short with small iterations, rapid feedback, close customer participation, constant communication, pair programming etc.

Advantages: End users are actively involved in the process of software development. Hence the product development is very close to what customer wants.

-Team feedback is also taken very seriously and there is a lot of focus on self-improvement.

-Best practices are well defined and religiously followed by the teams.

-It encourages team-work.

-Test-cases are easy to understand.

-The process of development can be visualized and measured.

-Fast development, cost saving, high satisfaction of client.

-Test driven development resulting in less errors and acceptance of changing requirements.

Gaps: Documentation is given very less emphasis and hence giving reference to an issue/ instance in past (while working on projects) is a challenge.

-At times it is difficult to bring in customer into the team since he is very distant from the development team. Hence some other team member plays the role of the customer. Thus lack of discipline observed at times.

2. Method Title : Scrum

Model overview: The Scrum has been developed for managing the software development process in a volatile environment. It is an empirical approach based on

flexibility, adaptability and productivity. Scrum leaves open for the developers to choose the specific software development techniques, methods and practices for the implementation process.

Advantages : It encourages to embrace changes.

-Scrum is "time boxed", helps with estimation and is relatively light weight.

-It is well known , easily understood and many people have it done before.

-It adopts changes quickly and has short turn around since mobile projects change so quickly.

-with shorter sprints it provides quick feedback and more opportunities to build the application.

-Helps to find out bugs easily at early stage that way one can deliver value faster and also come up with changes in market technology.

-High project management capability.

Gaps: Poorly documented and hence it is too easy to be abused.

-The lack of deadline allows customers to ask for more functionalities.

-If the team is not seriously self involved in the project the later may fail.

-The model is recommended to be used only for small and fast projects.

-If a member leaves the team; this can massively damage the project.

-Requests must be well defined in order to draw.

-Documentation is complex and time consuming.

3. Method Title: DSDM

Model overview: It is fully agile software development method. DSDM is a method developed by a dedicated consortium in the UK. The first release of the method was in 1994. The fundamental idea behind DSDM is that instead of fixing the amount of functionality in a product and then adjusting time and resources to reach the functionality, it is preferred to fix the time and resources and then adjust the

amount of functionality accordingly. The origins of DSDM are in RAD.

Advantages: -Allows for efficient project management and strong control on project life cycle.

-Requirement priority approach helpful in delivering most important functionalities first.

Gaps: Documentation is complex and time consuming.

4. Method Title: FDD (Feature Driven Development)

Model overview: FDD is a process-oriented software development method for developing business critical systems.

-The FDD focuses on the design and building phases.

-It emphasizes quality aspects throughout the process and includes frequent and tangible deliveries along with accurate monitoring of the progress of the project.

Advantages: Multi-tasking is possible in case of FDD.

-Methodology can be used in case of applications that are complex because of documentation and reports that are created.

Gaps: Complexity is so much that there is no point in using this method for smaller projects.

-Less communication within and out of team. Thus teams learn less from other individuals and teams

5. Method Title : Crystal.

Model overview: The Crystal methodology is one of the most lightweight, adaptable approaches to software development. Crystal is actually comprised of a family of agile methodologies such as Crystal Clear, Crystal Yellow, Crystal Orange and others, whose unique characteristics are driven by several factors such as team size, system criticality, and project priorities. Crystal focuses on six primary aspects: people, interaction, community, communication, skills, and talents. Process is considered secondary.

Advantages: High risk and highly important component are delivered first.

-Effective team communication and this facilitates learning amongst team members from each other.

-This methodology can be adjusted as per project type and team size.

Gaps: The planning and development are not depending on requirements; hence traceability is an issue in crystal.

- Only limited scalability
- Lack of an unambiguous common process
- Limited applicability: not suitable for developing highly critical systems

- Over-dependence on inter-human communication

5. Method Title: Lean Software Development (LSD)

Model overview: The focus of lean is continual learning and improvement along with reducing waste. LSD applies lean manufacturing principles to application development such as using value stream concept and leveraging tactics, sticking adhesive notes on board to communicate the next needed task, much like Kanban cards is used in factories to manage materials.

-Therefore Lean software development principles enable to implement Scrum and Kanban principles effectively.

Advantages: LSD philosophy eliminate anything that does not add value to customer or waste such as unnecessary code and functionality, delay in software development process, unclear requirements, insufficient testing and slow internal communication.

-The creation of software is continuous learning with continual customer feedback so the team as a whole is improving through out the delivery cycle.

Gaps:

6. Method Title: Adaptive Software Development

Model overview: ASD attempts to bring about new way of seeing the software development in an organization, promoting an adaptive paradigm.

-This method encourages incremental and iterative development with constant prototyping.

-One ancestor of ASD is "Radical Software Development.

Advantages: It offers a solution for the development of large and complex systems.

Gaps: ASD doesn't provide a framework with enough guidance to prevent projects from falling into chaos, which could suppress emergence and creativity.

7. Method Title: Agile modeling.

Model overview: AM, attempts to adopt agile modeling practices using agile philosophy as its backbone. The underlying idea is to encourage developers to produce sufficiently advanced models to support accurate design, needs and documentation purposes. The aim is to keep documentation and amount models as low as possible.

Advantages:

- Agile methodology has an adaptive approach which is able to respond to the changing requirements of the clients
- Direct communication and constant feedback from customer representative leave no space for any guesswork in the system

Gaps:

- This methodology focuses on working software rather than documentation, hence it may result in a lack of documentation

- The software development project can get off track if the customer is not very clear about the final outcome of his project

8. Method Title: Internet Speed Development:

Model overview: ISD is least known approach to agile software development. ISD refers to a situation where software needs to be released fast, thereby requiring short development cycles.

ISD puts forth a descriptive, management-oriented framework for addressing the problem of handling fast releases. This framework consists of time drivers quality dependencies and process adjustments.

Advantages: The framework for ISD is considered as more management and business-oriented than other related approaches. ISD draws from the "synch and stabilize" approach by Microsoft, aimed at copying with a fast-moving or even chaotic, software development business. ISD is a collection of random, opportunities processes driven by accident. These processes are simultaneous, overlapping and there are gaps and the development itself occurs in completely unique and idiographic forms.

9. Method Title : Pragmatic Programming.

Model overview: PP, introduces a set of programming "best practices". It puts forward techniques that concretely argument the practices discussed in the other agile methods.

-PP covers programmatic practicalities. The method itself is a collection of short ups that focus on day-to-day problems. These practices are pragmatic perspective and place focus

on incremental, iterative development, rigorous testing and user centered design.

10. Method Title: Rational unified process methodology.

Model overview: It is shortly known as an RUP is a one modern software development process. This methodology divides the development process into four distinct phases that each involves business modeling, analysis and design, implementation, testing, and deployment. This is an object-oriented and web-enabled program development methodology. This model also helps software developer for providing them guidelines, templates, and examples for all aspects and stages of software development.

Advantages:

- This methodology emphasizes on accurate documentation
- It is proactively able to resolve the project risks that are associated with the clients evolving requirements for careful changes and request management
- Very less need for integration as the process of integration goes on throughout the development process

Gaps:

- The software developer needs to be expert in their work to develop software under this methodology.
- The development process in this methodology is very complex and not exactly organized.
- Integration throughout the process of software development adds the confusion that causes more issues during the stages of testing.
- This process is too complex therefore it is very hard to understand.

11.Method Title: Rapid application development

Model overview: RAD is an effective methodology to provide much quicker development and higher-quality results than those achieved with the other software development methodologies. It is designed in such a way that, it easily take the maximum advantages of the software development. The main objective of this methodology is to accelerate the entire software development process. The goal is easily achievable because it allows active user participation in the development process.

Advantages:

- Rapid Application development model helps to reduce the risk and required efforts on the part of the software developer

- This model also helps client’s to take quick reviews for the project
- This methodology encourages customer feedback which always provides improvement scope for any software development project

Gaps:

- This model depends on the strong team and individual performances for clearly identifying the exact requirement of the business
- It only works on systems that can be modularized can be built using this methodology
- This approach demands highly skilled developers and designer’s team which may not be possible for every organization
- This method is not applicable for the developer to use in small budget projects as a cost of modeling and automated code generation is very high

COMPARATIVE ANALYSIS OF EXISTING AGILE METHODS

Fig:2 shows that different agile methods are focused on different aspects of the software development life cycle. DSDM is an independent method in the sense that it attempts to provide complete support over all life cycle phases.

The internet-speed development approach also addresses all the phases of the software development life-cycle but only at managerial level. ASD covers all other phases except for project inception, acceptance test and system in use. AM aims at providing modeling support for requirements specification and design phases.

The crystal family covers the phases from design to integration test. XP, PP, FDD and Scrum are focused on requirements specification, design, implementation (except for scrum) and testing up until the system test.

From the process perspective AM, ISD, SCRUM (for the implementation part) and PP approaches don’t emphasize (or have not described) the process through which the software development proceeds. AM and PP are supplements to other methods. In case of AM and PP the lack of the process perspective seems reasonable. However, ISD lacks clarity in this regard.

Based on this distinction, it is shown that five out of nine agile software development methods included in analysis place emphasis on abstract principles over concrete guidance.

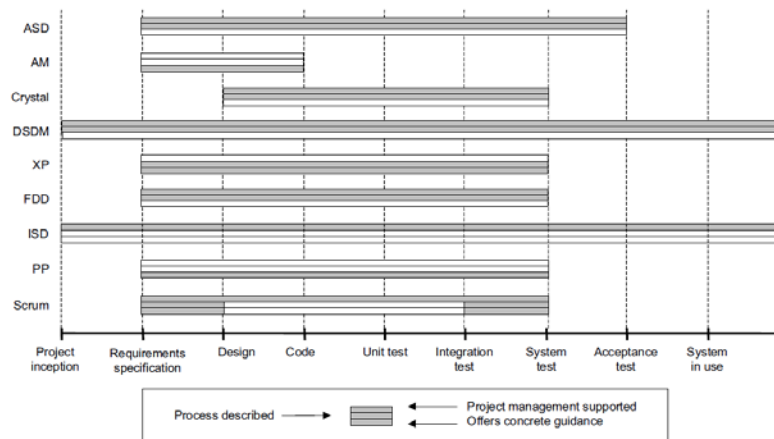


Figure 2. Comparing life-cycle, project management and concrete guidance support.

Figure 2. Comparing ife-cycle, project management and concrete guidance support.

CONCLUSION

Based on the literature study and our research findings, the study evaluates the use of Agile approach for success in mobile application development by determining the importance of the most used agile methodologies such as scrum, lean, XP etc. This study generates a number of contributions. The literature review identified suitable Agile practices for the development of mobile applications. The benefits of Agile practices for mobile application development is to improve the management and control. It makes sense, since agile approaches are focused project management (Scharff and Verma,2010). At the same time, agile practices help to increase the development speed[11].It is very important in the mobile market since new applications are available every day in the Apps store.

REFERENCES:

- [1] Wasserman,A.I.(2010). Software engineering issues for Mobile application development. In proceedings of the FSE/SDP workshop on figure of software engineering research,FoSER '10,pages 397-400,Santa Fe, New Mexico,USA.AC.
- [2] Abrahamsson, P., Hanhineva, A., Hulkko, H., Ihme, T., Jaalinoja, J., Korkala, M., et al.(2004).Mobile-D:agile approach for mobile application development. Conference on Object Oriented Programming Systems Languages and Application;Companion to the 19th annual ACM SIGPLAN conference on object-oriented programming systems, languages, and applications(pp.174-175).Vancouver:ACM.
- [3] Beydeda, S., Book, M., & Gruhn ,V.(2005).Model-driven software development. Birkhauser.
- [4] Balagtas-Fernandez,F., &Hussanmann,H.(2008). Model-Driven Development of Mobile Applications.23rd IEEE/ACM International conference on Automated software Engineering,(pp.509-512).L'Aquila.
- [5] Carton, A., Clarke, S., Senart, A., & Cahil, V.(2007). Aspect-Oriented Model-Driven Development for mobile Context-Aware Computing. Proceedings of the first International Workshop on Software Engineering for Pervasive Computing Applications,Systems and Environment,(pp.5-8).
- [6] Braun,P.,& Eckhaus, R.(2008).Experiences on Model-Driven Software Development for Mobile Applicxations.Proceedings of the 15th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems,(pp.490-493).
- [7] Khambati,A., Grundy,J.,Warren, J., & Hosking, J.(2008). Model-driven Development of Mobile Personal Health Care Application. Proceedings of the 23rd IEEE/ACM International Conference on Automated Software Engineering,(pp.467-470).
- [8] Reichlmayr,T.(2011). Working towards the students scrum-Developing Agile Android Applications. ASEE Annual Conference and Exposition, Conference Proceedings.
- [9] Moore, A. and Flannery, W. (2007) Use of Extreme Programming Methodologies in IT Application Design Processes: An Empirical Analysis. In Portland International Center for Management of Engineering and Technology-Management of Converging Technologies- Management of Converging Technologies , Pages 2468-2475,Portland,OR,USA.
- [10] Macias, F., Holcombe, M..(2003) A Formal Experiment Comparing Extreme Programming with Traditional Software Construction. In proceedings of the fourth Mexican International Conference on Computer Science, pages 73-80.
- [11] Scharff,C and Verma ,R.(2010).Scrum to Support Mobile Application Development Projects in a Just-in-time Learning Context.Proceedings-International Conference on software Engineering, Pages 25-31.
- [12] V. Rahmimian and R. Ramsin, "Designing an agile methodology for mobile software development : A hybrid method engineering approach" in proceedings of second international conference on Research Challenges in Information Science, RCIS(2008). Marrakech,2008, pp. 337-342.
- [13] A.C. Spataru, "Agile Development Methods for Mobile Applications" PhD Thesis, University of Edinburgh, The University Of Edinburg,Edinburg,2010.
- [14] P. Abrahamsson, "Keynote: Mobile software development-the business opportunity of today". In proc. of the international conference on software Deveopment,2005,pp. 64-69.