A Review of Mobile Application Development in the Agile Software Development Environment

Neymis Misiko Jacob[1]

[1] Umma University

Abstract

With the ever changing dynamic environment in mobile devices, mobile software applications need to cope with some particular performance environment that include the limited resources, high autonomy requirements and mark et regulations among other constraints [1]. To provide a software development process that responds to these challenges, adoption of Agile methodology that follows an iterative approach to build software rapidly where the entire software development lifecycle is brok en down into smaller iterations helps to minimize overall risk s. This allows software projects developed though agile methodologies to be easily modified rapidly to meet the users/customers? requirements. This paper assesses the development of mobile applications in the agile software development environment.

Index terms— agile development environment, mobile applications, mobile-D.

1 Introduction

ccording to Sommerville [2], software processes are categorized as either plan-driven or agile processes. Plan-driven processes involves all process activities being planned in advance and progress is measured against the devised plan. Agile processes on the other hand, involves planning that is incremental and is much easier to change the process to reflect changing customer requirements. A mix of both processes in SE is important as each comes with its values necessary for product development. As Boehm and Turner [3] discuss, each approach is suitable for different types of software. Generally, SE need to find a balance between plan-driven and agile processes [2]. Spataru [4] established that agile methods are good approaches towards mobile application development coined together with end-user inclusion in the lifecycle, performance testing of components and adoption of software product line principles.

2 II.

3 The Concept Agile Software Development

Corrall et.al [1] stated that mobile software development and applications need to cope with a particular execution environment that have limited resources, high autonomy requirements, market regulations, and many other constraints. There is need for developers to therefore to employ the agile Author: Jomo Kenyatta University of Agriculture and Technology. e-mail: Jneyole434@gmail.com development which is an umbrella term for several iterative and incremental software development methodologies [5]. According to Sriram and Saji [6] a development methodology has a direct relationship with managing project complexity. It accompanies and has implications for usability, maintainability, adaptability, reliability and portability of the software being developed.

In the late 1990's, agile methodologies became dominant to address some shortcomings of traditional methodologies like heavy documentation, lack of productivity, reliability and simplicity [7]. Agile alliance, in response to more process driven traditional methodologies, stresses upon people, communication and customer priorities [8]. Also, different agile methodologies has exhibited flexibility to working within constraints, without demanding major upfront investments, while being adaptable to changing market conditions [9].

The fundamentals of agile methodology namely the incremental and iterative development methods required the researchers to work on the principle of separating design, implementation and testing [10]. To date the most popular agile methodologies include Extreme Programming (XP), Scrum, Crystal, Dynamic Systems Development Method (DSDM), Lean Development, and Feature-Driven Development (FDD) [5] have been employed in software development giving software products that are versatile and more robust to the market. Sriram and Saji [6] cited a document by Nerur & Balijepally [11] on the main characteristics of agile development methodologies stating that the main characteristics of agile development methodologies include more autonomy, decision-making discretion and adaptive understanding, have a theoretical background, which is consistent with problem framing, problem solving approaches in architecture and strategic management. This main characteristics indicate that agile methodologies provide the designers with greater opportunities to develop software products that reliable.

4   III. Plan-Driven and Agile Development

Software processes can be categorized as either plan -driven or agile processes. According to [2], Plan -driven are processes where all of the process activities are planned in advance and progress is measured against this plan. In comparison to agile A processes, Sommerville [2] indicates that in agile planning is incremental and it is easier to change the process to reflect changing customer requirements. Each approach is therefore suitable for different types of software. Good practice as discussed by Boehm and Turner [3] is that there is need to find a balance between plan-driven and agile processes.

Software processes are complex and, like all intellectual and creative processes, rely on people making decisions and judgments [2]. In line with this, software processes can be improved by process standardization where the diversity in software processes across an organization is reduced. Through this, there is improved communication and a reduction in training time, and makes automated process support more economical. Standardization is important first step in introducing new software engineering methods and techniques and good software engineering practice.

According to Mudasir Kirmani [12], agile methodology follow an iterative approach to build software rapidly where the entire software development lifecycle is broken down into smaller iterations. The iterations helps to minimize overall risks allowing a software development project be easily modified on the upfront and strictly maintaining the project schedule and budget. I all this the software product is delivered on time and according to the budget that it was initially proposed on.

Inspite of the agile methodology being iterative, the methodology has some limitations [6]. Such as dependence on run-time tacit knowledge rather than more documented information, lack of traceable and proved implementation guidelines for mission critical projects, lack of adequate support for repetitive and large scale projects and team requirement of highly talented, self-motivated individuals with a high degree of implementation freedom [7].The resulting software product/ project developed using agile methodologies are dependent mainly on organizational factors like customer commitment, decision time, team location and composition, corporate culture and people factors like competency and self-motivation [13].

According to Andrei C. Spataru [4] the main argument against agile methods is the asserted lack of scientific validation for associated activities and practices, as well as the difficulty of integrating planbased practices with agile ones. In order to address various concerns, several methodologies within the agile category have evolved Abrahamsson et al. [14] identifies ten of them as truly agile. They include: -SCRUM, PP, ASP, FDD, XP, DSDM, Crystal, AM and ASD. For mobile software development, agile methods represent a relatively new approach to software development [4].

IV.

5   Agile Project Management

According to Beck et al. [8] Agile alliance, in response to more process driven traditional methodologies, stresses upon people, communication and customer priorities. Rajagopalan and Mathew [6] indicated that agile software development methodology has a direct relationship with managing project complexity. According to Martin et al. the customer plays a very important role in agile projects with key responsibilities to drive the project, interact constantly with business users and provide requirements and participate in retrospection to test the intermediate deliverable and its compliance [15]. However, at times the customer may fail to keep these practices or be consistent due to the dynamic nature of development projects.

In solving this, the customer's role is essentially not played by a single person but there are pseudo roles assumed by different people to drive the project effective conclusion. Known by role labels such as the technical liaison, negotiator, and customer coach, skill specialists for designer, tester and quality facilitator [6]. Chow et al. [16] Identified factors that can serve to guide in the selection of agile methodologies for projects in organizations with specific characteristics. Project success or failure in agile methodologies are guided by six key dimensions of agile methodologies with specific attributes to guide selection of a specific methodology. They include: -delivery strategies, software engineering techniques, team capability, project management process, team environment and customer involvement [6].

V.

6 Agile Methods in Mobile Software Development

According to Spataru [4] mobile applications market is currently undergoing rapid expansion. He further asserts that mobile platforms continue to improve in performance as the users' need for a wide variety of mobile applications increases. Software development for mobile platforms comes with unique features and constraints. The development environment and the technologies that support the software are different compared to "traditional" settings. Abrahamsson et al [17] identified important distinguishing characteristics to be a high level of competitiveness, necessarily short time-to-delivery and added difficulty in identifying stakeholders and their requirements.

Due to substantial differences in the environment and in platform specifications modifications in customer needs and expectations. Abrahamsson [18] states Technological constraints apply to mobile platforms in the form of limited physical resources and rapidly changing specifications. There is need to focus on agile methods and their suitability for mobile application development. Agile methods for mobile software development emphasize the adaptability of enterprises to a dynamic environment [19]. The unique features of agile methods derive from the list of principles found in the Agile Manifesto [20].

Smart mobile devices like cellphones and tablets are a key target for software products and services [1]. Consequently, software applications for the smart devices is receiving greater attention by companies' individuals and business enterprises globally. Along with the features and computing power offered by these mobile equipment, mobile software applications (apps) bear the big challenge of performing satisfactorily in a heterogeneous and resource-limited environment [12] that demands high availability, efficient performance and short response time while delivering value to the end user.

In the mobile ecosystem mobile software development rapidly requires updates and developments to meet the diverse customer requirements against a universe of constraints that exist on mobile environments namely evolving and inherent. The evolving constraints include current limitations that will be solved in the future by the evolution in technology, resources, bandwidth and coverage. Inherent constraints on the other hand are those intrinsic to mobile platforms, since they are part of the operational condition of the system and will not be solved in the near future [1]. These constraints should be taken into account by developers to determine the practices to relieve them as means of non-functional requirements.

Wasserman [21], indicated that mobile applications development are similar to software engineering for other embedded applications. However, the mobile applications development present s ome additional requirements that are less commonly found if compared to traditional software applications [12]. Boehm and Turner [22] identify fundamental concepts to agile development such as: simple design principles, a large number of releases in a short time frame, extensive use of refactoring, pair programming, testdriven development, and seeing change as an advantage. Abrahamsson, et al. [23] mention that an agile development method is incremental, cooperative, straightforward and adaptive.

Along the agile methodology, a new development methodology specifically tailored for mobile application development, called Mobile-D, is presented [17]. The method is based on agile practices, drawing elements from well-established agile methods such as Extreme Pro gramming and Crystal Methodologies and also from the heavier Rational Unified Process [24]. The Rational Unified Process is explained from a practical point of view in [25]. Practices associated to Mobile-D include test-driven development, pair programming, continuous integration, refactoring, as well as software process improvement tasks. The methodology serves as a basis for this work, and will be further detailed in the following section [26].

The Mobile-D process should be used by a team of at most ten co-located developers, working towards a product delivery within ten weeks. The Mobile-D methodology of the agile method comprises five phases: Explore, Initialize, Productionize, Stabilize, and System Test & Fix. Each of these phases has a number of associated stages, tasks and practices. The complete specifications of the method are available in [27].

One issue with the mobile-D method is that the well-known practices adopted by the Mobile-D process do not account for the performance of components under development, either at design time or during coding [4]. To address the issue integration of performance testing activities in the lifecycle, more specifically extending the Test-Driven Development practice with performance aspects is mentioned.

7 VI.

8 Conclusions

The importance of projection completion JIT cannot be over emphasized, with the use of the agile methodology in mobile software development, it introduces end -user feedback support, application categories and software product lines principles. The Mobile-D presents more of the ideal characteristics. Since the methodology was improved in aspects such as the inclusion of performance testing in the lifecycle. The ever dynamic changing environment of the mobile devices, software developers for mobile applications must be up to the task to develop software apps that meet this limitations and still satisfy the user needs. The different agile methodologies have

8 CONCLUSIONS

160 demonstrated flexibility to working within constraints, without demanding major upfront investments, while
161 being adaptable to changing market conditions [9]. [1] [2] [3]

162 [Beck et al.] , K Beck , M Beedle , A Cockburn , W Cunningham , M Fowler , J Grenning , J Highsmith , A
163   Hunt , R Jeffries , J Kern , B Marick , R C Martin , S Mellor , K Schwaber , J Sutherland , D Thomas , B .

164 [Spataru ()] , A C Spataru . *Agile Development Methods for Mobile Applications* 2010. University of Edinburgh

165 [Spataru ()] , A C Spataru . *Agile Development Methods for Mobile Applications* 2010. University of Edinburgh

166 [Cho ()] 'A hybrid software development method for large-scale projects: rational unified process with scrum'. J
167   Cho . *Issues in Information Systems* 2009. 10 (2) p. .

168 [Chow and Cao ()] 'A survey study of critical success factors in agile software projects'. T Chow , D B Cao .
169   *Journal of Systems and Software* 2008. 81 (6) p. .

170 [Versionone (2017)] *Agile 101 General Learnings*, Versionone . 30th July 2017. (VersionOne Company)

171 [Kirmani ()] 'Agile methods for mobile application development: A comparative analysis'. M M Kirmani .
172   *International Journal of Advanced Research in Computer Science* 2017. 8 (5) p. .

173 [Agile Software Development Manifesto. Retrieved from Manifesto for Agile Software Development ()]
174   *Agile Software Development Manifesto. Retrieved from Manifesto for Agile Software Development*,
175   `http://agilemanifesto.org/` 2001. (Agile Alliance)

176 [Abrahamsson et al. ()] 'Agile Software Development Methods: A Comparative Review1'. P Abrahamsson , N
177   Oza , M T Siponen . *Agile Software Development*, (Berlin, Heidelberg) 2010. Springer. p. .

178 [Abrahamsson et al. ()] 'Agile Software Development Metods: Review and Analysis'. P Abrahamsson , O Salo ,
179   J Ronkainen , J Warsta . *VTT Electronics*, 2002.

180 [Abrahamsson ()] 'Agile Software Development of Mobile Information Systems'. P Abrahamsson . *Advanced
181   Information Systems*, (Berlin) 2007.

182 [Corral et al. ()] *Agile Software Development Processes for Mobile Systems: Accomplishment, Evidence and
183   Evolution*, L Corral , A Sillitti , G Succi . 2013. Berlin Heidelberg: Springer.

184 [Mohammad et al. ()] 'Agile Software Methodologies: Strength and Weakness'. A H Mohammad , T Alwada'n ,
185   J A Ababneh . *International Journal of Engineering Science and Technology (IJEST)* 2013. 5 (3) p. .

186 [Available (2017)] `http://agilemanifesto.org/` *Available*, August 2017.

187 [Boehm and Turner ()] *Balancing Agility and Discipline: A Guide for the Perplexed*, B Boehm , R Turner . 2003.
188   Boston: Addison-Wesley.

189 [Boehm and Turner ()] *Balancing Agility and Discipline: A Guide for the Perplexed*, B Boehm , R Turner . 2003.
190   USA: Addison-Wesley.

191 [Vinekar et al. ()] 'Can agile and traditional systems development approaches coexist? An ambidextrous view'.
192   V Vinekar , C W Slinkman , S Nerur . *Journal of Information systems management* 2006. 2 (3) p. .

193 [Rajagopalan and Mathew ()] 'Choice of Agile Methodologies in Software Development: A Vendor Perspective'.
194   S Rajagopalan , S K Mathew . *Journal of International Technology and Information* 2016. 25 (1) p. .

195 [Beck and Andres ()] *Extreme Programming Explained: Embrace Change*, K Beck , C Andres . 2004. USA:
196   Addison-Wesley Professional. (2nd Edition)

197 [Larman and Basili ()] 'Iterative and incremental development: A brief history'. C Larman , V R Basili .
198   *Computer* 2003. 36 (6) p. .

199 [Abrahamsson et al. ()] 'Mobile-D: an agile approach for mobile application development'. P Abrahamsson , A
200   Hanhineva , H Hulkko , T Ihme , J Jäälinoja , M Korkala . *Conference on Object Oriented Programming
201   Systems Languages and Application; Companion to the 19th annual ACM SIGPLAN*, (Vancouver) 2004.

202 [Vtt Electronics ()] 'Portal of Agile Software Development Methodologies'. Vtt Electronics . `http://virtual.`
203   `vtt.fi/virtual/agile/mobiled.html` *VTT Electronics*, 2006. 2017. (Accessed 01st September)

204 [Sommerville ()] *Requirements Engineering*, I Sommerville . 2016. Boston, Massachusetts: Pearson Education,
205   Inc. p. . (in Software Engineering, 10th Edition)

206 [Salo ()] O Salo . *Enabling Software Process Improvement in Agile Software Development Teams and Organisa-
207   tions*, (Helsinki) 2006. VTT.

208 [Wasserman ()] 'Software engineering issues for Mobile application development'. A I Wasserman . *proceedings
209   of the FSE/SDP workshop on figure of software engineering research*, (the FSE/SDP workshop on figure of
210   software engineering researchFoSER, New Mexico, USA.AC.) 2010.

211 [Van ()] *The agile manifesto*, A Van . 2014.

212 [Kroll and Kruchten ()] *The Rational Unified Process Made Easy: A Practitioner's Guide to the RUP*, P Kroll
213   , P Kruchten . 2003. USA: Addison-Wesley Professional.

214 [Nerur and Balijepally ()] 'Theoretical reflections on agile development methodologies'. S Nerur , V Balijepally .
215   *Communications of the ACM* 2007. 50 (3) p. .

216 [Martin et al. ()] 'XP customer practices: A grounded theory'. A Martin , R Biddle , J Noble . *Agile Conference,
217   AGILE'09 -IEEE*, 2009.