

Theory and Procedures of Multidisciplinary Agile Software Testing Team

Madhu B K¹, Roshni Kanth², Sanjay K Nagendra³, Lokesh V⁴

^{1,3} G S S Institute of Technology, Bangalore

² Center for Post Graduate Studies, Bengaluru Region, VTU, Bangalore

⁵ Acharya Institute of Technology, Bangalore

Abstract

In recent days many companies are using Agile methodology as their life cycle to develop Software product. It is well known that Agile software testing is most suited procedure for testing under time bound complex scenario. Discussion has been initiated to collaborate different domain experts in testing team to increase the effectiveness of testing process. In this direction, the paper discusses and establishes the concept of Multidisciplinary Agile Software Testing (MAST) Team, which consists of technical experts from different sections of Software life cycle. The uniqueness of MAST Teamwork is in its potential to integrate different bodies of knowledge into a new synergy. However, previous empirical studies have shown that member heterogeneity and geographic separation hinder effective sharing and use of team knowledge. The paper also explores how such teams interact to overcome the barriers and take advantage of their "built in" knowledge diversity and also considering its human parameters. At the end, the paper gives a new approach for establishing MAST in any organization.

1. Introduction

Agile teams are multidisciplinary groups composed of members representing many Software engineering disciplines. Specialists from various disciplines (e.g. Feasibility Study, Business Analysis, Quality Control, Core Testing and development) are gathered to test a new product developed. The groups include formal but temporary assignments to groups, committees, and special projects. After a project has been completed, group members return to their routine activities or go to another temporary project group. The research works concerning team building in the context of design projects share some common characteristics: multidisciplinary tasks and designers, a large quantity of tasks in a project, a large number of designers, and a task-designer assignment problem. To reduce the project complexity, one way to build teams is to decompose large, complex design processes and project organization into a set of smaller task groups corresponding to different teams. A team should consist

of multiple designers with different technical backgrounds and expertise, contributing to a design task as part of the whole design project^[7]. It is expected that essentially, a team model should represent the interdependence between teams such that each team has its own objective and constraints for a distributed problem. In a task testing assignment, project managers always have to make a tradeoff between the preserving of intra-domain expertise and the development of extra-domain expertise. The problems of high rate of workforce turnover and competency deterioration can be stimulating factors for project managers to assign tasks to team members in order to increase their capabilities. There is a need to better integrate competency modeling in team building in order to take competency dynamics into account. With the coming potential for dramatic shifts in production paradigms, an opportunity exists for a faster design-to-production cycle for precision electromechanical devices.

A Multidisciplinary team practitioner was asked to assume leadership of the communications team. His responsibilities included the design and development of an information infrastructure to support the project structure and the facilitation of information flows in this large, geographically dispersed, multidisciplinary project team. This responsibility reflected recognition of the importance of human interactions in an information driven product development process. The Multidisciplinary team practitioner was presented with the challenge of applying Multidisciplinary team principles to an emerging Agile Software Testing team of professionals to define their contributions and establish their role early, while agile manufacturing concepts are still being formulated. Agile manufacturing paradigms assert that to be competitive, companies must focus on strategies enabling them to get highly customized, quality products to market faster than their competitors^[2]. This capability is achieved through flexible design and production processes and the application of information technologies to automate and streamline design and production. These capabilities are combined with an awareness of corporate strengths and a willingness to form "virtual corporations," on a product-by-product basis, with other

companies offering complementary strengths. Ideally, agile software development teams should be multidisciplinary, that is, possess all the skills needed to deliver the business solution.

2. Related Work

User-centered processes recommend combining non-functional as well as functional requirements by involving a Multidisciplinary Team^[2]. The early design stages of user centered design (UCD) include a user needs analysis and generally result in several artifacts such as usability requirements^[3], scenarios^[4] and personas^[5] describing the user needs. These artifacts are written in a narrative style and are usually created by interaction designers. Similar artifacts are used in software engineering and agile development^[6] (e.g. essential use cases, scenarios, story cards, user stories). Although several disciplines provide notations to describe user needs, the notations are not always comprehensible for all members of a Multidisciplinary Team^[7] address the difficulties in presenting user needs for requirements engineering. Earlier studies describe the needs of interaction designers in a Multidisciplinary Team. Researchers conducted an ethnographic study to investigate the collaboration between user interaction designers and developers^[8]. The study describes the benefits of stories and sketches in the early stages of user-centered approaches and emphasizes the power of combining both. Assembling stories and sketches is a powerful technique to reveal errors, and to consider temporal and contextual information. A recent study^[9] reports that designers are experiencing difficulties when designing the behavior of user interfaces. While prototyping the appearance of user interfaces is straightforward, designing and communicating the behavior is an ongoing process. Furthermore, the survey revealed designers frequently use sketches and storyboards.

3. Group Working in Agile

The main adversaries of agile processes are established, rigid documentation regimen, lack of time and lack of stakeholder buy-in. These can be felt in projects like exploring different solution in stipulated time frame, Establishing and designing concepts without proper base plane, Satisfying diverse (and often contradictory) client and user expectations, Having something up and running quickly in smaller scale projects, Ensuring team preparedness for an agile approach in case of converting the existing working style, No standardized documentation procedures, Enabling a multidisciplinary group of people to work together as a team, Aiding team-members to adhere to

new demands and expectations, Ensuring the team's buy-in to the process and resulting products, Supporting a shared understanding of both needs, constraints and solutions, Making sure that decision-making is based on informed choices. Multidisciplinary Teamwork is one of the success criteria in both user-centred approaches and agile methods. It is considered to be essential for agile processes in order to meet project objectives and to ensure stakeholder buy-in to both process and results. It must be stressed that though this a paper concentrates on the teamwork, other factors such as clear goals, planning, iterative delivery, etc. also play a vital role in the success of agile projects. Multidisciplinary Teamwork depends on, putting together, establishing and sustaining a team, Doing all important team-based work in facilitated workshops, Visualizing and documenting the team's efforts throughout the process.

In process of enabling a diverse team to function well a MAST Team needs to be made up of all the different stakeholders in the process. This includes project sponsors, domain experts, users, designers, developers and architects - just to mention a few. It is important that all the different interests are represented. This can be a difficult job because you can end up with a team that is too unwieldy to manage. So selecting team members can be a painful process. When building a team, each participant's potential contribution to the process has to be evaluated. Involving the right stakeholders in the process increases ownership from the users' and customer side. The difficult part is excluding some of potential participants. The optimized participant is a part of the team and identifies himself/herself with the team's goals, and also empowered and entrusted with the task, has the support of his/her manager, is experienced and knowledgeable in his/her field, represents colleagues, has the time to participate, wants to contribute to the success. Real-life team participants obviously represent these attributes to a varying degree. The participants from the customer or user groups are often hard to get hold of because they have important responsibilities in the organization. Quite often some of our team members are picked for us, so we have to work with people who in one way or another represent 'less' than the ideal participant. Since teams don't function well when decisions have to be made elsewhere, we try to avoid representation through proxy, e.g. the boss sending a secretary or the system architect sending a junior programmer to represent them. An ideal team includes people from different disciplines with various skills.

Agile Teamwork can only be productive if the team members appreciate the goals and strive towards reaching them. So it is important that they actually

represent the different aspects of the problem or solution space and are aware of their own role in the team. The team must treat people's ideas and concerns as equally important, irrespective of each team member's position or power, or they won't feel motivated to contribute. The team must also have the flexibility to change its course - we easily abandon one idea for another. This enables experimental thinking, but can lead to lower levels of buy-in and commitment. One of the ways teams can achieve a shared understanding, come to an agreement and commit to group decisions is by visualizing requirements, ideas and decisions in the Software development. This obviously requires focussed management so that the workshops are productive. So we go in for thorough planning and preparation in preshops (pre-workshop meetings) to reduce the risk of failure. The success of MAST Teamwork partially depends upon no single team member feeling substantially inferior to any of the others. Technical models like class diagrams; ERM models, event handling queues, etc. are never a part of the teamwork - despite the fact that the developers and architects in their own work use them. Achieving something is important to successful teamwork, so make sure that each development and testing procedures has achievable goals. The team has to decide towards the end of each workshop whether we have actually fulfilled the goals we set. This confirmation helps the team evaluate its own progress. Post-test documentation of the team's results ensures completeness. Figure 2 depicts the work flow of information in MAST Team

4. Mast Team Strategies

Agile manufacturing achieves reductions in product development time largely through the application of information technology strategies. Testing designs and patterns are developed and managed through information media. Alternative techniques are evaluated and design issues addressed through simulation analyses. Translation of designs to development paths and testing methods and generation of testing cases are greatly automated. Product development teams function in a fast-paced, concurrent engineering environment in which decisions are made quickly and often involve geographically dispersed participants. To a significant degree, the success of the agile enterprise things on the application of information technologies to access, exchange, and use information.

4.1. MAST Team Information Management process

Information flow analysis demonstrated that information requirements through the course of a product development and testing cycle are many, dynamic, and time-critical. The management of information and task allocation is as significant as are channels by which information flows. In traditional, serial product development cycles, as well as many concurrent engineering approaches, information flows from person to person; any individual's awareness and access to specific information depend on interpersonal, often coincidental, communication with others. Information bottlenecks occur when personnel who generate or possess key product information are unavailable, when information is not disseminated to all those affected, and when multiple, uncontrolled, or informally controlled versions of product information coexist (e.g., multiple versions of a design). Agility requires project-wide and immediate access to most information, project-wide awareness of the existence and availability of information, and strict control over the versions of information that are available. In addition, the success of agile enterprises hinges considerably on their ability to dynamically incorporate lessons learned and to make optimal use of their corporate product development history with each new project. These considerations point to the need for enterprise-wide electronic archives that may be readily and efficiently accessed by all engineering disciplines throughout the product development process. Agile Software Testing's Multidisciplinary team practitioner contributed to design through the analysis of enterprise functional requirements and of user and job-specific requirements, as well as user interface design and evaluation. By emphasizing information requirements and information flow and applying a user-driven design approach, the Multidisciplinary team practitioner offered a unique perspective that contributed to development of an optimal solution that successfully satisfied both enterprise and individual requirements.

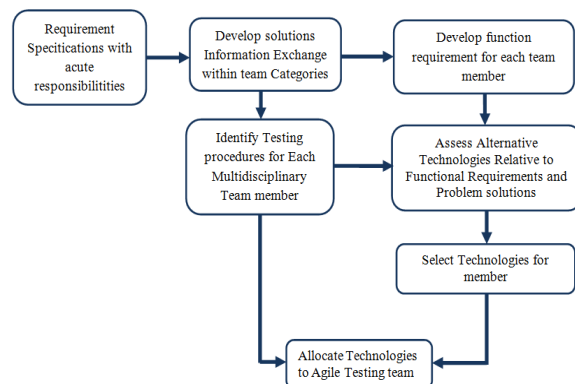


Figure 1: Work flow of information in MAST Team

4.2. Agile software Testing Automation

A primary mechanism by which agile manufacturing shortens product development times is automation of design and production tasks. Contributing to substantial time savings are manufacturing tools, translators that facilitate the seamless transference of product models between software applications, electronic representations of knowledge and processes, and accurate computer analysis used for simulating physical testing. However, consideration must also be given to appropriate and inappropriate operations for automation. Appreciation for uniquely human contributions to the design and production process must not be neglected, marginalized, or oversimplified. For example, automating design steps may eliminate cross-discipline collaboration among team members, communication that could positively influence design and production decisions. To assess the appropriate application of automation, it is helpful to have an understanding of decision-making processes and an appreciation for where decision making follows simple rule-based reasoning as opposed to reliance on intuitive, creative processes that are not easily captured within currently attainable automated systems. It is also important to understand the process by which decisions are made and the range of factors that contribute to making them. Seemingly similar decisions may involve non-overlapping paths of reasoning and be influenced by somewhat different factors. Furthermore, the choice of what input is required for a decision and the manner in which different inputs are weighted in the decision-making process may vary from simple to complex. For these reasons, decisions regarding automation should be based on careful assessment of the human contributions to the process and the confidence with which automated systems may capture those human contributions. Using cognitive task analysis and similar tools, the Multidisciplinary team practitioner is capable of providing valuable recommendations regarding the appropriate application of automated systems. Ideally, automation should be employed to eliminate simple rule-based, labor-intensive tasks.

4.3. Human Reliability in MAST Team

Within a tightly coordinated, fast-paced agile enterprise, delays that are tolerable within traditional enterprises may significantly degrade the ability to deliver products on schedule. Human-induced delays represent an important - if not the primary - source of potential delays. In Agile Software Testing, we encountered many instances of process delays and impedance resulting from human actions (or inactions). Traditionally, software application and operating system upgrades have been somewhat happenstance, as part of an individual's or organization's responsibility, often independent of specific project requirements.

Users are frustrated but generally accepting of the incompatibilities that result when more than one operating system or software version is in use at a time. For Agile Software Testing, it was ironic that after designing a backup system for the project that would effectively eliminate system unavailability caused by equipment failures, the database became unavailable to two key team members because they upgraded their operating systems to a version that was incompatible with the software. Network maintenance and administration between different LANs with different security requirements, computer platforms, and standard system configurations was a significant source of system vulnerability. When equipment failed, substantial delays occurred between the time of the failure and the time when appropriate network support staff became aware of the problem. Typically, responsibility for specific components of the network was held by different support staff from different organizations who were located in different places. Resolution of network problems was often delayed by staff absence or unavailability or by the failure to locate appropriate support staff. This problem was compounded by their excessive workload, which often prohibited problem resolution and minimized opportunities to monitor system status. Limited communication between network support staff and project staff also caused technical incompatibilities and a loss of productivity from what would otherwise have been well-received enhancements. For example, without warning, several key team members were moved from one local area network to another; as a result, they lost access to some project resources, and the use of other project resources was made impractical. Given the reliance on network communications, elimination of human points of failure in the maintenance and administration of the communications network is critical to achieving necessary levels of system reliability and availability. Task analysis and human reliability analysis offer tools by which Multidisciplinary team practitioners may identify, assess, and aid in eliminating vulnerabilities caused by human points of failure.

4.4. Technology Introduction and Culture Change

Introduction of new technologies and achievement of the cultural changes enabled by these technologies were the most difficult Multidisciplinary team challenges encountered with the Agile Software Testing project. Agile manufacturing requires an unprecedented degree of concurrent engineering between often geographically dispersed project participants with varying degrees of overlap in their motives, expectations, and common understandings. These factors were compounded by technical difficulties that enhanced the unwelcoming images of the new technologies. A more fundamental

factor, however, was the need to achieve culture changes essential to the full adoption of the new technologies. Team and individual training, strong and persistent encouragement from managers, and general recognition of the benefits of the new technologies were not sufficient. Progress did not begin until team members redefined their engineering practices, with the derivative effect of incorporation of the new technologies into these practices. Multidisciplinary team practitioners may facilitate the introduction of technology in several ways. In redefining business practices to incorporate new technologies, they offer insight into work and information flows, cultural and organizational issues, and job functions and roles. This can help engineering team members understand how the use of new tools can increase their efficiency.

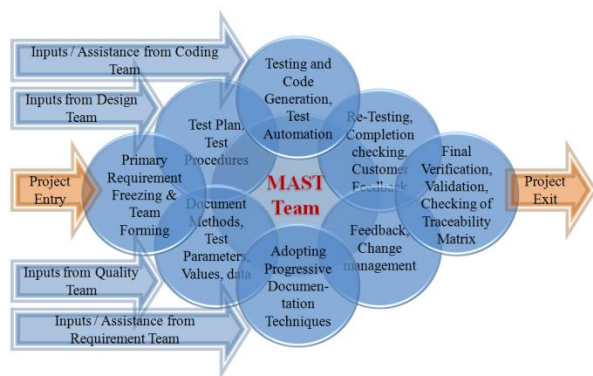


Figure 2: Functionality merger of MAST Team

5. The Role of Mast Team

For the Agile Software Testing project, in addition to leading the communications team, the Multidisciplinary team practitioner provided input to each of the other teams, most notably leading the process description and improvement activities of quality assurance. The figure 3 gives diagrammatical explanation of how the MAST Team shares the information and works towards the project completion. The various inputs are given at different stages making it procedure inside the Agile life cycle. As the project proposal received and the development team starts working a dedicated MAST team starts robust testing procedures. As the team involves many people's from different sections, things like customer interaction, documentation, code improvisation and quality control is taken care by team members without delay in the delivery. The overlapping circles indicate the parallel run of the procedures in that specific routine. It's true that ambiguity of completeness is a major concern here and as the team members are experts in their specific field this issues is suitability taken care by quality team. Agile manufacturing requires that personnel be very flexible in adjusting to the continuously changing work environment. Consequently, all project team members

will be required to fulfill project needs that extend beyond the traditional boundaries of their professions. For this reason, although MAST Team/ergonomics is certainly not a prerequisite for leading communications, a Multidisciplinary team practitioner would be an appropriate choice for this position.

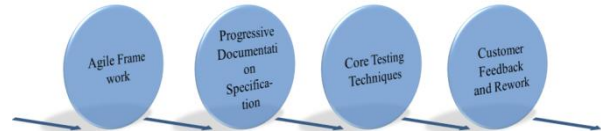


Figure 3: Progressive flow of life cycle

Most of the Multidisciplinary team activities discussed here represents derivations of tasks usually undertaken during system development, human error analysis, or user interface design. These are all familiar activities for Multidisciplinary team practitioners. In contrast, most Multidisciplinary team practitioners are much less accustomed to addressing socio-technical issues associated with the introduction of new technologies and achieving desired culture changes enabled by them. Figure 4 indicates the progressive flow of the techniques to in the MAST Team. It makes Agile frame to insert itself in the process and yet to follow all the progressive steps without delay in project completion time. It is in answering the socio-technical issues that MAST Team stands to make its greatest contribution to agile manufacturing. Improved profitability has been sufficient justification for the incorporation of Multidisciplinary team principles into modern software engineering techniques. Through both industrial ergonomics and software engineering, Multidisciplinary team was incorporated into an existing manufacturing paradigm either from necessity or from the need to increase productivity and profitability. As a new, rapidly evolving paradigm, agile manufacturing offers Multidisciplinary team ergonomics professionals an opportunity that never existed with traditional manufacturing. This approach has advantage over traditional testing procedure as, minor issues are taken care at the team level itself, killing the communication delay. It is always a challenge to form a team which is superior in all respect. More over much of human factors shows itself in large-scale projects rather in small time product or software development.

6. Conclusion

Bringing together people from different environments with diverse goals requires planned facilitation to enable proper teamwork, but the team can work to produce results in an agile manner under the right working conditions. The main components of their success are having the right team members, working in a transparent way in workshops and using lo-tech tools

to visualize all ideas and decisions. Another important step here is to adhere to standards of documentation and procedural standards. MAST team can be unpredictable if procedures are not followed as discussed. MAST Team can be formed in small number or many team of small size also can be done depending on the amount and duration of the testing process. Software tested with this approach will be of highly trustable and effective when installed. More research need to be done on issues in adopting this team approach. Also study can initiated on newer techniques in Agile development framework.

7. References

- [1]. Martin, R. C. (2003). Agile Software Development: Principles, Patterns, and Practices Ioannis G. Stamelos, Pagagiotis Sfetsos: Agile Software Development Quality Assurance Information Science Reference
- [2] Lisa C, Bob G (2009). Agile Conference “The Testing Stage” Sessions proposal, <http://agile2009.agilealliance.org/testing>
- [3] Mieke Haesen, Karin Coninx, Jan Van den Bergh, and Kris Luyten. MuiCSeR: A Process Framework for Multi-Disciplinary User-Centered Software Engineering processes. In Proceedings of Human-Centred Software Engineering
- [4] Harish R, Madhu B K, Lokesha V, “ A Sophisticated Study on Best Practices of Agile Software Testing”, IJECCE, Volume 3, Issue (1) NCRTCST, ISSN 2249-071X.
- [5] International Standards Organization. ISO 13407. Human Centred Design Process for Interactive Systems. Geneva, Swiss, 1999.
- [6] D. Redmond-Pyle and A. Moore. Graphical User Interface Design and Evaluation. Prentice Hall, London, 1995.
- [7] John M. Carroll. Making use : scenario-based design of human-computer interactions. MIT Press, Cambridge, Mass. [u.a.], 2000.
- [8] John Pruitt and Tamara Adlin. The Persona Lifecycle : Keeping People in Mind Throughout Product Design. Morgan Kaufmann, 2006.
- [9] Karen Holtzblatt, Jessamyn B. Wendell, and Shelley Wood. Rapid Contextual Design: A How-to Guide to Key Techniques for User-Centered Design (Interactive Technologies). Morgan Kaufmann, December 2004.
- [10] Dyba T, Dingsøyr T (2008). Empirical Studies of Agile Software Deve: A Systematic Review, Information and Software Technology, doi: 10.1016/j.infsof.2008.01.006
- [11] Gitte Lindgaard, Richard Dillon, Patricia Trbovich, Rachel White, Gary Fernandes, Sonny Lundahl, and Anu Pinnamaneni. User needs analysis and requirements engineering: Theory and practice. Interact. Comput., 2006.
- [12] Elisabeth Hendrickson.: 2005: Agile Testing
- [13] Judith Brown, Gitte Lindgaard, and Robert Biddle. Stories, Sketches, and Lists: Developers and Interaction Designers Interacting Through Artefacts. In Proceedings of Agile 2008, 2008.
- [14] Madhu B K and Lokesha V: Mathematical Modeling and Analysis of Agile Software Testing National Conference on Applied and Engineering Mathematics (NCAEM – 2011), RNSIT, Bangalore, Karnataka, India
- [15] Brad A. Myers, Sun Young Park, Yoko Nakano, Greg Mueller, and Andrew Ko How designers design and program interactive behaviors. In VL/HCC, 2008.
- [16] Brian P. Bailey, Joseph A. Konstan, and John V. Carlis. Demais: designing multimedia applications with interactive storyboards. In ACM Multimedia, 2001.
- [17] Mark W. Newman and James A. L. Sitemaps, storyboards, and specifications: A sketch of web site design practice. In ‘DIS 2000 Designing Interactive Systems. ACM Press, 2000.
- [18] Jeffrey Nichols and Tessa Lau. Mobilization by demonstration: using traces to re-author existing web sites. In IUI '08: Proceedings of the 13th international conference on Intelligent user interfaces
- [19] Corrie van der Lelie. The value of storyboards in the product design process Personal Ubiquitous Computing.
- [20] Ron Sova and Deborah Hinderer Sova. Storyboards: a dynamic storytelling tool. Technical report, Sova Consulting Group, Tec-Ed Inc., 2006.
- [21] Jan Meskens, Jo Vermeulen, Kris Luyten, and Karin Coninx. Gummy for multiplatform user interface designs: Shape me, multiply me, fix me, use me. In Proceedings of the working conference on Advanced Visual Interfaces, AVI 2008, New York, NY, USA, 2008.
- [22] Ashby, M. R., and Lin, H. W. (1994). *Interactive collaborative environments (ICE): Platform independent X application sharing and multi-media over wide area networks at Sandia National Laboratories*. Presented at the Fifth International Symposium on Robotics and Manufacturing, Maui, HI. Bloom, H. M. (1994). *Technical program description: Systems integration for manufacturing (SIMA)* (Report NISTIR-5476). Gaithersburg, MD: National Institute of Standards and Technology.
- [23] Brost, R. C., Strip, D. R., and Eicker, P. J. (1992). The technology base for agile manufacturing. In *Proceedings of the NASA Workshop on Space Operations*. Houston, TX: NASA. Donlin, M. (1991, April 1). Data management tools tie frameworks to concurrent engineering. *Computer Design*. Forsythe, C., and Ashby, M. R. (1994). *Developing com*
- [24] C. Larman, Agile & Iterative Development: A Manager’s Guide. Addison-Wesley, 2004.
- [25] J. Stapleton, Dynamic systems development method – The method in practice. Addison Wesley 1997.
- [26] Beck K. et al.: Manifesto for Agile Software Development, The Agile Alliance, (February 2001), <http://www.agilealliance.org/>
- [27] Cockburn A.: Characterizing People as Non-Linear, First-Order Components in Software Development, Humans and Technology Technical Report, TR 99.05, (October 1999)
- [28] Fowler M.: The New Methodology, MartinFowler.com, (March 2001).