How to Measure Performance of Your Outsourced Developers



Focus on the outcome, not the output.

April 20, 2019 5 min read

Opinions expressed by *Entrepreneur* contributors are their own.

A common challenge founders face when they outsource to a software development company is tracking the performance of developers. Since you cannot really monitor the outsourced team on a full-time basis, it is critical to know whether your investment in hiring developers is justified.

Performance, being a qualitative trait, poses measurability challenges. There is no single metric which would lead you to a definitive answer. Knowledge about the goals to track and the areas to focus on can bring you closer to performance measurement of the outsourced developers as a whole.

Related: 3 Ways You Can Avoid Wasting Your Company's Money on Technology

Here is a framework that can give you the much-needed insights into the performance of the outsourced tech team.

1. Start with conceptualization of goals.

Setting down the organizational goals and making sure all the teams adhere to the goals specified should be made a priority as soon as your company starts out. Even if the development is outsourced, the tech team isn't a standalone entity with the sole function of writing lines of code and testing them. Integration of the tech goals with the overall business goals is a crucial step in your journey. Conceptualization of goals at the initial stage lays down the basis of measuring the performance at the later stages.

For example, if a company's goal is generating more revenue, the tech goal would be writing the code for the application being developed at increased speed without compromising on the quality of the software developed. Every goal needs to be in line with the overall goals of the business, whether it is at the team level or individual level.

2. Laying down the fundamentals.

Measuring performance fundamentally boils down to two types of questions – whats and hows.

The "what" questions deal with the actual tasks allotted to the tech team. These include both the explicitly stated facts as well as the features that are implicitly expected. An example of a "what" question could be about the tech team's role in following the strategic roadmap, finishing coding and testing the software within the deadline. Here, the timeline

stated is an explicit fact while the requirement of the code being developed to be of high quality is implicit.

The “how” questions are the process related queries. It may be related to how well each developer worked within the team, how innovative their approach to the problem was and so on.

Evaluating the fundamental whats and hows not only gives an accurate insight into what tasks the tech team is performing but also tells you how well they are performing and how effective their process is.

Related: 12 Sites That Will Teach You Coding at a Low Cost, Even Free

3. Focus on the outcome and not the output.

In app development, focusing on the quantitative aspect of the product being developed is never the solution. Developers writing extra lines of code just results in development of a bloated software that poses maintainability challenges. Conversely, minimizing the number of lines of code written doesn’t work out either as it would result in development of a software that is difficult to comprehend and scale.

An outcome-based approach implies that instead of measuring the quantitative aspect of output – the outcome – the software product is evaluated for measuring the performance of the tech team. In order to measure the software development and delivery process, the product being deployed should be evaluated for stability, delivery time and deployment frequency.

A measure of a high performing tech team is the achievement of favorable outcomes. Focusing on these not only enhances productivity and performance, it also helps in achievement of organizational goals leading to higher profitability in the

long run.

4. Balancing productivity with quality.

The productivity versus quality debate is ever persistent when it comes to measuring performance. While the former is a quantitative measure, quality is a highly subjective domain. Productivity is largely output-based and as discussed above, that isn’t the preferred methodology.

So how can you go about balancing productivity with quality? The amount of work done transpires as productivity. For example, in the case of a sales rep, the number of outbound calls made is the measure of productivity, but the number of leads that actually converted and ended up in the sales pipeline is a measure of the quality.

Related: 5 Ways to Be More Productive in 2019 Without Driving Your Team Insane

Similarly while measuring the performance of the tech team, writing the code is productivity while the amount of code that passes through the testing phase into actual deployment transpires as quality. This is, of course, an oversimplification.

John Seddon puts it perfectly: “The paradox is that when managers focus on productivity, long term improvements are rarely made. On the other hand, when they focus on quality, there is continuous improvement in productivity.”

Sourcing a high performing tech team whose methods and functioning match well with your company culture and goals and retaining them for the long run is critical for startup success. Every entrepreneur who aims to be an excellent leader should get adept at identifying whether the resources being invested are worth the results being achieved. Measuring

performance of the outsourced teams is important for the investments being justified.

Conducting periodic performance audits comprise management best practices. This ensures that all the stakeholders within your business venture are working to their best capacity towards the attainment of the organizational goals and directly impacts startup success.

https://www.entrepreneur.com/article/330134