Table of Contents

What is the Role of a (Development) Team?

The Team does everything to win the game within the project boundaries – to deliver the project.

The Team is cross-functional, which means the full know-how to realize the product is located in the Team. The Team needs to understand the project and sprint vision, goals, etc. of the Product Owner in order to deliver potentially shippable product increments.

While it is best to have all functions represented on the Teams, there are often cases where this cannot be. For example, we may not have certification, planning or documentation representation on some Teams. This does not absolve the Team from these responsibilities. *Design, development, test and documentation are activities, not people*, and the Team takes full responsibility for getting the work done to known quality standard. If the Team does not have not expertise in an area, the Team needs to learn it in order to perform the activities necessary to complete the product.

Responsibilities

- Cross-functional, typically seven people plus or minus two members
- Based on the prioritized list selects the set of user stories they will work on in the Sprint / Iteration and specifies work results
- Has the right to do everything within the boundaries of the project guidelines to reach the Sprint / Iteration goal
- Organizes itself and its work
- Demonstrates the work results

What We Don't Want to See

- Teams that are too big: Effective Teams are effective because they communicate well among each-other. As you add more people to the Team, you increase the number people the Team has to communicate with and so reduce the potential effectiveness of the Team.
- Teams that are too small: Effective Teams need to feel like they are contributing. If the Team is too small, then it is sometimes hard to feel like a real contribution is being made, reducing their effectiveness. At the extreme, a Team of 1 really does not make sense - software development is a team sport.
- People telling the Team how much work they should take on: To get a real view of what is possible, to the quality standard, you have to let Teams determine how much work they take on. If you push a commitment on the Team you should not be surprised that the commitment is not met or that corners get cut in an effort to meet the commitment.
- People telling the Team how they should work: The people doing the work typically know how to do their job. If they don't then they should ask for help. Note that this does not mean that mentoring is not required. This statement is more about day-to-day management of the people on the Team.

- Pure "development" team: Even if the Team is made up entirely of development people, the Team is still responsible for building "done" software. In order to do this, development heavy teams need to seek mentor-ship from QA, functional design and documentation people, for example.
- Remote teams where there are 12 hour time-zone differences between team members: While this is unavoidable (e.g. Product Owner is located near the business rather than near the development center, or there are people on the team that need to be trained) a goal of Team formation should be to help Team members communicate. This means having Teams be as close to each-other physically and, if there is separation, reducing the effect of time-zones as much as possible.

Consultant, Tools, Team, Role, DevelopmentTeam, FAQ

From:
https://www.hanssamios.com/dokuwiki/ - **Hans Samios' Personal Lean-Agile Knowledge Base**

Permanent link:
https://www.hanssamios.com/dokuwiki/what_is_the_role_of_the_team

Last update: **2020/06/04 11:58**