

Use of Agile Development in Large Organizations

In usual plan-driven approaches (like waterfall model), more time is spent on planning the whole process instead of doing the actual implementation. Though it might be essential for safety critical applications and large geographically dispersed teams, it increases a lot of overheads in small teams. Especially when requirements are not completely known beforehand.

So companies now use **agile methods** which allow the team to focus more on writing the code and less on bureaucracy or documentation. Agile methods are based on an iterative approach that focuses on customer involvement, people instead of process, responding to changing requirements and maintaining simplicity (Sommerville 2016).

Benefits of agile methods

Large companies dealing with complex software generally produce lots of documentation as a side product, which consumes a lot of time and effort. Agile methods cut that part and prioritize working software over extensive documentation (Sommerville 2016).

There are short cycles (sprints) with each cycle having a milestone that adds more functionality to the software. The backlog is also added to the next milestone. This makes the whole process fast and allows for early deliveries.

Agile methods involve customer representatives directly in the development process. This makes it a lot easier to get early feedback and responding to any changes in requirements.

Drawbacks

Agile methods focus on small teams. But it doesn't work very well for large teams because a proper communication is needed among the whole development team. So scaling is an issue here.

It requires direct customer involvement, which might not be feasible at times.

Since agile methods are not plan-driven, there might be problems in developing very complex software.

It requires skilled programmers and managers for these methods to work well. But in a large organization, all kinds of people are there. So this might make the whole process slow. And risks of failure are increased as well. (Sommerville 2016)

Since people in a company might be used to older plan-driven approaches, they might be reluctant to adopt a new method.

Adoption problems

An organization with a wide range of skills and abilities needs to train its developers to work in a new way. It takes some time to get used to a new method of development.

Since sprints are usually run for a very short time period (say 1 week), all the tools must be fast enough. For example, build or automated testing shouldn't take too long. So new tools and experienced people to operate them are required. (Berman, no date)

Generally, a large organization has multiple cross-functional teams, so it becomes difficult to manage them as agile methods are designed to work for a small team. Also, proper communication tools are needed too.

A quick feedback needs to be available all the time to the developers for continuous integration.

Tips

The process should be customer-centric. That is, the customer or customer representatives should be directly involved in the development process.

Proper communication is essential. There should be right kind of tools for teams to communicate. And regulars meetups should be organized where the feedback for previous work is taken and problems are discussed. And the next aim is decided.

Basically, everyone should have a broader view of what's going on in the process.

References

SOMMERVILLE , I., 2016. Software Engineering. 10th ed. Boston: Pearson.

Poppendieck, M., Cusumano, M. A., 2012. Lean Software Development: A tutorial. In IEEE Computer Society.

Berman, J., no date. Agile at Scale [online]. Available from:
<https://www.atlassian.com/agile/agile-at-scale>