

Top-Notch Quality Assurance for a Custom LMS

How implementing end-to-end manual and automated testing enabled an EdTech company to detect **x40** more bugs at the early stage, speed up development cycles **x5**, cut down the cost of fixing errors and bugs **x160**, and bring down overall software maintenance costs by **16%**.

Industry:
Education, Technology

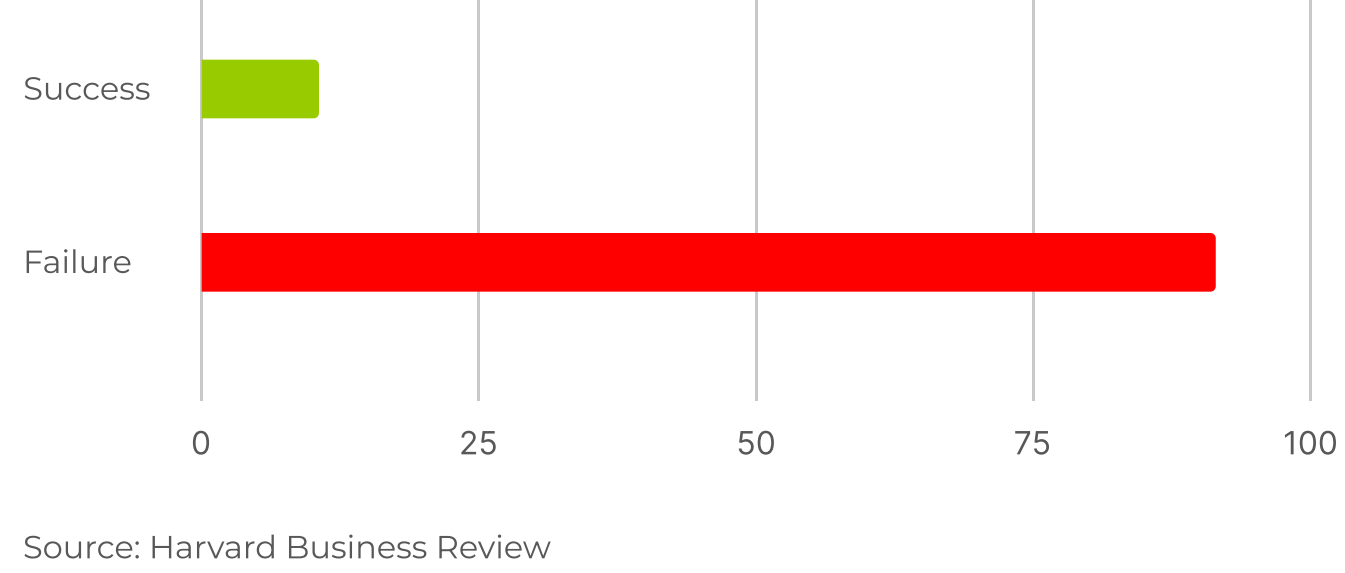
SOFTWARE PRODUCT DEVELOPMENT

SaaS DEVELOPMENT

Business challenge

Only 10% of startups can keep up with a current highly competitive market — and there's a reason for that. Newcomers tend to adopt a 'we can take care of it later' mindset, often skimping on crucial areas of SDLC, such as requirements analysis, project documentation management, software testing, etc.

Startup success & failure rate



Source: Harvard Business Review

Nevertheless, as your solution evolves, these **shortcuts inevitably stab you in the back**, sometimes putting at risk the overall success of your product.

That's exactly where our client found themselves. An **EdTech company providing a custom SaaS LMS** with gamified courses had put **software testing and project documentation on the back burner at the outset**. Several years after the rollout, their e-learning library had ballooned to 40+ courses in 30+ languages, and their initial issues started to backfire. The client's **LMS became unstable, underperforming, plagued with errors and bugs, and expensive to maintain**.

A closer look revealed even **more issues**:

- **The system was overcomplicated and non-transparent, even for the admins.** Managers, not engineers, were steering the project regarding all technical decisions, including tech stack and architecture planning.
- **The platform lacked standard LMS UI elements** like notifications and progress bars to easily track user progress.

Realizing that doing nothing would lead to losing customers, the client decided to **turn things around by adopting software testing, documenting their projects, and enhancing the platform's UX/UI**.

The benefits they aimed to achieve included:

- **More user-friendly** LMS, both for admins and end users
- **Greater** customer satisfaction from enhanced UX/UI
- **Transparent** SDLC processes
- **Higher** software quality due to making QA a mandatory pre-release step
- **Faster** development cycles through automated testing

However, the client's internal team couldn't carry out these tasks on their own due to the lack of **top-tier QA expertise** and their **already heavy workload**. Therefore, they opted for IT staff augmentation services and partnered with ***instinctools**. We provided them with a **senior-level QA engineers and frontend developers**.

Solution

Before getting to user experience setbacks, we focused on adequate functional testing as a straight way to boost the LMS's quality and accelerate development cycles.

01 Setting up manual functional testing

For the client to fully cash in on the perks of automation, it was necessary to make sure that their manual processes run like clockwork.

Therefore, starting with the basics, our QA engineers mapped out a clear **workflow for various types of manual functional testing**, including unit, integration, interface, system, smoke, sanity, acceptance, and regression tests.

We've hit the ground running with the **Cucumber testing framework** as one of the leading options in the market, and during the first two months of cooperation, our QA specialists covered:



01 Extending QA capabilities by adding automated functional testing

With manual testing paving the way, automation was the next natural step.

We blended the Cucumber framework with the **WebDriverIO automation framework** as this combo is not only powerful enough for **large and technically sophisticated projects** but also **free of charge**.

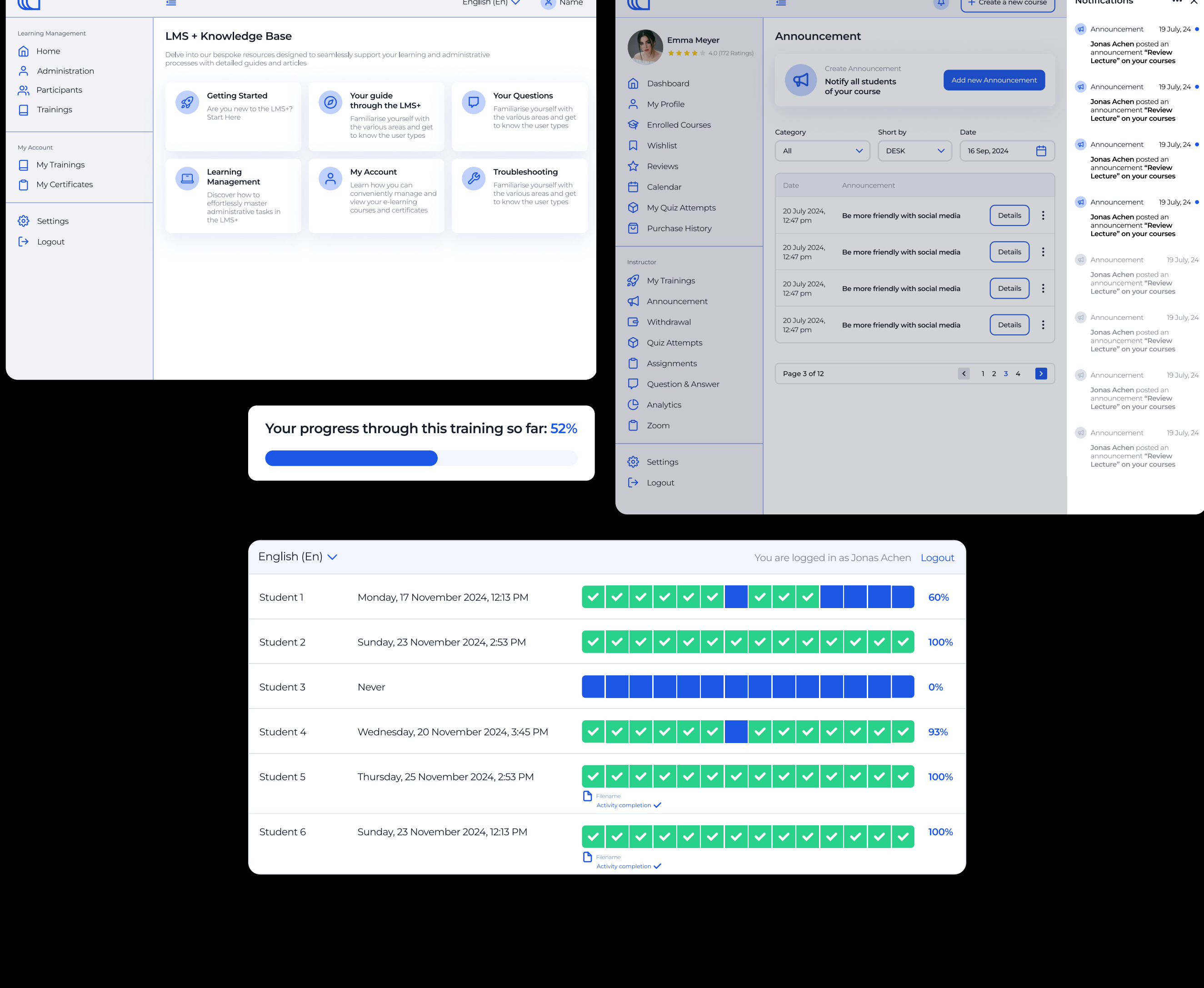
Thanks to our dedication during the manual testing phase, the transition to automation was **hitch-free**. Our automated functional testing now covers:

- LMS database
- APIs
- User interface
- Communication between the server and clients

We blew past the client's expectations — **automated tests now handle 85% of the LMS functionality**, 25% more than the initial 60% target.

01 Leveling up UX/UI

Our developers teamed up with the client's UX/UI designers to upgrade the platform's interface and functionality. The revamped LMS now boasts **intuitive notifications and progress bars** for users and admins.



We continue working on test automation, implementing non-functional tests, and supporting the client's in-house team in handling the rising scope of the platform's interface makeover tasks.

Before

- **Overcomplicated** system
- **Absence** of adequate software testing
- **Inability** to fix defects
- **Lack** of project documentation
- **Questionable** software quality
- **Skyrocketing** maintenance cost

After

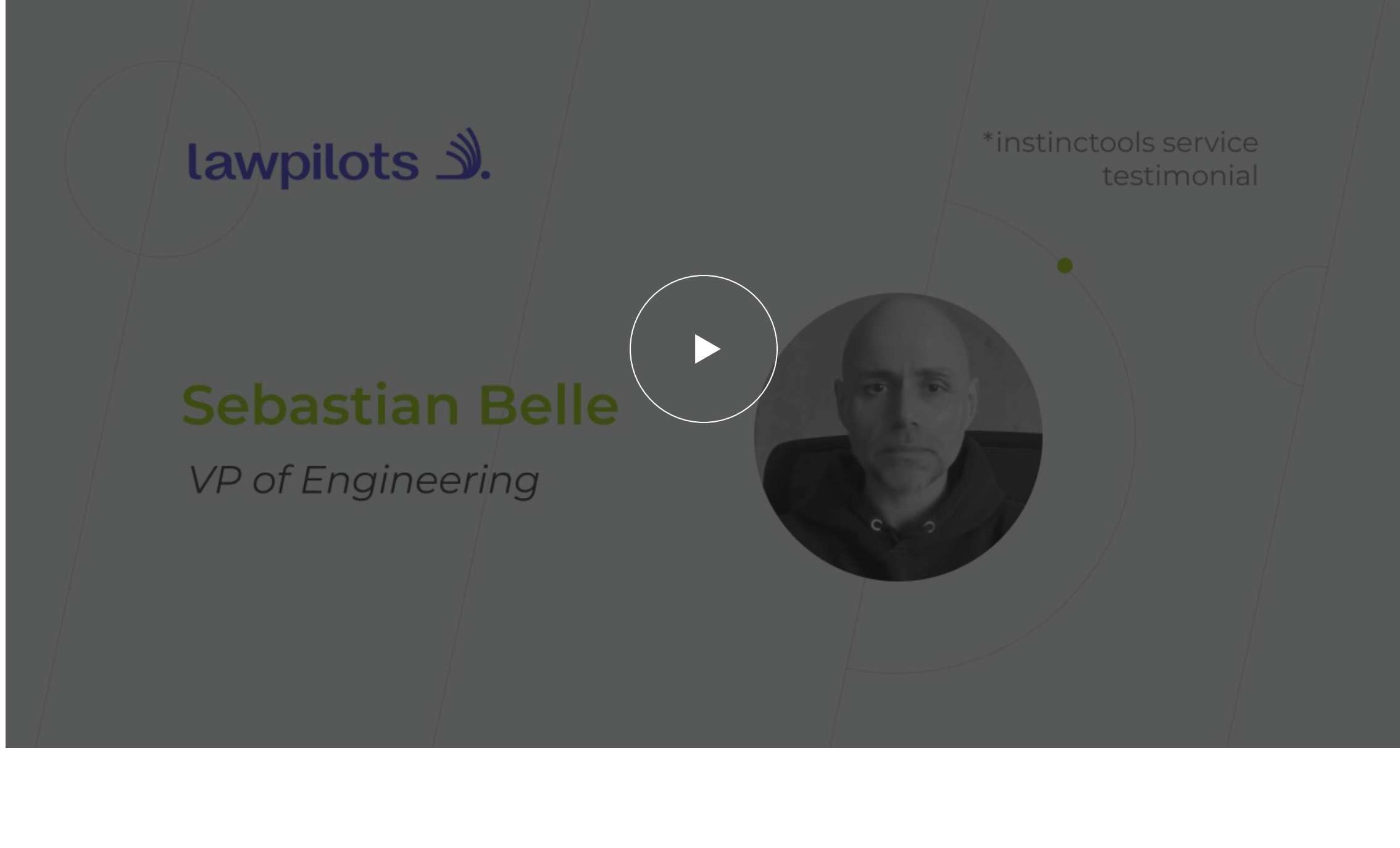
- **Well-thought-out and user-friendly** system
- **End-to-end** manual and automated functional testing
- **Clear** workflow for fixing any issues that arise
- **Documented** test cases
- **Enhanced** software quality
- **Decreased** maintenance cost

Business value

- Catching defects **x40** earlier in the SDLC before they play havoc on the bottom line
- Slashing bug repair costs **x160**
- Speeding up development cycles **x5** to get features to market quicker
- Bringing down LMS maintenance expenditures by **16%**
- **Minimizing** the possibility of costly human error

Client's testimonial

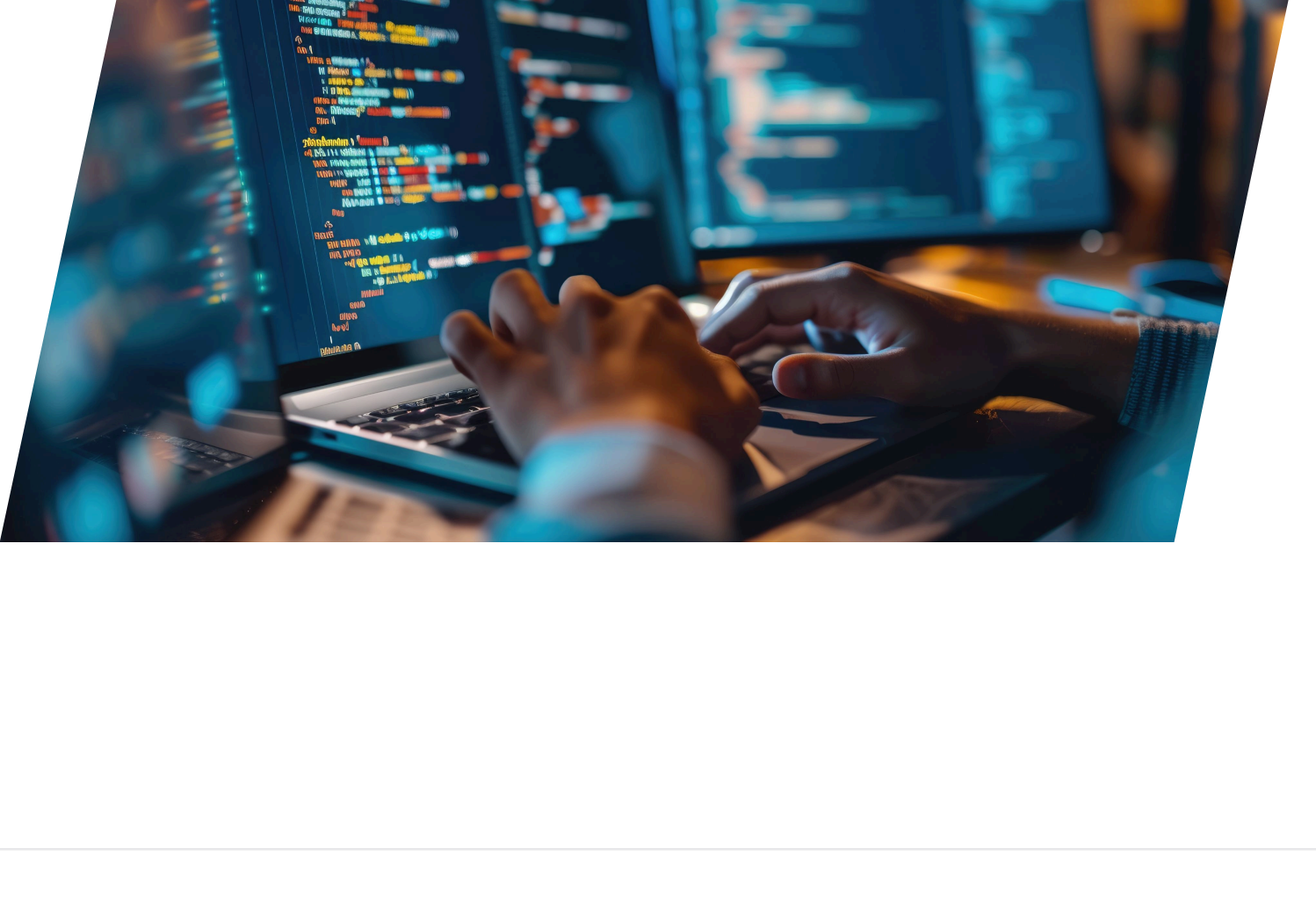
The client's VP of Engineering highlights cooperation with *instinctools:



Multiplier effect

Cutting corners on quality assurance is like setting yourself up for a sure failure. In 2024, **businesses can't afford bad software**.

It's **never too late** to bridge the gaps in your existing software testing with a comprehensive QA strategy and add powerful automation to your manual testing for maximum impact.



Do you have a similar project idea?