

# Delivering Quality the NashTech Way

Nash  
Tech.



[www.nashtechglobal.com](http://www.nashtechglobal.com)



# What's inside

Foreword	03
Introduction	05
The dangers of not investing in quality	05
The ripple effect of poor software quality	06
What NashTech quality means	07
How NashTech delivers high quality software	10
Our software quality metrics	13
Finding the right balance between quality, innovation, speed and cost	21
Real business value for our customers	22
Conclusion	23

# Foreword

**At NashTech, we have a passion for developing high quality software.**

We think that this is a key reason why our customers rate us so highly and retain our services, year after year. We also think this is why we have high employee retention levels and engagement.

**It seems true, that if you genuinely put quality at the heart of your engineering philosophy, you create a virtuous circle – which benefits customers, employees and your business.**

But, when you look into it, the notion of ‘quality’ can be difficult to pin down; how quality is consistently achieved is also a complex combination of factors. In this eBook, we aim to describe our approach to achieving quality in software engineering and how we measure this. We realise this is a deceptively complex subject, but we hope this is a small contribution to this important subject and illustrates our thought leadership and insight.

In essence, our quality philosophy is underpinned by:



### Talent

The attraction and retention of outstanding technical talent. As one of a leading employers of technology talent in our chosen countries we can select, and retain, the best.



### Culture

We focus on developing and creating a distinctive culture that values excellence, learning and collaboration. Our people are passionate about client value – we are respectful but willing to challenge and this balance is key to a healthy engineering practice.



### Focus

NashTech specialises in custom software development – specifically 'systems of differentiation and innovation'. We are very careful not to go outside our chosen specialisations in a busy market. We feel this allows us to create and enhance deep technical competency in our segment.



### Technical skills

It may go without saying, but we invest significantly in honing the range of technical skills and capabilities which are required to serve our customers. This is a continuous market scanning, learning, development and accreditation process.



### Measurement

We measure and continuously improve. This is done at several levels – overall quality accreditations we hold, the quality management system we deploy and at a detailed level within the software development lifecycle. Some of these elements are described more later.

It is interesting to consider that several of these factors may appear like 'soft' attributes, but they drive hard business benefits for our customers in terms of speed to market, productivity and cost management.

If some of the themes here resonate with you, or if you have different views on the subject, we would love to hear from you.



**John O'Brien,**  
Chief Executive Officer

# Introduction

The cost of poor-quality software in the US [\(CISQ\)](#)

has grown  
over  
**15%**

at least  
**\$2.41**  
trillion

technical debt  
around  
**\$1.52**  
trillion

Organisations need a way to buck this trend if they're to generate business value from their investment in software applications and systems.

Judging by our high customer satisfaction – shown in feedback, our high Net Promoter Score and long-lasting relationships – NashTech has found a way.

We put it down to quality.

Quality at NashTech has many dimensions, but they're not always easy to define or assess. **In this eBook, we've tried to articulate how we see quality, how it infuses everything we do and how we measure it.** Our metrics speak for themselves and we're happy to share them with you.

We believe that quality is the bedrock of successful software and underpins real business value for our customers.

# The dangers of not investing in quality

Poor quality software has many damaging and far-reaching impacts.

Projects suffer from cost and time overruns as developers struggle to find and fix defects in badly designed software. Too many projects don't complete at all. Applications that work at first can later prove difficult – or impossible – to scale or enhance.

Buggy apps damage customer confidence. In more serious cases they can **cause security breaches, with the consequent financial and reputation issues** these bring. Demoralised employees leave and it's hard to recruit good replacements.

When transformation projects don't deliver their anticipated benefits, organisations fight to survive, let alone flourish, in a competitive market.

# The ripple effect of poor software quality: from frustrated developers to lost business

Poor software quality isn't just a minor inconvenience, it's a financial sinkhole with far-reaching consequences. Let's delve deeper into the damaging effects that ripple outwards from a poorly built software foundation.



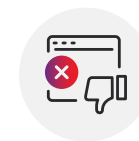
## Development hinders and project stalls

Initiating a project on an inadequate foundation can lead to a scenario of ongoing maintenance and potential structural deficiencies. Similarly, software with inherent design flaws can create a recurring cycle of defect remediation for development teams. This often results in cost and schedule overruns as developers address issues stemming from the initial design. In extreme cases, projects become mired in these challenges and fail to achieve completion.



## Scaling becomes a nightmare

When your application first launches, it may seem to run smoothly. However, as the user base expands and feature requests pour in, a hidden danger lurks beneath the surface. Many apps hit a scaling wall, where their initial architecture buckles under increased demand. This challenge often stems from limitations in the code. This forces businesses to face a daunting choice of either completely **rebuilding the software** or **find workarounds** that cripple performance. Both options can drain resources and hinder your company's momentum.



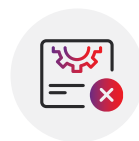
## Customer confidence takes a nosedive

Imagine using an app that crashes constantly or delivers unreliable results. Frustration sets in quickly, leading to **damaged customer confidence**. In today's competitive landscape, where user experience is paramount, buggy applications can drive customers straight to your competitor.



## Security breaches

Security vulnerabilities in poorly designed software create easy entry points for hackers. This can lead to **data breaches**, exposing sensitive information and resulting in significant financial losses and irreparable reputational damage.



## Demoralised workforce, hindered growth

Working with buggy software is a constant source of frustration for employees. It slows them down, hinders productivity and ultimately leads to **demoralisation**. This, in turn, **increases staff turnover**, making it difficult to retain top talent and recruit skilled replacements.



## The domino effect on business success

When software transformation projects fail to deliver on their promises, the impact on a business can be catastrophic. Businesses struggle to keep pace with competitors, impeding growth and even threatening survival in a cutthroat market.

# What NashTech quality means

## A magic mixture: people and culture

Many companies claim that their people are their greatest asset. At NashTech, the proof is in our vibrant, talented and motivated workforce. Competition for our jobs and our rigorous selection process mean we employ the very best.

And once people join NashTech, they don't want to leave – attrition rates lag well behind other tech companies. The results of our recent employee satisfaction survey are positive. **Our India and Vietnam development centres achieved scores of 68 (India) and 35 (Vietnam) on the employee net promoter scale, placing them within the top ranges of the “outstanding” and “very good / excellent” categories.** The company ethos is for excellence within a collaborative and supportive environment.

We've developed a 'people first' culture. Mature processes, leading-edge tools and training and good communication create the conditions for talent to thrive. And the whole is greater than the sum of the parts – brilliant individuals and the right environment mean NashTech can deliver the highest quality solutions to our customers.

“ I have been fortunate enough to work at, and have had leadership roles in, some of the biggest brands in the industry – Accenture, Capgemini, Hitachi, Ernst and Young, Infor and Gartner... and I have to say out of all those great companies, I like working for NashTech the most. A key reason, maybe THE key reason, is that our high delivery quality drives incredibly high customer satisfaction and advocacy – our clients seem to be as passionate about NashTech as we are about them.

**John O'Brien**, Chief Executive Officer  
NashTech

“ On a practical note, we can't deliver without NashTech. With the ability to scale quickly, and at a reasonable cost, I'm really impressed with the quality that NashTech has provided. It is really great, and I have fed that back on numerous occasions.

**Duncan Atkins**, Chief Information Officer  
Trinity College London

# Business focus and technical excellence

It's easy to get distracted in a busy digital marketplace – to feel tempted to move into new areas. But at NashTech we remain laser focused on our deep technical competency in custom software development, helping our customers develop systems of differentiation and innovation.

We provide our people with the best tools and training they need to perform their work and meet customer requirements. All engineers follow our training and testing programme to learn advanced technical skills, as well as business and people skills. Experienced engineers train and guide our newer recruits.

Our research and development (R&D) unit continually evaluates new technologies and trends so we can take advantage of any developments that could benefit our processes or meet new market needs. For example, we deploy automation frameworks where we can achieve efficiencies in development and testing. And our research into generative AI started well before the launch of ChatGPT!



*It's always good to work with an organisation like NashTech because you have an understanding of what the latest thinking is and the team members we have on the project from your side, in Vietnam, are absolutely excellent."*

**Frederic Vaeremans**

Chief Data and Solutions Officer, RPS

# Foundational quality – our Quality Management System

**Our Quality Management System (QMS) provides the framework and foundation for all our software development and delivery work.** It's the reason we can deliver such a high and consistent level of quality in all our projects.

The QMS is based on a cycle of continuous improvement, covering roles and responsibilities, processes and functions. Regular audits, feedback mechanisms and measurements help us monitor performance and keep improving.

We are certified in the key standards for our industry and have achieved level 3 of CMMI (Capability Maturity Model Integration). This demonstrates our proficiency in controlling our processes and our commitment to continual improvement.

With our **Agile methodology, quality is integral** to the development process. And the **flexible and collaborative** approach helps us work with our clients as one team, leading to more successful outcomes.

“

*Since the project started, I have also studied and learned about development languages and frameworks. The choice of NashTech was right. Not only does the team tell me what is best for me, I feel that constantly studying and following the latest technology trends is a great strength of NashTech. At the same time, I learned a lot.*

Mr. Hirayama, Co-founder, Japox



# How NashTech delivers high quality software

## Quality responsibilities

At NashTech, everyone is **responsible for quality**.  
But we also have teams charged with ensuring **quality compliance** and **improvement**.



### Quality Assurance

Our Quality Assurance team ensures that everyone is meeting their responsibilities within the Quality Management System. The team has a structured and comprehensive approach to providing assurance, running independent checks and audits and incorporating measurements and feedback into the continuous improvement cycle.



### Technical Team

Our Technical Team – informally known as the SWAT team – comprises some of our most experienced engineers. They swoop in to perform architectural and code reviews to ensure best practices are followed, provide guidance to development teams and work on continuous process improvement.



*“Looking at what we achieved, the professionalism and skill of the NashTech team is an example for me on how a supplier should work together with a client. Whether it was about functional requirements, development or testing, one thing that I always noticed is a positive, constructive way to resolve challenges. And then looking at the actual work, it is, and was always, on a very high level.”*

**Thomas van Rompaey**  
Programme Director, RPS

# Delivery lifecycle

We consider quality in all stages of our project delivery. Within the Agile methodology, this is focused on two-week sprints that deliver a set of user stories or features.

A sprint covers requirements analysis, coding, unit and functional tests, demo and release to the customer's DevOps environment.

We rely on our motivated and aware people who understand what's required for developing high quality software. They use a mix of tools and automation for efficiency combined with experience and skill to understand the customer's requirements and highlight any risks they see before development starts.

# Code quality

The quality of code at the unit level is a key component of overall quality – the foundation for delivering reliable and secure software applications.

Well-written code is less likely to contain defects and security vulnerabilities. It's easier to understand, debug and extend, reducing maintenance and enhancement costs. Developing to common standards helps developers collaborate and work together efficiently.

We use industry-leading tools such as SonarQube and Gitleaks to support code development. Our tools are integrated into our delivery processes and tool pipelines, ensuring quality code moves to the next stage in the delivery lifecycle.

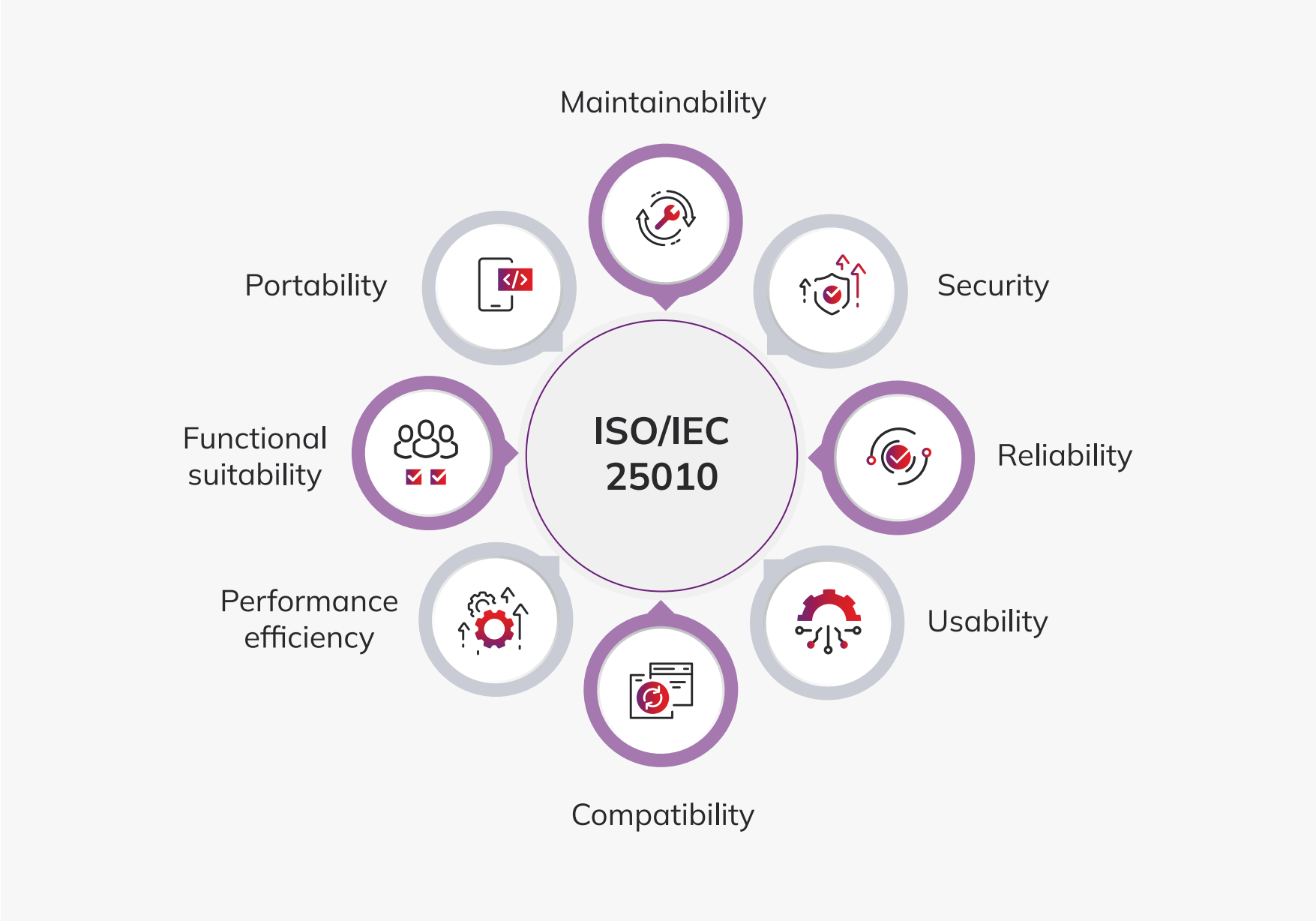


# Software applications and non-functional requirements

Non-functional requirements include **performance, reliability, security, scalability and usability**. Addressing these requirements early is critical to the long-term effectiveness and success of software applications.

But these factors can be hard to quantify, measure and compare against other products. And it's not always easy to understand the impact they can have over the lifetime of a software application – or the cost of trying to retrofit capabilities after the software is developed.

At NashTech, we've developed our own standard for non-functional requirements. Work in recent years has also led to adhering to the **ISO 25010** standard for evaluating product quality. We incorporate best practices from **ISO 25010** to provide a comprehensive framework for developing and evaluating software applications and we are assessing tools, such as the Sigrid Platform, to see if it is effective at measuring quality and delivering insights and recommendations for non-functional requirements.



# Our software quality metrics

We have calculated a set of delivery metrics using data collected for two-week sprints over three years for a diverse range of multiple projects.

In our results, we present the mean and median values for each metric. When we use numbers of defects in a metric, we weight the defects to reflect their significance.

# Delivery lifecycle

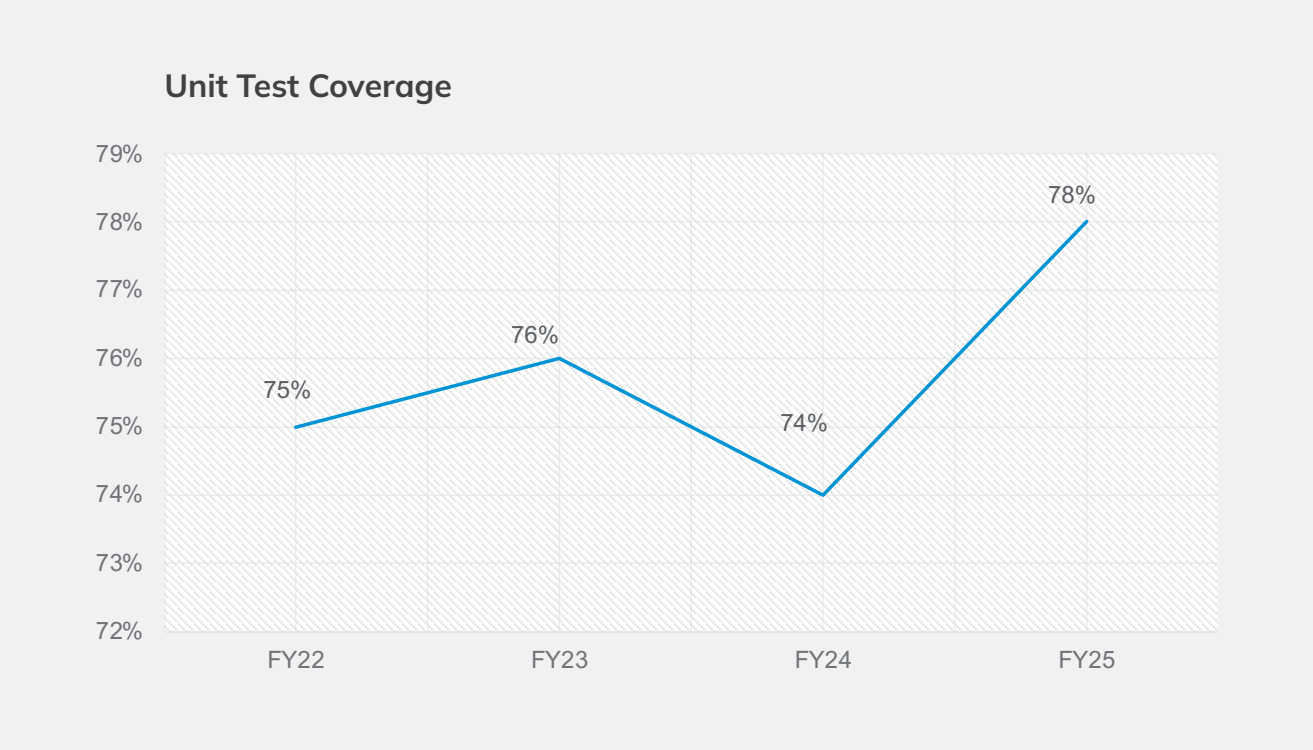
Unit test coverage measures the percentage of code executed during unit testing.

A high percentage doesn't guarantee that all defects are found, or that the tests are high quality. However, we have noticed a correlation between high unit test coverage and a lower number of defects found later.

We use different tools to measure unit test coverage, depending on the programming language and specific project requirements.

$$\text{Unit test coverage \%} = \frac{\text{Lines of code tested}}{\text{Lines of code}} \times 100$$

Industry best practice recommends **70 – 80%** test coverage to find the balance between the cost of testing and the likelihood of finding defects. **Our data shows that our unit test coverage is perfectly aligned with this goal.**



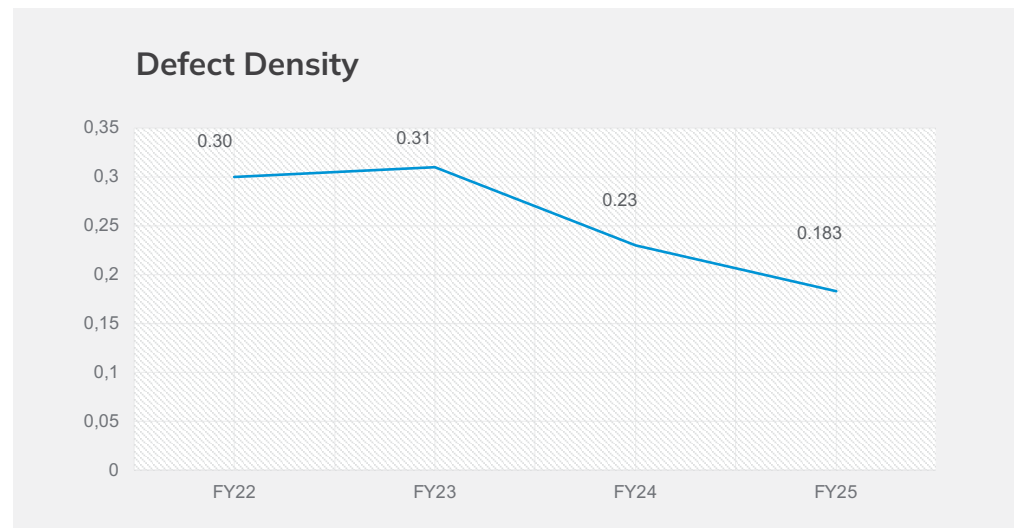
# Defect density – measuring the quality of our software

Defect density is the number of defects found in a software component relative to its size or complexity.

$$\text{Defect density} = \frac{\text{Weighted defects}}{\text{Development effort}}$$

The defect density metric is a reflection of the quality and stability of the software.

Our measurements show a trend towards zero. The small range of results demonstrates highly predictable quality in our software.



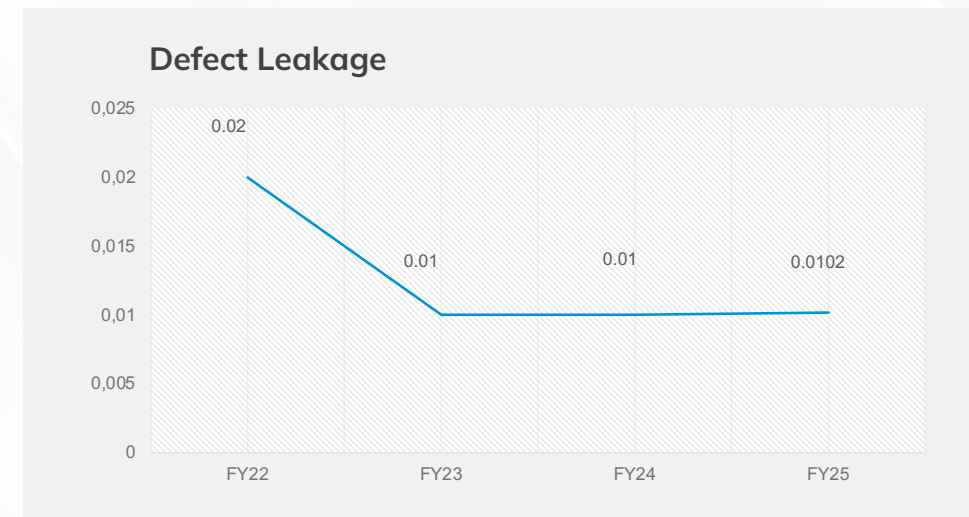
# Defect leakage – measuring the effectiveness of our testing strategies

Defect leakage measures the percentage of defects that were not detected during the development and testing phases and have escaped into the customer environment.

Measuring defect leakage shows the effectiveness of our QA testing procedures and helps us to identify improvements to our testing strategies.

$$\text{Defect leakage} = \frac{\text{Weighted defects (found after release to customer)}}{\text{Development effort}}$$

Our metrics show a consistently low defect leakage level, **which has remained below 0.05 from FY22 to FY25**. Our exceptionally low levels of defect leakage demonstrate the effectiveness of our code quality and testing procedures.

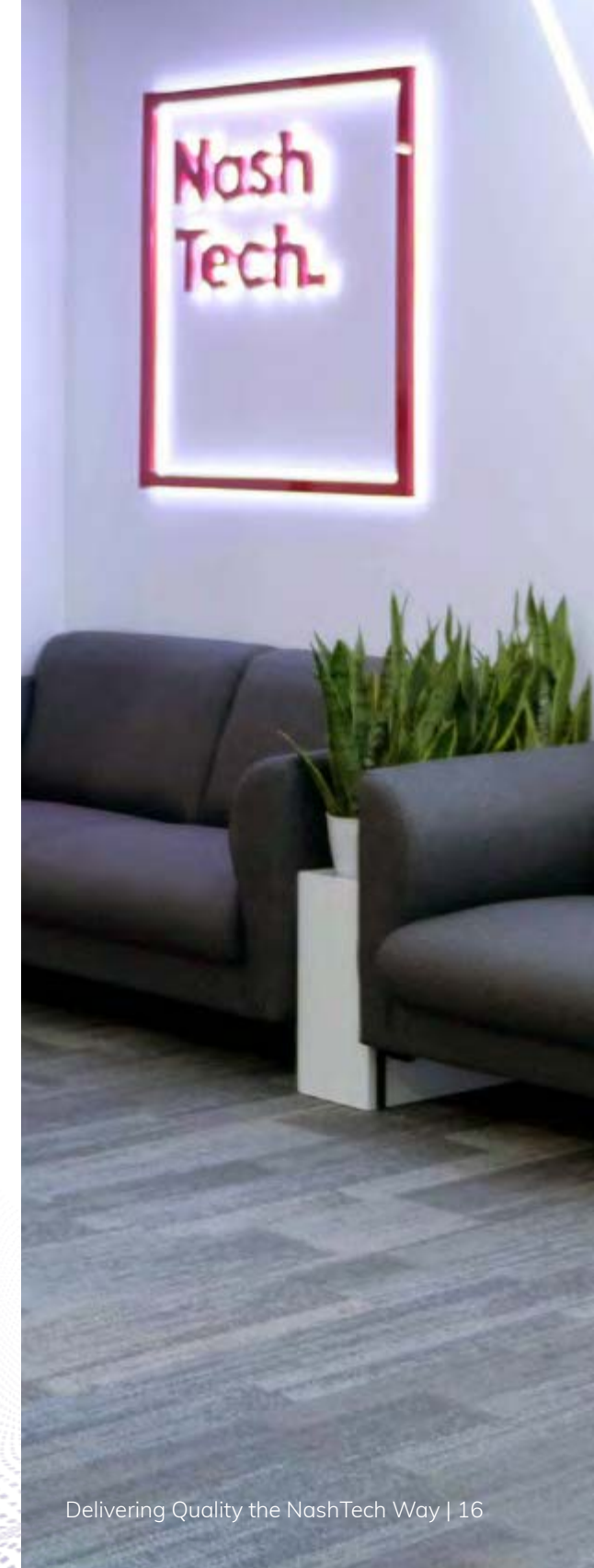
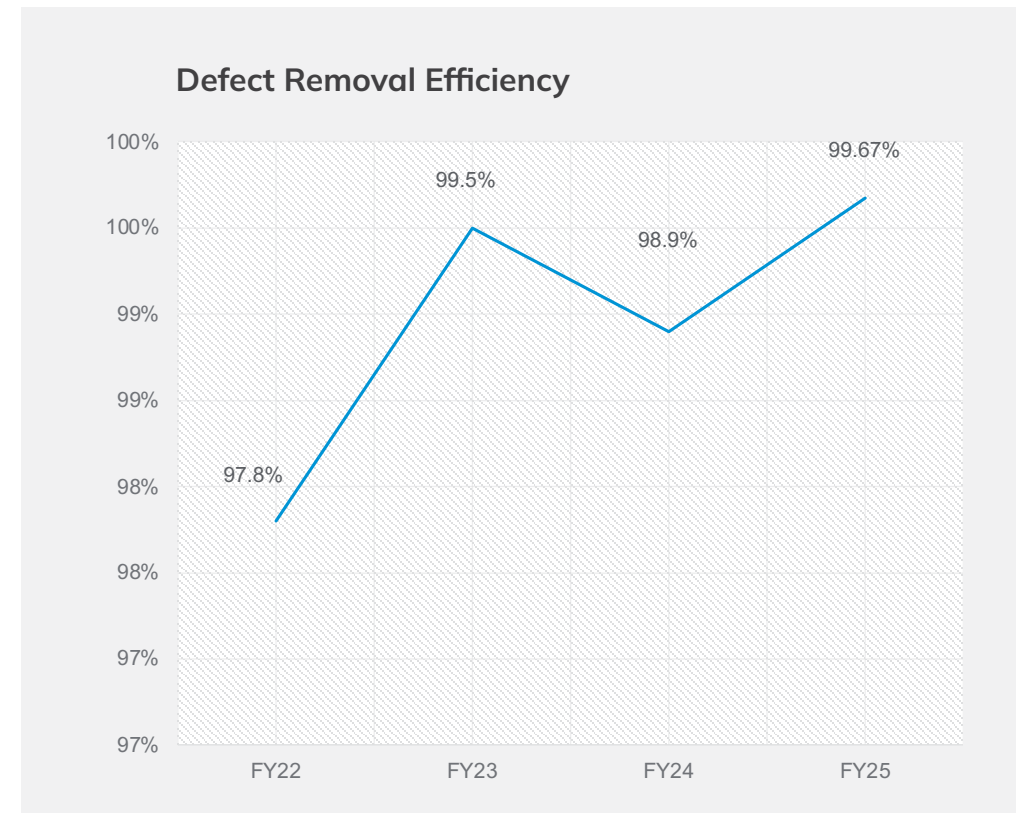


# Defect removal efficiency – measuring how well we find, and remove, defects

Defect removal efficiency (DRE) is a measure of the effectiveness of defect removal activities during a specific process or activity. We calculate the percentage of defects that have been successfully fixed compared to the total number of defects found during and after a two-week software development sprint.

$$\text{Defect removal efficiency \%} = \frac{\text{Weighted closed defects}}{\text{Weighted defects}} \times 100$$

A high DRE indicates an effective defect detection process and higher quality software. A DRE over 80% is considered good in the industry. **Our metrics show a mean value of over 98.9%. A value of over 80% is generally considered good in the industry.**



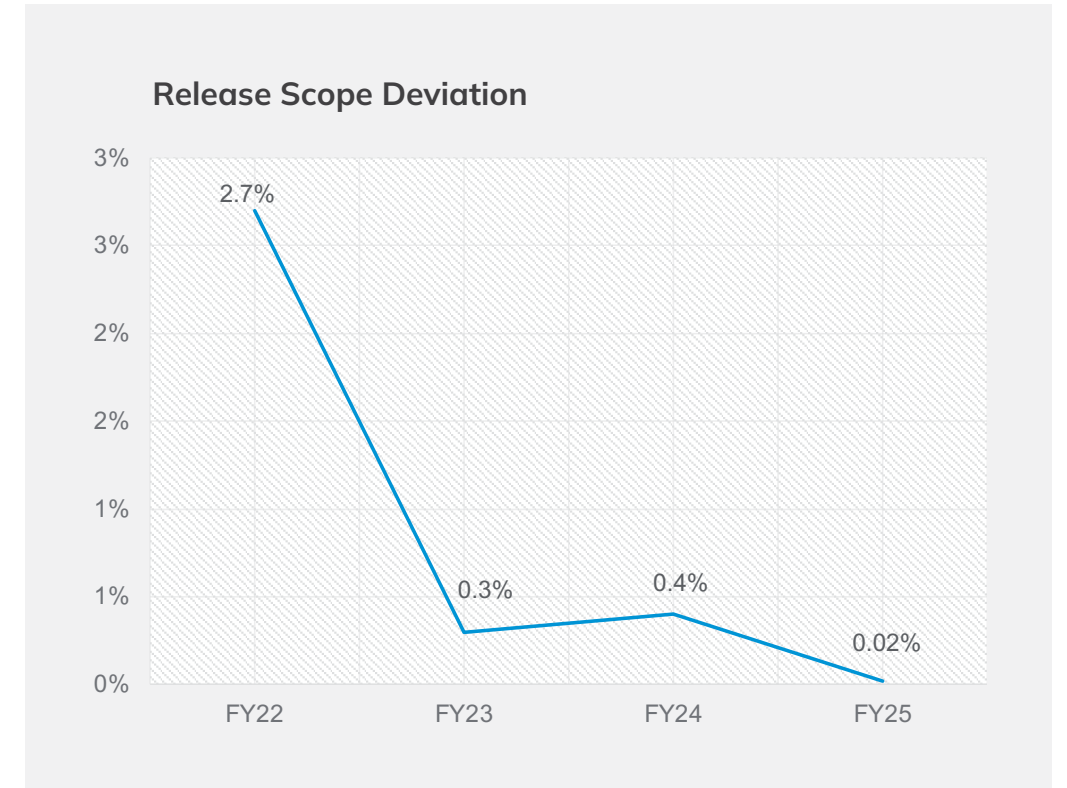


# Release scope deviation – measuring how well we deliver against our plans

Release scope deviation measures the difference between the planned scope of a release or sprint and the actual scope delivered. We measure the number of user stories that we planned to complete in a sprint against the number that were completed.

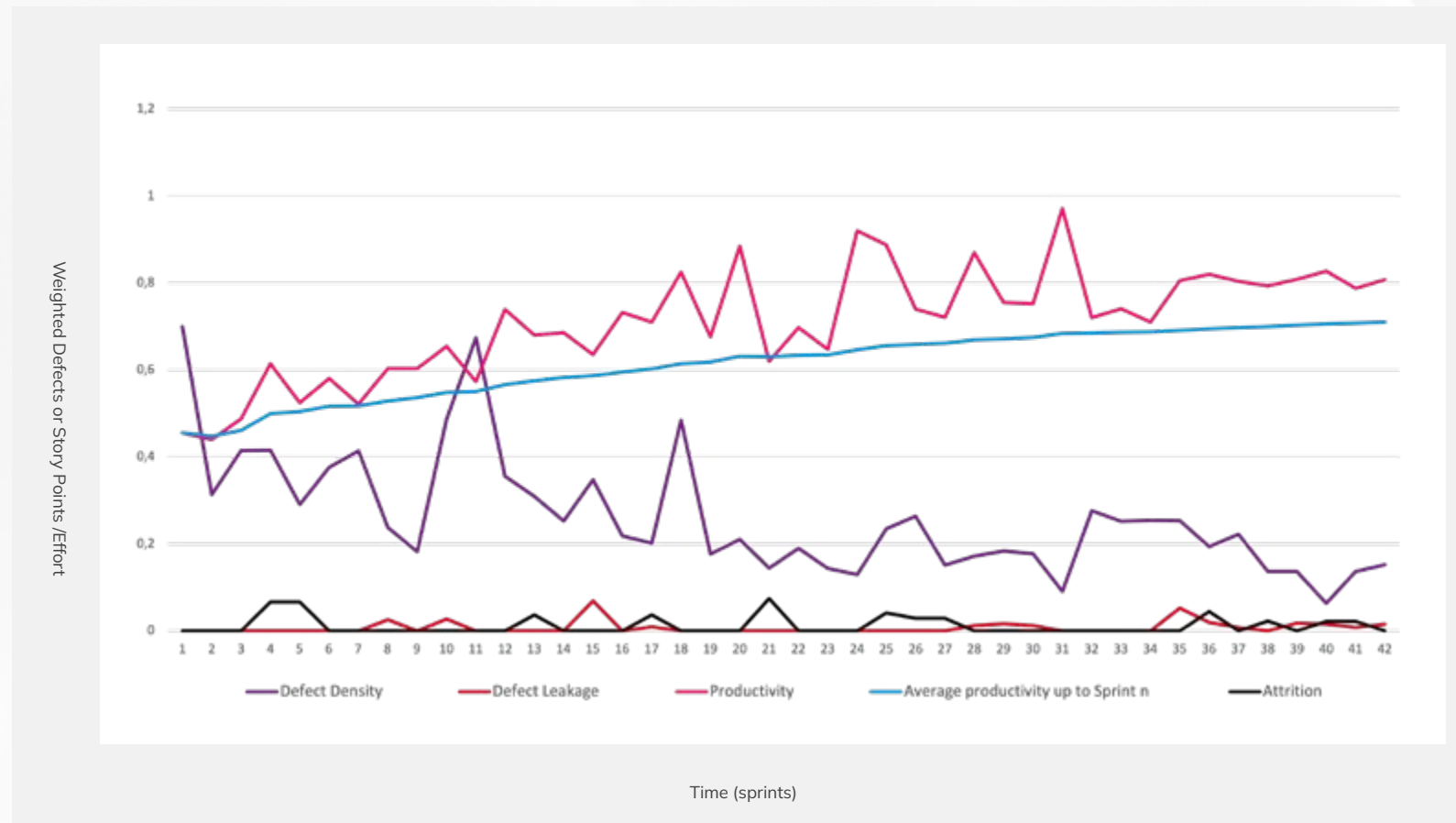
$$\text{Release scope deviation \%} = \frac{\text{Planned user stories}}{\text{Actual user stories}} \times 100$$

A high score demonstrates effective project management and the predictability and reliability of our software development process. Our metrics show that, on average, we deliver between 99% and 101% of planned work each sprint.



# Correlation among attrition rate, productivity and quality in a typical project

Demonstrating a typical SCRUM project where NashTech develops a product, owns the solution and has at least one release to the client. Data is collected from sprint #1 to sprint #15.



- Defect density** shows a trend of a reduction in defect density over time indicating continuous improvement in the team's understanding of the problem domain (learning curve), and the quality of the input requirements.
- Defects leakage** for this data set is very low, often 0 in each sprint. It does happen, yet a rare occurrence, and we perform a thorough Root Cause Analysis investigation when this happens to feedback into the process.
- Productivity** metrics shows that as quality improves, productivity improves.
- Average productivity** shows a steady increase over time.
- Attrition** shows no material impact on the quality and productivity metrics, meaning our processes handle the potential upheaval without impacting quality and productivity.

At NashTech, throwing more people at a project doesn't mean sacrificing quality. That's because our rigorous quality measures ensure we deliver results, even as we scale up resources.

# Findings

There is no correlation between attrition rate, defect density and defect leakage. This indicates effective management of team transitions and handovers, ensuring that productivity and quality remain unaffected when team members leave.

**There is a correlation between productivity and quality: productivity increases when fewer bugs are found.**



*For startups, speed is everything. Good for you if you can set up your own team of developers, engineers and architects. But if you can't, you want to have partner that can get you staff quickly, reliably and cost effectively. A partner where that team member has a home that makes them stay longer, learn outside the project and collaborate with colleagues. NashTech is such a partner. For the past ten years the NashTech team found the right people for my startups, startups I advise and also in my last corporate position as a global CTO.*

**Dr. Hagen Wenzek**

Co-Founder, Corporate Health International

# DevOps metrics

DevOps Research and Assessment (DORA) metrics focus on deployment frequency, lead time for changes, change failure rate and mean time to restore. The measurements provide direction for enhancing DevOps practices and feedback for the software development process to achieve higher quality and reliability.



**Deployment frequency** measures how often code changes are deployed to any given environment. Frequent deployments allow teams to collect feedback sooner and iterate faster, demonstrating the agility and efficiency of the software delivery process.



**Change failure rate** represents the percentage of deployments that result in failures in production, such as service outages or critical defects. A lower change failure rate indicates higher quality and more reliable deployments.



**Lead time for changes** measures the average time it takes for a code change to go from commit to deployment, reflecting the speed of the development process and the efficiency of the pipeline. Short lead times enable rapid iteration and quick delivery of new features and fixes.



**Time to restore service (recovery time)** measures the average time it takes to restore service or recover from a failure. A short recovery time indicates the effectiveness of the recovery processes, minimising downtime and the impact on users.

Some of our customers are already using DORA metrics to measure their DevOps performance. With the data, they're gaining a deeper understanding of the performance of their software development and delivery processes. They can measure their response to market and customer changes, and prioritise their investments.

**It is crucial for clear software quality metrics to be in place and measured before enhancing DevOps practices.**

**Don't wait to streamline your software delivery!**  
Get in touch today for a DevOps readiness assessment. We'll help you identify areas for improvement and get you started on the path to DevOps success.



# Finding the right balance between quality, innovation, speed and cost

Achieving good quality has a cost. While it's clear that poor quality software has an even higher cost – and not just financially – it's important to find the right balance between investing enough for good quality, and not striving for complete perfection.

**At NashTech, we have established robust and resilient processes that deliver high quality software – as our metrics show!**

We know, for example, there's a correlation between higher unit test coverage and lower defect density. However, we also believe that achieving a unit test coverage of 100% instead of 80% does not necessarily translate to better quality software.

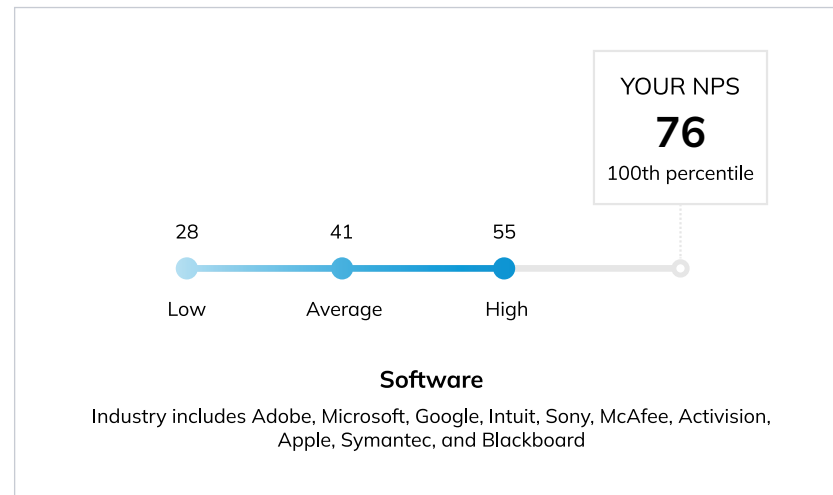
We have a flexible approach. We are happy to work with our customers to find the right balance for their priorities. For example, speed and innovation may be more important during a prototyping phase with more formal mechanisms when a product is developed for production.

“ We've had to deliver some really complex projects on really tight deadlines and they have not batted an eyelid. We wouldn't be where we are today ... without the team I have in Ho Chi Minh city.”

**Jonathan Chong**, Managing Director  
THE OUT

# Real business value for our customers

We believe in quantifying results where we can.



Our most recent customer satisfaction survey shows that we're achieving our goals – and helping our customers achieve theirs. **Our customer Net Promoter Score of 76 is in the 100th percentile in the software industry.**

In their feedback, our customers emphasised the quality, value and good service that we always strive to achieve.

Our quality translates into real business benefits for our customers. These include:

- 1 Successful delivery:** more than 80% of NashTech projects are delivered on schedule and within budget, a far higher rate than the industry norm.
- 2 Lower costs and fewer complaints:** the later that software defects are found, the more expensive they are to fix. After release, defects are not only harder to fix but can also damage user confidence. Our quality system helps us find defects early and, more importantly, prevents them from being introduced in the first place.
- 3 Peace of mind:** high quality software means our customers can release products and updates with confidence. By following best practices, we lower the risk of introducing technical debt that needs to be addressed later.
- 4 Great user experience:** poor quality applications that don't deliver a good user experience quickly lead to dissatisfaction and harm a brand's reputation. Our collaborative approach and user experience (UX) skills mean we deliver on the customer's user requirements.
- 5 Better planning:** our tracking and measurements give us reliable and predictable timescales for software development. Our customers can have confidence in their plans for release cycles, go-to-market strategies and IT budgeting.

“ It has been a real pleasure working with NashTech. They took the time to collaborate with the team and understand what we needed, fully documenting processes and creating a detailed solutions architecture. There is no way we could have scaled to where we are, in the accelerated timescale it's been achieved, without NashTech.

**Kenneth Haug**  
Enabling Platforms Team Lead  
System Owner  
Wellcome Sanger Institute

# Conclusion

## Quality: the heart of your engineering philosophy, not an afterthought

Software quality isn't an add on at the end of development. It's the very foundation upon which successful software is built. A quality-centric approach is paramount to maximising the business value of your software investments. It is not a collection of disparate features; it encompasses everything from the initial conceptualisation to the ongoing maintenance of the final product.

This includes the following aspects:



### Robust design

A well-designed architecture ensures the software can adapt and grow as your needs evolve



### Flawless code

Clean, well-documented code minimises bugs and facilitates future maintenance



### Impeccable security

Rigorous security measures safeguard your data and protect against vulnerabilities



### Intuitive user experience

User-friendly interfaces and workflows promote adoption and enhance user satisfaction



### Seamless performance

Efficient code and optimised infrastructure ensure smooth operation under load

By focusing on these elements throughout the development lifecycle, you create a holistic quality that translates into tangible business benefits.

# Quality is a continuous journey, not a destination.

While achieving high-quality software is our constant pursuit at NashTech, we understand it's an ongoing process. We believe in the power of continuous improvement. This means:



## Embracing agile methodologies

We leverage agile practices to adapt to changing requirements and ensure quality remains a constant priority



## Investing in innovation

We actively research and implement new tools and techniques to stay ahead of the curve in the ever-evolving software landscape



## Fostering a culture of quality

We empower our team members to be quality champions, taking ownership and responsibility for delivering exceptional work

This relentless dedication to quality improvement allows us to deliver software that not only meets your needs today, but also scales and adapts to your future demands.



# The conversation continues: let's talk quality!

We are passionate about software quality and its transformative power. Whether you're interested in learning how our quality-centric approach can benefit your business or simply want to discuss best practices and industry trends, we'd love to hear from you.

Let's put quality at the heart of your engineering philosophy that turns your software vision into a reality, benefitting your customers, employees and your business.

Ready to talk about quality?



We are experts in technology, delivering smart solutions that solve business challenges and create value. Our award-winning teams apply deep expertise and passion to deliver complex IT projects globally.

For more details, please send your enquiry to [info@nashtechglobal.com](mailto:info@nashtechglobal.com) or visit our website [www.nashtechglobal.com](http://www.nashtechglobal.com)

**Nash  
Tech.**