

Jun 22, 2021, 09:03am EDT

The Benefits Of Service Level Agreements In Custom Software Development



Boris Kontsevoi Forbes Councils Member

Forbes Technology Council COUNCIL POST

***Boris Kontsevoi** is a technology executive, President and CEO of [Intetics Inc.](#), a global software engineering and data processing company.*



Measuring the productivity of software engineers is a controversial topic. A traditional service level agreement (SLA) encompasses the scope of agreement-related issues, but productivity remains one of the most disputable, interesting, and actual points. It's a question that's been tackled by research studies and software engineering experts, and while there's no common consensus, there's agreement within the industry that it needs to be done.

When a software project is commissioned, a service level agreement (SLA) is put into place. The SLA is part of the contract between developer and client and measures overall project performance, usually with industry-approved metrics. It's incredibly

important to have an SLA as part of your contract(s) with a software development partner for five main reasons: (1) Establishing measurable levels for all IT services; (2) Setting performance and responsibilities expectations; (3) Documenting communication, governance, and reporting processes; (4) Underlining focus on customer service; and (5) Outlining dispute resolution procedures, awards and penalties.

The SLA usually includes support, regular maintenance, hosting/infrastructure, backups/testing, and in some cases, software processes. But what it never includes is custom software development. The reason why is quite simple: there aren't standard or universal metrics for predictable software development output.

It's time to change that. We need to create the metrics—or more specifically, the process—for which to measure software engineer productivity.

Given the prevalence in software engineering in this day and age, it's no surprise that the lack of measurement is an ongoing issue and topic in the industry. Many businesses have struggled to find a way to measure and track true productivity. No one has completely cracked the code but I do believe we're getting close.

At the most basic level, there are some inherent problems with measuring software engineer productivity. In the early days of coding, managers would measure individual productivity by tracking/counting Lines of Code (LOC). Keeping track of a programmer's output as measured by LOC didn't work because it's short-sighted—it doesn't take into consideration quality or reuse of code. It didn't take long before it stopped being a major metric.

Thankfully so! In modern times we need more sophisticated ways to assess productivity. And more importantly, we need to change the lens through which we measure. I believe all productivity metrics should focus on business outcomes.

A software project is considered successful when the product is delivered and signed off by the client. The "client" can be external or internal, but their satisfaction with the product determines success. An important point to remember as you dive into metrics for your team.

One of the biggest problems with using LOC as a way of measuring productivity is that it's one singular metric. And that's just not how it works. I love the analogy I've heard likening software development to sports. No one stat makes someone an amazing baseball player or footballer, it's the aggregation of stats that paints the full picture.

The same is true in software development. There is no one KPI that can be used to measure development accurately or comprehensively. It's more important to identify and measure a collection of metrics. And the aggregation of metrics should adjust, depending on the specific project and client. Every project's metrics will be different because every project and every client is different!

If the project has tons of features, it makes sense to track how many features each developer ships. Yet, the functional points approach proved to be too complicated. Other metrics, like velocity and bug tracking, are likely to be used frequently, while others might not be used at all. But perhaps the most important metric to capture is quality. It is the most difficult to quantify but it seems we solved this problem already. It's also important standards are created and enforced prior to adding quality metrics to an SLA.

I think we're quite close to adding custom software development to SLAs. There won't be one consistent way of doing it, but that shouldn't be too surprising in the limitless world of software development. After defining your client's determined business outcome, work with the collective team to determine the most appropriate KPIs and metrics to use. Ask the engineers themselves! As the ones working in the trenches, they'll have the best idea of what is most important for each individual project.

SLAs in custom software development are coming soon. Be sure you and your teams are prepared for the shift. And though your SLA is a documented agreement, it doesn't need to be lengthy or overly complicated. Just build one and validate with your client, it's flexible and can be adjusted as needed.



Boris Kontsevoi

President & CEO

Intetics Inc.

Naples, Florida Area

Member Since 2018



Boris Kontsevoi is a founder and President of [Intetics Inc.](#), a leading global software engineering and digital transformation company. Under his leadership, a group of software engineers developed into a truly global technology company with multiple professional certifications and industry awards, including the Global Outsourcing 100, Software 500, and Global Sourcing Association best of class company.

COMPANY INFO

Intetics Inc.



INTETICS MEANS YOUR SUCCESS

Toll Free: +1 (877) SOFTDEV

US: +1 (239) 217-4907

DE: +49 (211) 3878-9350

UK: +44 (20) 3514-1416

Email: contact@intetics.com