

EMERGING PERSPECTIVE IN M-GOVERNANCE: AN AGILE METHODOLOGY APPROACH OF DESIGN

Abhishek Kumar Pathak¹
Research Scholar
Uttarakhand Technical University
Dehradun, India
pathak4utu@gmail.com

Dr. Himanshu Bahuguna²
Associate Professor
Shivalik College of Engineering
Dehradun, India
himanshu1081@gmail.com

Abstract: *In software application development, agile software development (ASD) is a methodology for the creative process according to the future need of the client it flexible enough and applies those changes at the time of the delivery of the finished product. Agile software development focuses on keeping code simple, testing often, and delivering functional bits of the application as soon as they're ready. The goal of agile methodology is to build upon small client-approved parts as the project progresses, as opposed to delivering one large application at the end of the project.*

Agile is a software development methodology to build a software incrementally using short iterations of 1 to 4 weeks so that the development is aligned with the changing business needs. This simple tutorial uses appropriate examples to help you understand agile development in a general and quick way.

m-Governance is not a replacement for e-Governance, rather it complements e-Governance. e-Governance is the use of information technology like WAN, Internet and mobile computing by Governance agencies, to transform private businesses and public agencies, as well as to empower the citizens. m-Governance, on the other hand, is the use of mobile or wireless to improve Governance service and information "anytime, anywhere". Mobile applications also rely on good back office ICT infrastructure and work processes. This paper shares the potential of using mobile phones as input devices in certain areas where last mile connectivity becomes issues for simple data inputs of critical importance for decision making in government departments.

Keywords: Agile methodology, ASD, M-governance, iterations, ITC, Last mile connectivity, Mobile mindset, Digital Divide.

Introduction:

Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. [1] These builds are provided in

iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross functional teams working simultaneously on various areas like –

Planning

Requirements Analysis

Design

Coding

Unit Testing and

Acceptance Testing.

At the end of the iteration, a working product is displayed to the customer and important stakeholders.

What is Agile?

Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. [2] In Agile, the tasks are divided to time boxes (small time frames) to deliver specific features for a release.

Iterative approach is taken and working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features required by the customer.

Here is a graphical illustration of the Agile Model –

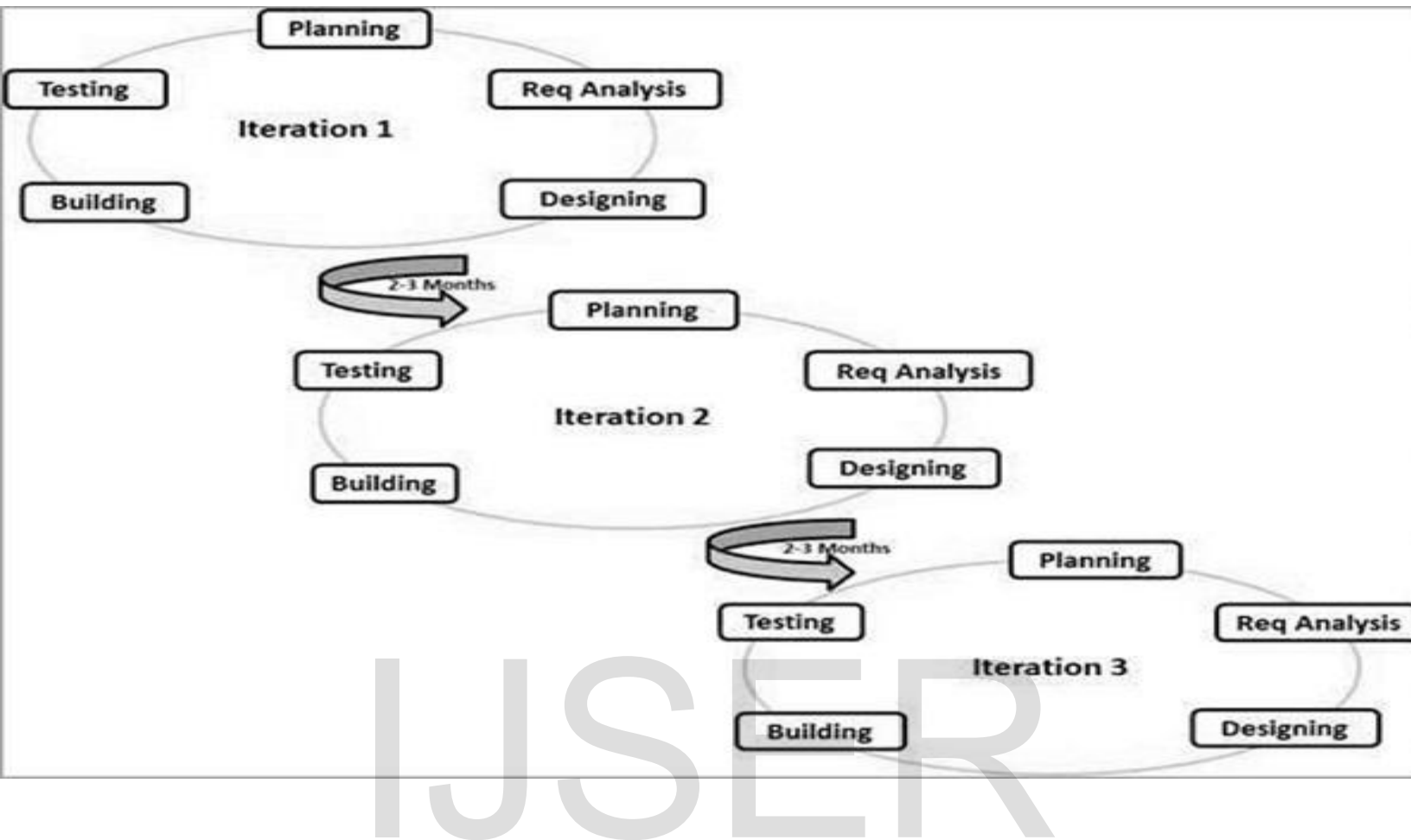


Figure1: Graphical representation of Agile Methodology

The Agile thought process had started early in the software development and started becoming popular with time due to its flexibility and adaptability.

The most popular Agile methods include Rational Unified Process (1994), Scrum (1995), Crystal Clear, Extreme Programming (1996), [3] Adaptive Software Development, Feature Driven Development, and Dynamic Systems Development Method (DSDM) (1995). These are now collectively referred to as Agile Methodologies, after the Agile Manifesto was published in 2001.

Following are the Agile Manifesto principles –

Individuals and interactions – In Agile development, self-organization and motivation are important, as are interactions like co-location and pair programming.

Working software – Demo working software is considered the best means of communication with the customers to understand their requirements, instead of just depending on documentation.

Customer collaboration – As the requirements cannot be gathered completely in the beginning of the project due to various factors, continuous customer interaction is very important to get proper product requirements. [5].

Responding to change – Agile Development is focused on quick responses to change and continuous development.

Agile Vs Traditional SDLC Models

Agile is based on the adaptive software development methods, whereas the traditional SDLC models like the waterfall model is based on a predictive approach. Predictive teams in the traditional SDLC models usually work with detailed planning and have a complete forecast of the exact tasks and features to be delivered in the next few months or during the product life cycle.

Predictive methods entirely depend on the requirement analysis and planning done in the beginning of cycle. Any changes to be incorporated go through a strict change control management and prioritization.

Agile uses an adaptive approach where there is no detailed planning and there is clarity on future tasks only in respect of what features need to be developed. There is feature driven development and the team adapts to the changing product requirements dynamically. The product is tested very frequently, through the release iterations, minimizing the risk of any major failures in future.

Customer Interaction is the backbone of this Agile methodology, and open communication with minimum documentation are the typical features of Agile development environment. [6] The agile teams work in close collaboration with each other and are most often located in the same geographical location.

Agile Model - Pros and Cons

Agile methods are being widely accepted in the software world recently. However, this method may not always be suitable for all products. Here are some pros and cons of the Agile model[7].

The advantages of the Agile Model are as follows –

Is a very realistic approach to software development.

Promotes teamwork and cross training.

Functionality can be developed rapidly and demonstrated.

Resource requirements are minimum.

Suitable for fixed or changing requirements

Delivers early partial working solutions.

Good model for environments that change steadily.

Minimal rules, documentation easily employed.

Enables concurrent development and delivery within an overall planned context.

Little or no planning required.

Easy to manage.

Gives flexibility to developers.

The disadvantages of the Agile Model are as follows –

Not suitable for handling complex dependencies.

More risk of sustainability, maintainability and extensibility.

An overall plan, an agile leader and agile PM practice is a must without which it will not work.

Strict delivery management dictates the scope, functionality to be delivered, and adjustments to meet the deadlines.

Depends heavily on customer interaction, so if customer is not clear, team can be driven in the wrong direction.

There is a very high individual dependency, since there is minimum documentation generated.

Transfer of technology to new team members may be quite challenging due to lack of documentation.

How Agile methodology can help in M-governance.

Corporate Governance

– the set of processes, customs, policies, laws, and institutions affecting the way a corporation (or company) is directed, administered or controlled.

• IT Governance

– A subset of Corporate Governance

– Focused on information technology (IT) systems and their performance and risk management.

Application of Governance

• Typical Governance

– Assumes traditional project management / delivery

– Characterized by big up-front analysis

– Status & Compliance ‘demonstrated’ through documentation

• The Agile Challenge

– Expected documentation is not intrinsic to Agile lifecycle

- Providing such documentation:
- Runs contrary to the Agile Philosophy
- Has no intrinsic value to the projects or the solution
- May place a burden on the project that at worst destroys Agility and at best impedes it.

Alternatives

If traditional governance is not right...

- No governance of Agile projects
 - Legislative compliance cannot be demonstrated
 - Path to business losses, reputational damage or weakened
 - competitive position may not be seen soon enough
 - Poor strategic alignment, higher costs and compromised quality may lead to a failure to realize benefits
- Agile governance of Agile projects
- Organizations can decide for themselves how they demonstrate compliance. This requires:
 - Analysis of the business and/or legislative imperatives
 - Understanding of the underlying delivery philosophy
 - Aligning governance with that philosophy
- In an Agile context this will probably require:
 - The creation of an alternative governance approach
 - A complete change of mindset from the traditional
 - A focus on artefacts delivered that are intrinsic to the delivery process (not documents that are bolted on).

Status of M Governance in India

Information and Communication Technology (ICT) is very useful for processing, storing, organizing, and presenting data and information. Indian telecom subscriber base reached the extraordinary figure of 700 mn; mobile phones have become the most accessible tool of communication available to such a large population. Recently, [8] Reserve Bank of India has allowed commercial banks to provide banking services on mobile phone, whereas Government of India has approved the “Framework for delivering financial services through mobile phone”

developed by Inter-ministerial group. After the launch of 3G technologies in India, users will be able to access health, educational, agricultural, infotainment services on their mobile phone. Around 54 Gram Panchayats in five remote blocks of West Bengal State will soon have SMS alerts on disasters, funds inflow and outflow, information about health camps and pulse polio campaigns will be sent to and from between the State departments, district offices and Gram Panchayats, block development offices. In India, m-Governance is still at a nascent or new stage. While many innovative applications are underway in both private sector as well as government domains, it may be a little premature to celebrate its success. However, embracing the possibilities by developing mobile applications in local languages and more mobile utilization applications. To improve the m-Governance in India, [9] it is proposed that development of suitable mechanisms to enable users to pay for public services through mobile phones, and develop and deploy innovative public private partnership. The inception of the M-governance concept will certainly lead to prove that how the effective digital medium can bring the profound returns in the country and lead to effective development of society.

Challenges

Although the concept of m-governance has got its footage in southern part of country and have shown the positive levels in the development, yet there are certain important challenges which are to be faced for the implementation of this concept on national basis [10]:

Cost: M-government can truly substitute for other delivery channels. Such substitution will be viable for applications within government. Such systems are likely to be cost-addition rather than cost-substitution initiatives for example, using fee sharing arrangements that avoid the public sector having to provide many up-front costs.

M Digital devise: In particular, ever one, older and poorer groups in society tend to be excluded from this technology. If there are benefits to be had from m-government, these groups will be denied them, and a challenge to m-government is to ensure it is not just one more way in which the "haves" benefit at the expense of the "have nots".

Mobile mindsets: mobile devices-cell phones particularly - are seen by many as tools more for fun and entertainment than for serious activities. Yet politics is a serious business involving difficult choices. Aligning these two mismatched worlds

Conclusion

So agile methodology can work as helping hand for providing the better result in the proper implementation of m-Governance in India. As the strategy is very much convenient and the working parameters which are being provided by agile methodology is very easy to use, as it consists all the advantages which are being provided by different software developments models.

References:

- [1]. Beck et. al. (2001). Manifesto for Agile Software Development. Retrieved 05 01, 2009, from agilemanifesto.org: <http://agilemanifesto.org/>
- [2]. Beck, K., & Andres, C. (2004). Extreme Programming Explained: Embrace Change. Addison-Wesley Professional.

- [3]. Beck, K., & Fowler, M. (2000). *Planning Extreme Programming*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.
- [4]. Boehm, B., & Ross, R. (1989). Theory-W software project management principles and examples. *IEEE Trans Software Engineering*, 15 (7), 902-916.
- [5]. Ceschi, M., Sillitti, A., Succi, G., & Panfilis, S. (2005). Project management in plan-based and agile companies. *IEEE Software*, 22 (3), 21-27.
- [6]. Chan, F., & Thong, J. (2009). Acceptance of agile methodologies: A critical review and conceptual framework. *Decision Support Systems* (46), 803-814.
- [7]. Charette, R. N. (2005, 09). Why Software Fails. Retrieved 3 14, 2007, from IEEE Spectrum: <http://www.spectrum.ieee.org/sep05/16858>.
- [8]. Antovski, L. & Marjan Gusev, G. (2005). M-Government Framework. Retrieved 29 September, 2009, from m-government.
- [9]. Baryamureeba, V. (2008). ICT-enabled Services: A Critical Analysis of the Opportunities and Challenges in Uganda. ICCIR '08: The 4th Annual International Conference on Computing & ICT Research (pp. 235-255). Kampala.
- [10]. Gutpa, Vandana and Ajay Sharma (2012), "E-Governance in India: Prospects, Problems and Challenges", *Research Journal of Economics & Business Studies*, Vol. 1, No. 9, pp. 50-54.