Cost Minimization Plan for Dynamic System Development Method in Agile Methodology

Anand Kumar Rai[1], Karuna Shankar Awasthi[2] and Laxmi Shankar Awasthi[3]

[1]Deptt. of Computer Science, Mumtaz Post Graduate College, Lucknow.
[2]Deptt. of Computer Science, Lucknow Public College of Professional Studies, Lucknow.
[3]Deptt. of Computer Science, Lucknow Public College of Professional Studies, Lucknow.

Abstract

Dynamic system development method of agile methodology is very less popular amongst software developers because of its higher cost and its development team overheads. DSDM method is generally used for very large project with higher requirement change possibility. In this paper it has been suggested that with the help of some AI based automation tools the roles and responsibility of development teams can be reduced. Reduced development efforts can reduce the development cost also.

Keywords: *Agile methodology, AI tools, DSDM.*

1. INTRODUCTION

Since 1990, Rapid application development (RAD) has always been important for the IT industry. New application development tools have been introduced like PowerBuilder. These tools have allowed to developers to help in their proposed solutions in easier way for customers Now prototyping has been come true and problems of the classical and sequential development methods removed [1][4][6][7].

Actually the RAD movement was not structured, No common definition was defined for a suitable process and many organizations introduced their own process and definition. Many important corporations had shown their interest in the possibilities but worried about they did not want to drop the level of quality [3][4].

The DSDM Consortium was introduced in 1994 in the field of software engineering and It was developed for the objective of "jointly developing and promoting an independent RAD framework". It could be achieved by joining their best practice experiences. The origins of the DSDM Consortium were an event which was organized by the Butler Group in British. In that meeting all people worked for blue-chip organizations like British Airways, American Express, Oracle, and Logical[10][12][14].

In late July 2006, Public Version of DSDM was available for people to view and use this, but if anyone wants to resell DSDM then needed to be a member of the non-profitable consortium.

In early 2014, handbook for the DSDM was available online and public and templates for DSDM could be downloaded.

In October 2016, the DSDM was rebranded as the **Agile Business Consortium** (ABC). The Agile Business Consortium is known as a not-for-profit vendor-independent organisation

(i) **Time boxing:** It is the approach project is divided into many portions. Each portion is having fixed budget and each would be fixed delivery date. In each portion, priority is set for requirements. In this time and budget are already fixed and requirements are variables. In this approach if the project is lacking in time or lacking in money then lower priority requirements are cut down. It does not unfinished product is delivered, reason behind the Pareto Principle that 80% of the project comes from 20% of the system requirements, then most important 20% of requirements are implemented into the system, Since no one system can be built perfectly in first time but the system meets the business needs[5][6]

(ii) **MoSCoW:** In this technique work items or requirements are priorities. It is an acronym that stands for: M (MUST have) S (SHOULD have) C (COULD have) W (WON'T have)

(iii) **Prototyping:** In early stage of the project prototype is developed. It enables the early finding of the problems in the system and help users to 'test-drive' the system. In this way good user involvement is realised, one of the key success factors of DSDM, or any System Development project for that matter.

(iv) **Testing:** It helps to ensure a solution of good quality, DSDM promotes the testing throughout every iteration. Since DSDM is an approach and technique independent method, the project team can choose its own test management method freely.

(v) **Workshop:** It helps in managing the project stakeholders together to brief requirements, functionalities and mutual understanding.

(vi) **Modeling:** It helps in visualising a business domain and improves understanding. It Produce a diagrammatic representation of specific aspects of the system or business area that is being developed.

(viii) **Configuration Management:** at the end of the each time box delivery of the developed product must be ensured. It should be well managed towards completion.

1.1 Roles

Before commencing the project, there are requirement of some people who will play the specific role in the development each role will fulfil its own responsibility. The roles are[7][8]:

- **Executive Sponsor** This is from the user organization who commit proper funds and resources. This has ultimate power to take decisions.
- **Visionary** to find the essential requirement of the project early and initialise the project. Visionary has the accurate perception of the project. It ensures that project progress on right track.
- **Ambassador User** the responsibility of Ambassador User is to bring the knowledge from user community to the project developers group.
- **Advisor User** this is from user end that can represent different viewpoint of the project can be any user that represents an important viewpoint and brings daily knowledge of the project.
- **Project Manager** The project manager takes care of overall project management; this may be anyone from the user community or IT staff.
- **Technical Co-ordinator** this is responsible for designing the system architecture and quality control of the project.
- **Team Leader** this ensures that team works effectively.
- **Solution Developer** this makes system requirement interpretation and build the prototype.
- **Solution Tester** this verify the system correctness by performing some testing techniques. The tester has to create test cases and documentation.
- **Scribe** this is Responsible for collecting the system requirements, agreements, and decisions made in every workshop.
- **Facilitator** this is acts as a motivator for project preparation and proper communication.
- **Specialist Roles** Business Architect, Quality Manager, System Integrator, etc.

1.1 Research Problem

DSDM is very effective technique in agile methodology to develop any large size project but many industries are hesitant to adopt the DSDM life due its process overheads and cost. in comparison to other agile methods DSDM is rare usable.

2. LITERATURE REVIEW

Since its inception in 1994, DSDM, the Dynamic Systems Development Method, has gradually become the first framework for rapid application development (RAD) in the UK. DSDM is a non-profit framework for RAD development, maintained by the DSDM Consortium. The developers of the system maintain that in addition to operating as a concept in a widely accepted concept DSDM also provides a framework for RAD controls, supplemented with guidance on how to best use those controls. The basic idea of DSDM is that instead of adjusting the amount of performance in a product, and then adjusting the time and resources to achieve that function, it is preferable to adjust the time and resources, and adjust the value accordingly [2][3].

DSDM has been widely used in the UK since the mid-90. Eight case studies are documented in, and the experience clearly shows that DSDM is a different form of RAD. To facilitate the adoption of the method, the DSDM Consortium has published an appropriate filtering method, in which there are three integrated areas: business, systems and technology [4].

3. PROPOSED DSDM LIFE CYCLE MODEL

In this proposed model first phase is conducted with project study where requirement analysis is done through business study and feasibility study, as each and every development model containing this phase. The purpose of this proposed model is to automate some time taking processes by using AI tools.

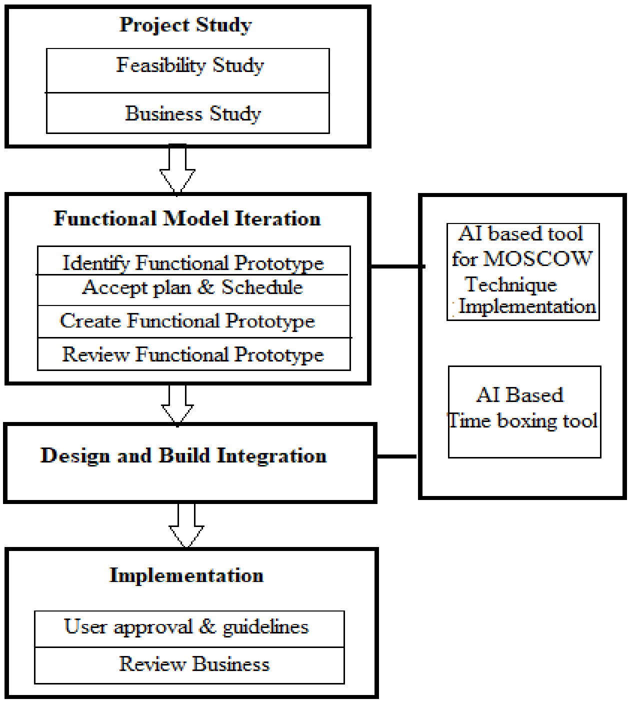
Project Study
Feasibility Study
Business Study
Functional Model Iteration
Identify Functional Prototype
Accept plan & Schedule
Create Functional Prototype
Review Functional Prototype
Design and Build Integration
Implementation
User approval & guidelines
Review Business
AI based tool for MOSCOW Technique Implementation
AI Based Time boxing tool

Figure 1. Proposed DSDM Life Cycle Model

The second phase of this proposed model describe the functional model iteration. The first step of this phase is to identify the functional prototype. Functional prototype describe about the different functions and further planning and scheduling of the project sprint.

In the second phase of this model user requirement prioritization MOSCOW technique is applied for the reduction of the roles and responsibilities of the few more team members. In the third and fourth step of this phase functional prototype creation and review is done. The AI based time boxing tools works at this step to keep track of the project schedule where human intervention is not required. By using the AI tools human efforts and cost can be reduced.

The third phase of this model describe about 'Design and Build Integration' in which different functions which have been developed and reviewed in previous phase are integrated altogether.

The fourth phase of this model is implementation phase where user approval and guidelines, business review are performed for implementation of the project. We can validate this proposed model with the help of some assumptions.

3.1 Research Validation

We can validate this proposed model with help of some assumptions, these assumptions data may be based on some case study or previous used research data.

Assumption-1

The company 'xyz' is planning a project 'abc'. The project is very large therefore project manager decides to do the whole project development using the agile methodology. Project is very crucial therefore company will not take risk with the quality of the project. Project manager chooses DSDM. The effort estimation for the given sprint are as follows.

1-Sprint duration- 4 weeks (208 hours estimated)

2- Team Size -16 persons

3- Project study (feasibility study and planning)- 20 hours 6 persons.

4- Functional Model Iteration (Prototype identification, Approve plan & schedule, create and review prototype)-100 hours 5 persons.

5- Design and build integration- 60 hours 5 persons.

6- Implementation (user approval and business review) -28 hours 5 persons.

Total efforts= 20x6 + 100 x 5 + 60x 5 + 28 x 5 = 1060 hours efforts

Assumption-2

This assumption is for the same project as mentioned above by using the proposed model life cycle. The above assumption is based on the traditional DSDM. We can assess the efficiency of the proposed model by the given assumptions. The effort estimation for the given sprint are as follows.

1-Sprint duration 4 weeks (208 hours estimated)

2- Team Size 10 persons

3- Project study (feasibility study and planning) 20 hours 3 persons.

4- Functional Model Iteration (Prototype identification, Approve plan & schedule, create and review prototype) 100 hours 4 persons.

5- Design and build integration 60 hours 4 persons.

6- Implementation (user approval and business review) 28 hours 3 persons.

Total efforts= 20 x 3 + 100 x 4 + 60 x 4 + 28 x 3 = 784 hours efforts

4. CONCLUSION AND FUTURE WORK

The proposed model is become fruitful for the agile software industries, where DSDM life cycle is desirable to use. By using the proposed model we can see that only 10 persons are completing the task where with the previous DSDM life cycle, 16 persons are completing the same task. Here quality is the major concern for the DSDM life cycle, it is required to conduct further research for the rigorous quality testing techniques.

5. REFERENCE

[1] PMI. The High Cost of Low Performance. Available: https://www.pmi.org/~/media/PDF/Business-Solutions/PMI_Pulse_2014.ashx.

[2] Dingsoyr T, Nerur S, Balijepally V, Moe NB. A decade of agile methodologies: Towards explaining agile software development. JSyst Soft. 85(6); 2012. p. 1213–1221.

[3] Dyba T, Dingsoyr T. Empirical studies of agile software development: A systematic review. Inf Softw Technol. 50(9–10); 2008. p. 833–859.

[4] Livermore JA. Factors that Significantly Impact the Implementation of an Agile Software Development Methodology. J Softw. 3(4); 2008. p. 31–36.

[5]Agile Methodology and PMBOK - PMP Mastery. Available: http://www.pm-primer.com/agile-methodology-and-pmbok/.

[6] Hoda R, Noble J, Marshall S. Developing a grounded theory to explain the practices of self-organizing Agile teams. Empir Softw En. 17(6); 2011. p. 609–639.

[7]Moe NB, Dings T, Dyb T. Understanding Self-Organizing Teams in Agile Software Development. In: 19th Australian Conference on Software Engineering. IEEE; 2008. p. 76–85.

[8] Chow T, Cao D-B. A survey study of critical success factors in agile software projects. J Syst Softw. 81(6); 2008. p. 961–971.

[9] Serrador P, Pinto JK. Does agile work? Int J Proj Manag. 33(5); 2015. p. 1040–1051.

[10] PMI. Implementating Organizational Project Management: A practice guide. PMI; 2014.

[11]Pope-Ruark, “Introducing Agile Project Management Strategies in Technical and Professional Communication Courses.” J. Bus. Tech. Commun., vol. 29, no. 1, pp. 112–133, Jan. 2015.
[12] E. C. Conforto and D. C. Amaral, “Agile project management and stage-gate model-A hybrid framework for technology-based companies: JET-M JET-M,” J. Eng. Technol. Manag., vol. 40, p. 1, 2016.
[13] A. J. Sohi, M. Hertogh, M. Bosch-Rekveldt, and R. Blom, “Does Lean & Agile Project Management Help Coping with Project Complexity?,” Procedia - Soc. Behav. Sci., vol. 226, pp. 252–259, Jul. 2016, doi: 10.1016/j.sbspro.2016.06.186.
[14]M. Hummel, “State-of-the-Art: A Systematic Literature Review on Agile Information Systems Development,” in 2014 47th Hawaii International Conference on System Sciences, Waikoloa, HI, Jan. 2014, pp. 4712–4721, doi: 10.1109/HICSS.2014.579.