

Key Challenges and Tips in **Managing Distributed Agile Teams**

By *KINSHUK JHALA*

CONTENT

1. Introduction 3

2. Growth of distributed Agile teams and key benefits 4

3. Seven key challenges in managing distributed Agile 6

4. Tips to succeed at managing distributed Agile teams 8

5. Conclusion 13

6. About the Author 13

7. About TO THE NEW 14

8. References 14

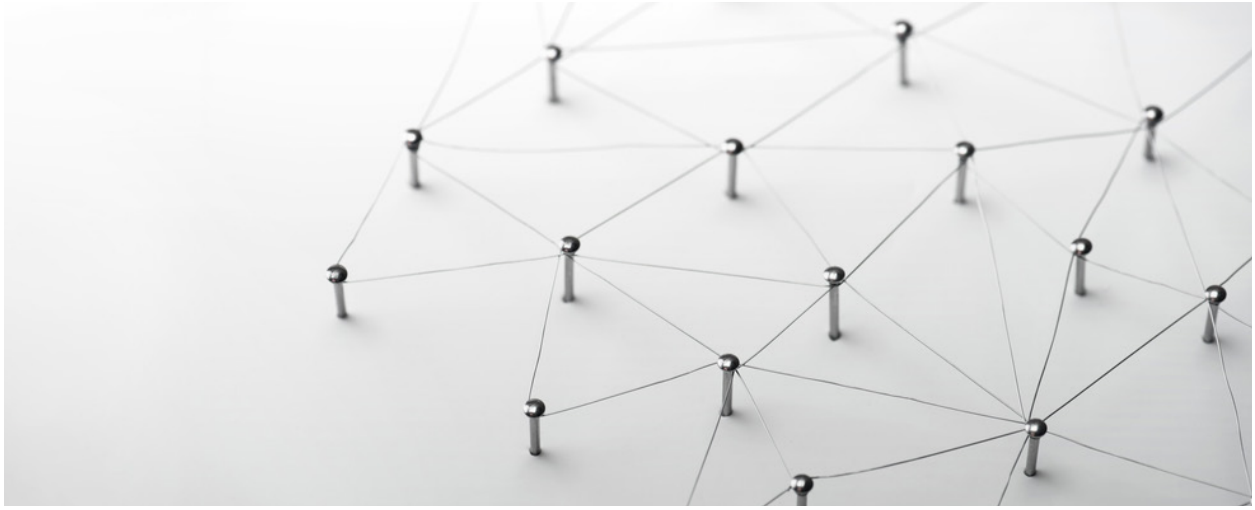
1. INTRODUCTION

Developing software products is full of challenges. Companies venturing into this space need to understand the demands of digitally empowered customers, evolving technology landscape and rising competition. Most companies find it challenging to develop skillsets required to build products in-house. This is mainly because there are new advances in digital technologies and developing the required skillsets is a time-consuming and costly affair.

Owing to its many benefits, outsourcing product development to remote or distributed teams is one of the most cost effective solutions companies eventually opt for.

Certain countries such as India, China, Malaysia, Indonesia and Philippines are most preferred outsourcing destinations. According to The 2014 A.T. Kearney Global Services Location Index, the top three factors basis which these countries were measured include financial attractiveness, people skills and availability and business environment.

While outsourced product engineering to remote partners or offshore centers continues to grow, there are multiple challenges that companies face in managing teams that are distributed across multiple locations. Some of the burning challenges include differences in the time zones, miscommunication or limited flow of communication and information, trust issues, motivation problems and much more.



Our whitepaper outlines the advantages of having an offshore Agile product engineering team, challenges in managing distributed teams and and remedies to overcome such challenges.

2. GROWTH OF DISTRIBUTED AGILE TEAMS AND KEY BENEFITS

Legacy organizations used to follow waterfall development model which was more rigid and structured. Companies used to incorporate feedback from stakeholders after the final product was developed. This led to multiple iterations thereby increasing the time to market.

With slow time to market, these traditional companies started losing out to competitors. The need to embrace Agile was quite clear. . Moreover, a shift from waterfall to Agile was also on account of evolving customer behavior and the growth of digital technologies.

Agile not only is a key enabler to business transformation, but also a great way to manage project teams. Agile development has multiple benefits such as :

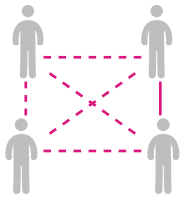
- ✓ Faster time to market
- ✓ Shorter release cycles
- ✓ Early and constant feedback cycles
- ✓ Customer Satisfaction
- ✓ Iterative development
- ✓ Early mover advantage
- ✓ Quality and Visibility
- ✓ Rapid product experimentation



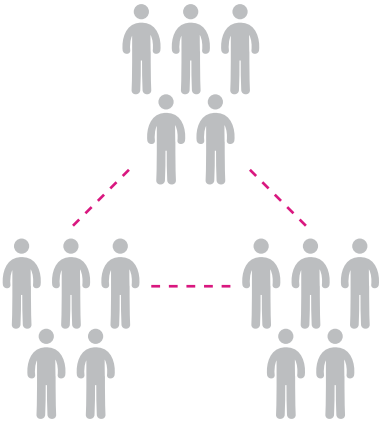
With more and more companies blending Agile methodology with remote development, there is an increase and growth in distributed Agile teams working on a project from different locations. According to one of the recent surveys by Atlassian, 72% of respondents said that some of their team works remotely. Leveraging these distributed teams, companies to gain offshore competitiveness and deliver products faster and cheaper.



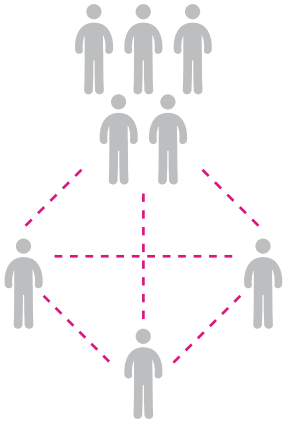
Co-Located



Fully Dispersed



Distributed Subteams



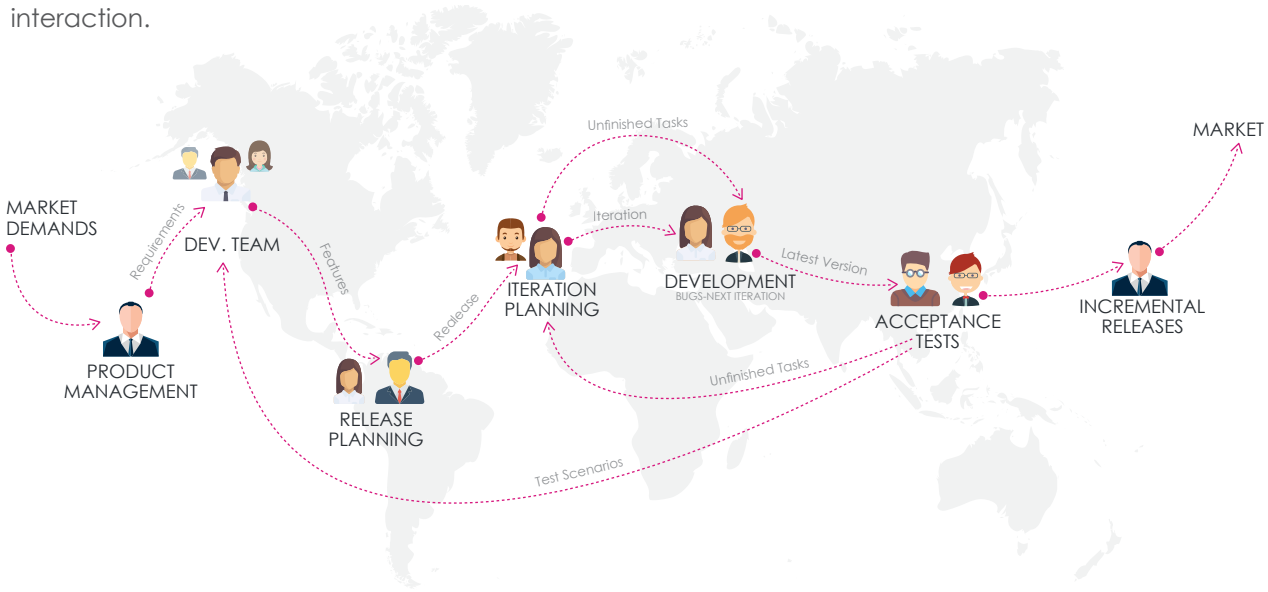
Partially Dispersed

OUTLINED BELOW ARE SOME REASONS FOR THE NEED OF HAVING DISTRIBUTED AGILE TEAMS:

- Globally distributed teams are cost-effective
- Distributed teams expand the reach and provide an access to new markets
- "Follow-the-sun" support model helps to increase responsiveness and reduce delays even when the teams are distributed across different time zones
- Distributed teams can promote cross-country culture contributing to growth and thought leadership
- Highly skilled countries give an access to skilled resources
- Increased business efficiency

4. SEVEN KEY CHALLENGES OF MANAGING DISTRIBUTED AGILE TEAMS

There are various benefits of distributed product engineering, but at the same time managing remote teams is a hard nut to crack. This is mainly because collaborating with teams that are spread across geographies involve a lot of issues related to trust, transparency, miscommunication and continuous interaction.



OUTLINE BELOW ARE SOME OF THE CHALLENGES IN MANAGING DISTRIBUTED AGILE TEAMS:

1. Building rapport and giving one-team feel

One of the biggest challenges in driving the offshore teams is building rapport between the co-located team and the offshore team. Project managers find it really difficult to build the same level of trust with remote team members and provide them with a one-team feeling.

2. Timezone and language differences

With remote and distributed teams working differently in different timezones, it is difficult to have an overlap and maintain the delivery timelines. Scheduling the activities and distributing them across time zones is full of challenges and might result into delays. Moreover, with difference in time zones, it is very difficult to get a response to query in real-time. Apart from time-zone differences, language barriers also create problems. It becomes difficult to communicate with multi-lingual teams without the help of translators.

3. Biasedness towards co-located teams

Although this is not true in many cases, at times, project leads feel biased towards co-located teams and tend to give them more benefits, perks and appreciation than those working remotely. With this biased attitude, remote workers might feel demotivated resulting into poor performance.

4. Regular flow of communication and access to repositories

While some of the distributed Agile teams have started using skype, JIRA and other tools, many teams still struggle to pass on regular information in a seamless way to their peers working remotely. Improper audio / video setup impacts communication. Some of the common problems are not having an access to call, difficulty in identifying the speaker, limiting conversation using round-robin technique, and encouraging participants to check on mutual agreements and disagreements Moreover, the problem isn't just restricted to accessing the information, but also the code repositories. Some teams do not have the access to the code leading to delays. Distributed Agile projects require an effective tool chain to gather requirements, build setup, code review, track bugs,, project management and repositories.

5. Feedback incorporation

Understanding feedback and incorporating it timely in the project is mission critical to make Agile product engineering projects successful. Remote teams sometimes miss out on incorporating the feedback received due to information leakage and communication gaps. Moreover, at times the central team or coordinating project manager also forgets to share the entire feedback to the distributed team members. This hampers product delivery and leads to multiple iterations. In certain cases, as the teams aren't able to openly collaborate (given their remote placements), when their feedback is not incorporated, they feel under-valued.

6. Cultural differences

Distributed Agile team members belong to different nationalities which need to be taken into account before organizing global meetings. For an instance, constructive criticism might be looked upon differently in different countries. In certain countries, colleagues might feel hurt and think it as a rude behavior. In some countries, sprint plans and deadlines are taken up seriously and there is limited flexibility in the attitude of colleagues to consider changes down the line; whereas some countries in the west take it easy.

7. Middle management

At times, distributed teams have a dedicated local leadership, managing the remote team independently. This conflicts with the principle of flat structure required in Agile projects. In case of a middle management which is not easily accessible, project members often fail to raise their concerns thereby impacting build quality, performance, and overall morale of the employee.

4. TIPS TO SUCCEED AT MANAGING DISTRIBUTED AGILE TEAMS

There are various ways to resolve team management challenges in distributed Agile projects. Outlined below are some of the ways to address the above mentioned challenges:

1. Team formation

Before managing the team, formation of team takes the larger focus. Below are the different approaches for organizing distributed Agile teams:

Team Approach	Advantages	Disadvantages	Considerations
Component	Reduces communication between sub-teams	Requires a loosely coupled, highly cohesive architecture	Use for components or subsystems that are highly cohesive and loosely coupled
	Enables teams to build technical expertise specific to their component(s)	Functional dependency management can be complex	Use for technical-oriented components, such as a security framework or dat base encapsulation services, which require technical expertise
		Requirements need to be aggregated by component	
Feature	Enables teams to focus on a subset of the business	Requires common development conventions	Use for complex LoBs or applications which require developers to have deep understanding of the problem domain
	Potential to make it easier to assign features to teams	Requires sophisticated configuration management	
		Technical & requirements dependency management can be complex	

Team Approach	Advantages	Disadvantages	Considerations
Functional	Enables functional specialization of individuals	People motivated to be specialists, not generalizing specialists	It often makes sense to have an integration team responsible for integrating the entire solution and testing it end-to-end.
	Comfortable approach for people with deep experience with traditional approaches	Significant communication overhead between teams	Support a "follow-the-sun" strategy where development is performed in one location and testing in another
		Requires more documentation and reviews thereof	
		Requires greater management oversight	
		Increases overall project and organizational risk	
Internal open source	Supports development of shared components or subsystems	Requires other teams to provide developers, at least on a part time basis, to work on this effort	Apply in organizations with developers with deep experience with "public" open source efforts
	Spreads the cost of building these components across several development teams	Requires management and governance structures which reflect open source development	

Product owner needs to form a team of more than 7 to 10 people distributing the skills sets required to execute the project. It is advisable to encourage feature teams over component teams and every team should possess different skill sets so that each team can deliver the desired user story across each distributed location.

2. Pre-Sprint planning

While planning the distributed team, also consider having a separate backlog for multiple teams. Along with team formulation, product owners also need to take into account sprint length, velocity and release planning.

Distributed teams should estimate the stories they will work on as different teams have different scales. They also need to sync their sprint lengths with each other to ensure their interdependencies are met.

Moreover, apart from syncing up within themselves, the non-Agile team should also be aware of the priorities of the Agile teams. This will increase the visibility to clean the roadblocks early.

With regards to the release plan, the project head will ideally identify all the perils related to the project and the teams along with mitigation plan for the identified risks.

3. Product owner meetings

Coordination among product owners is extremely important. They need to regularly meet to discuss backlogs, dependencies, and user stories of different teams. Followed by these meetings, scrum manager and project manager keep meeting virtually to display the vision, sprint goal and estimates.

4. Invest in tools to improve communication and development

Distributed development demands using enterprise tools for managing teams and their work to ensure the product is developed according to the roadmap. Companies have started investing in test automation suite, Continuous Integration tools, audio/video conferencing systems and other required toolsets facilitating faster time to market.

5. Rapport building

Having a good rapport across cross-country teams is extremely important. Personal rapport increases trust, boosts morale and minimizes conflicts. It is vital for a project manager to know the delivery team of remote offices. Most Agile companies use collaboration tools to spread information about each other seamlessly. It is also a good practise to start a blog and let all the employees share their use cases or knowledge through the blog. Project manager or Scrum Manager can also introduce newcomers to others in the team through an email or video conference call. Face to face meetings are one of the best ways to build a rapport. Scrum manager should bring everyone together at least once in few months if logistics if not a very big problem. This will help him to understand the challenges faced by the co workers, interdependencies between different geo teams, and helping them to resolve the same.

6. Increase the morale

Improving the distributed team's efficiency isn't just about meeting them and solving their challenges. It is important to boost their morale. Keep sharing all the vital activities among both onshore and distributed teams. Such activities include idea generation, conceptualization, build quality checks, knowledge sharing sessions and so on.

7. Get usability feedback from local teams

Sometimes products that are developed have different usability test results depending on the geographies. It is always advisable to get the product functionalities tested with local audience first and then have a second level of full test lab feedback at the headquarters. Moreover, if product is

tested first at the HQ, it becomes difficult to share results through screen sharing software and webcams to the distributed teams later.

8. Enhance code knowledge through reviews

Code reviews become very crucial for distributed Agile projects. As the remote team members and co-located team members work in different timezones, sharing code for reviews between the two teams makes it possible to resolve issues in production faster when one of the teams is away. Without having knowledge about the complete code, it becomes impossible to resolve issues faster without the help of the other team.

Modern teams also practise continuous integration so that the build runs regularly enabling the teams to identify issues early and fix them. CI also helps to identify design related issues and cover more scenarios to test. Project managers can also use alerts and notifications to ensure that the teams are aware about the updates in the delivery pipeline.

9. Create guidelines

Creating guidelines helps to find the gaps and resolve them well in time. As there are multiple interdependencies in a typical distributed Agile project, it is crucial that tasks are only marked complete when they actually are. For instance, when executing a release that involves multiple teams, define what it means to be "Task completed": code pull request created, code reviewed, code merged to respective branch and so on.

Create documentation where required for the reference of distributed teams. Make sure they are able to track bugs effectively using right tools and software.

10. Distributed daily scrum meetings

The agenda of distributed daily scrum meetings is to have a 15 minute sync up everyday with remote teams. In such a meeting, project manager uses the kanban board, project management tool or similar software to let the team members know about the tasks accomplished.



If the distributed teams are all spread across in various time-zones, speaking different languages, there are four different methods that could be used such as Daily Scrums through documentation, liaison approach, alternating meeting times, and share the pain.

The project manager and scrum manager need to ask the following three questions to the team every day:

- What did you accomplish since the last meeting?
- What are you working on until the next meeting?
- What is getting in your way or keeping you from doing your job?

Both the onshore team and distributed teams need to clearly communicate the answers so that it adds value and if there are issues, they are resolved quickly.

With the help of daily sync ups, it is easy to verify the progress and quickly escalate the issues in the blocker list to concerned superior.

11. Follow test driven development

Unit tests are the small tests that fail first. Developers should document these tests and share them with other teams so that the time to fix defects reduces. Test driven development enables distributed teams to remain Agile and collaborate with each other efficiently.

12. Mid Sprint / End of sprint reviews

Reviewing the project after end of an every sprint or at times even in the middle of a sprint is vital. Such a review should be conducted in a way that there is an overlap of time zone between offsite and onshore teams. If there are multiple teams and timezones, few teams could be clubbed together and for the remaining team, a separate review meeting could be held.

During such meetings, distributed teams should keep taking notes or record the sessions. Make sure that the tools are tested in advance to avoid any delays.

13. Retrospectives

For distributed teams, joint retrospectives can be really helpful. Team members can comment on release quality and capability, milestones, ways to improve future releases. While conducting the retrospectives, different cultures should also be taken into the account. Asking for a chance could be interpreted differently in different cultures. The project manager should also set an action plan in advance so that all the points pertaining to the growth of product development are covered.

5. CONCLUSION

Agile is all about reducing the time to market with more iterations, adapting to real-time business needs. Distributed Agile is an extension to Agile. It applies all the Agile principles to projects executed by distributed teams placed remotely at different locations or at times even on two different offices in same city operating in different timezones.

More and more companies are leveraging distributed development, but there are multiple roadblocks that hamper the project's success. Communication barriers or misleading communication tops all the barriers. With rising communication gaps, remote teams start feeling left out and this quickly impacts the overall development plan. It is highly important that distributed teams have a one-team feeling. Apart from communication, resolving problems quickly is also a challenge. Join retrospectives, kick off at co-location and sprint reviews can surely help to resolve problems in time. Ultimately, a satisfied distributed team will lead to bug-free product development, increased customer satisfaction and a higher ROI.

6. ABOUT THE AUTHOR

Kinshuk Jhala, Deputy Manager Marketing

Kinshuk is a passionate marketing technologist by day. He loves writing about various digital advances such as Cloud, Big Data, IOT, Mobility, Analytics and more. He is also passionate about using new age digital media vehicles for increasing the overall marketing effectiveness. He has spent reasonable years in the technology sector working with inbound teams on various branding and marketing projects.

7. ABOUT TO THE NEW

TO THE NEW is a digital technology company that builds disruptive products and transforms businesses. We leverage the power of experience design, cutting-edge engineering, cloud and analytics led marketing to enable Digital Transformation.

Our passionate team of 750+ people includes passionate technologists, digital analytics experts, video specialists and creative mavericks who have transformed businesses of more than 300 companies spread across 30 countries worldwide. We take pride in our culture which is driven by passion for making an impact through technology.

8. REFERENCES

• Agile India

http://agileindia.org/uploads/downloads/IntroductionToDistributedAgileDevelopment.pdf

• Atlassian

https://www.atlassian.com/agile/remote-teams

• Scrum Alliance

https://www.scrumalliance.org/community/articles/2013/july/managing-distributed-teams

• Disciplined Agile Delivery

http://www.disciplinedagiledelivery.com/agility-at-scale/geographically-distributed-agile-teams



info@tothenew.com
www.tothenew.com

Lets Connect