

VIEW POINT

Staying Agile in a Distributed Environment



- Anupama Rathi, Prasad Prabhakaran, Amol Patki, Ketan Navinchandra Shah

Executive summary

The agile methodology has been widely used in software development and testing for over a decade now. In the increasingly globalized economy, many organizations have development teams distributed across multiple geographic locations. This distributed environment poses certain challenges to agile development. However, a mature approach and efficient management of a few key aspects can make the agile methodology work successfully towards its goal of rapidly delivering superior software quality even with teams spread across distant locations and time zones.

Change is the only constant

Introduced in the Agile Manifesto in 2001, the agile approach to software development has proven successful in meeting business objectives and improving software quality across industries. Quintessentially highlighting the importance of flexibility, incremental development, and close collaboration among cross-functional teams, the approach overcomes the limitations of the traditional waterfall approach and efficiently addresses the challenges introduced by changing requirements. The focus in agile is on iterations and rapid delivery of working software. The key is to respond to change with agility.

While the agile approach allows the development team to respond quickly to constant change by interacting closely right through the development cycle, it has been seen as best suited for teams working in the same location. With the right approach, however, even distributed teams can deliver the benefits of agility with success.

Challenges to being apart and agile

From its inception, the agile approach assumes that the entire team is located in the same place. However, in reality, many businesses need to deploy geographically distributed teams for software development. The key drivers for this situation are the rising operational costs and the scarcity of talent in the market. Such distributed teams are often separated by office locations, cities, countries, time zones, and even continents. When these teams work on engagements that call for the agile approach and practices, they face challenges on multiple fronts.

Agile principles were based on co-location, close working, trust, and reduced need

for handovers and other time consuming activities. Being distributed means that communication can lead to miscommunication, trust building is more difficult, and ineffective handovers add to the overhead. Further, reduced time overlap can mean added delays which counter the agile advantage of faster. The challenge lies in how we make agile work in a distributed ecosystem efficiently and without inflating overheads.

To overcome these challenges, in addition to enhanced distribution and balanced sharing of work across locations, the management needs to ensure that agile principles are followed and the spirit of trust, discipline, and collaboration is kept alive within the ecosystem.



Agile steps to success

Every component of distributed agile development – including developers, processes and the technologies – has a critical role to play in addressing these challenges. The following factors need to be considered carefully for the successful application of the agile methodology in a distributed environment:

Size matters – so does complexity



The context of a distributed environment helps anticipate the complexity and related challenges. The understanding and implications of various parameters, such as the size of the project and the level of distribution, can bring to notice areas that need focus and attention. For example, a large program or a portfolio execution in a distributed agile environment needs to be treated differently as opposed to a relatively smaller project.

Project-level distribution may bring forth challenges related to collaboration with a stronger need for distributed agile governance. Measurements and metrics, dependency management, release management, as well as stakeholder mindset are a few other important facets for consideration when distribution is in a larger scope of program or portfolio. Addressing all these aspects and evaluating the impact of distribution on the agile approach is crucial for the success of distributed agile engagements.

What goes where?



Once the context of distribution has been identified, the next key step is to focus on the distribution of work among the agile teams. Based on the requirements of the project and the kind of development model used, the work needs to be distributed by features or by project activity. A higher degree of vertical distribution (for example, by features) offers greater chances of success for the distributed agile methodology.

The extent of distribution is another vital factor here and some metrics need to be identified with sufficient clarity: the number of geographic locations across which the team members are spread, the time overlaps among these locations, and co-located or distributed teams within a particular location. For the distributed agile approach to succeed, intelligently designed activity maps should be based on industry best practices such as:

- Co-locating members from horizontal teams to the extent possible
- Single location for the end-to-end delivery of complex pieces, or activities involving intense collaboration
- Defining hand-over policies clearly in order to reduce dependencies across locations.

Right team makes strong foundation



Distributed agile engagements demand detailed attention to the formation of the team. For example, if a scrum master, product owner, and the scrum team are spread across different locations, challenges involved in collaboration, communication and decision-making can intensify. In a situation where the scrum master is at a different location than the scrum team, it is advisable to assign the role of a scrum master to one of the team members.

Similar decisions need to be made when product owners and the architecture teams are located at different sites. However, the emphasis here is not on adding more roles to each team, but on building a team which is fully capable of performing cross-functional activities, and on creating an ecosystem where unavailability of certain roles at certain locations does not impede the progress.

Staying in touch



The tools used for Application Lifecycle Management (ALM) in the execution of a distributed agile project often play the role of system of records, whereas the agile teams across locations need a system of engagement. Distributed agile teams need ease of communication and collaboration to maintain their feedback loops. Organizations, therefore, need to implement intelligent systems of engagement with a focus on effective communication and collaboration. Such systems should help in retaining the unstructured format of data – which is generated due to free-flowing communication – in a more contextual manner for future reference.

Distance is no excuse



The practices that form the essence of the agile methodology are critical in a distributed agile environment. These practices include building trust, communicating regularly, using feedback loops, providing visibility, and encouraging conversations, among others. Often, due to insufficient time available during the common time slots, teams in overlapping time zones tend to skip many of these agile practices. However, following agile practices through daily stand-up meetings and retrospectives is far more important in a distributed environment than in a case when the teams are located at the same place. The distance and geographic diversity can increase the risk of communication failure, gaps in work, lack of appropriate hand-over, cultural misunderstanding leading to dilution of trust, and conflicts due to diverse leadership styles.

Therefore, to ensure the success of a project in a distributed environment, it is of paramount importance that agile practices are followed meticulously.

Getting the nuts and bolts right



Tools for automation, continuous deployment, and well-defined procedures and policies for release contribute greatly to quality in an agile project irrespective of the environment – distributed or located in the same place. However, in a distributed environment, losing even a day's work due to lack of infrastructure or related support services can seriously slow down the delivery of the working software at the end. Therefore, it must be ensured that all locations have access to adequate infrastructure including automation tools. Further, all teams must faithfully follow the procedures and policies laid down. To enable this, organizations need to invest in the right tools and define robust policies for using shared infrastructure.

The agile way forward in a distributed environment

Implementing the principles of agile methodology for software development in a distributed environment may initially appear to be an obstacle course. However, the key to success lies in never losing the sight of the agile practices through collaboration and communication. Due emphasis on the spirit of agile and its principles along with mature understanding of distribution, well-thought-out work allocation, and wise investment in the appropriate infrastructure across locations are the keys to success for any agile project in a distributed environment.



For more information, contact askus@infosys.com

Infosys®

© 2015 Infosys Limited, Bangalore, India. All Rights Reserved. Infosys believes the information in this document is accurate as of its publication date; such information is subject to change without notice. Infosys acknowledges the proprietary rights of other companies to the trademarks, product names and such other intellectual property rights mentioned in this document. Except as expressly permitted, neither this documentation nor any part of it may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, printing, photocopying, recording or otherwise, without the prior permission of Infosys Limited and/ or any named intellectual property rights holders under this document.

Stay Connected    