The New CTO's Handbook

Start your new role as CTO
the right way with practical advice
from senior tech executives.
Learn how to prepare and what to expect.

STX NEXT
python powerhouse

Table of Contents

Introduction

Career paths vary. No two journeys are exactly the same, even if you're heading for the same destination. And, as with life, there are bound to be twists and turns you may not have anticipated on the way.

But we can learn from each other—support and help each other by sharing information.

We hope the insights in this ebook will help you as you embark on your journey as CTO.

Who is this ebook for?

The New CTO's Handbook was written as an expert's guide for newly appointed Chief Technology Officers. This may mean that you have just accepted a CTO role for the very first time. Or perhaps you're an experienced CTO joining a new organization. You could even be looking ahead to the future and planning out your career.

Whatever the situation, taking on a new role as CTO—or any other technical executive role, for that matter—is a very exciting time. A time when you can influence positive change for your colleagues and your organization.

What will you get out of reading this ebook?

This handbook was put together to make sure your transition to a new CTO role will be as smooth and efficient as possible.

You will find contributions from seasoned CTOs and other tech executives here. From personal experience, they have kindly shared their observations and real-life advice to help you plan your next steps, overcome challenges, or avoid potential issues completely.

Good luck!

Getting to Know Your Company

Ask experienced CTOs what advice they'd offer someone new to the position and one word comes up time and again: learn.

Learn. Learn. Learn.

Learn about your company. Learn about your colleagues. Learn about your company's culture. Learn about your stakeholders. Learn about your competitors.

Learn about everything you can.

Like **Hugh Cumming, CTO at Vena,** says: *"The most important thing is to get to form your own 360 on people, product, and customers by sitting down (virtually) with as many people as possible. Focus on getting to know your leadership team and how they work together, but also reach beyond that and meet with the informal leaders in the organization to understand how work gets done and who can help create anchor points to the strategic narrative you want to build."*

Aaron Podolny, Co-Founder and CTO at Scribe, agrees: *"A new CTO needs to start as a sponge: soaking up everything from technology stack to team strengths and weaknesses."*

What to focus on

Information is necessary to plan your strategy, though you don't want to get overloaded with facts and figures. Concentrate on the essentials at the start and optimize your time to use it most effectively.

Matthew Shiner, CTO at Meta Technologies, has highlighted four areas where new CTOs should direct their attention:

- **Focus on learning the new company, industry** (if applicable), and day-to-day **responsibilities** specific to the CTO at that organization.
- **Evaluate how the company is performing** in relation to their KPIs, and how to use your current technology infrastructure or new options to increase performance.
- **Identify any knowledge and/or skill gaps** within your internal IT.
- **Develop a vision and strategy** for the company based on your findings and communicate that to the executive team.

By considering these four areas, you'll build up a complete picture of how everything fits together, currently and for the future.

As you can see, at the top of Matthew's list is focusing on learning the new company. Finding out about the business and the wider context in which it operates is vital. Even if you are being promoted to CTO within your existing organization, you can still educate yourself on the finer points.

So what's the best way you can get to know the company better?

Patrick Kelley, Founder and CEO at Critical Path Security, has laid out a six-point plan on how new CTOs can understand their organization—and how you can make your transition into the position smoother.

1) Get to know your predecessor

If possible, a new CTO or CISO (Chief Information Security Officer) should interview their predecessor. Most CTO-led programs are tied to long-term goals with relatively long implementation cycles.

The predecessor is likely to provide some unique insight that is often left out of the documentation. Try to have them introduce you to the other department leaders, executives, and your direct reports. Their stamp of approval will make your transition much easier.

2) Learn the culture

This requires asking a lot of questions and listening intently. Patrick recommends mapping to the McKinsey 7S Framework, if possible. It will help you find your fit and learn how to best communicate across the organization.

3) Perform a high-level assessment of the current IT capabilities and commitments

There won't be enough time to get deep into the weeds, but that's not really your focus. You need to get an understanding of your assets, how things are managed, and understand the cycles of deployment and maintenance. You might find some helpful tie-ins to ITIL or similar.

4) Look for quick wins for your first 90 days

This will create positive synergy in the team, as well as get momentum going.

5) Build your initiatives for the first year

At the same time, be mindful of requesting too much money or resources. Most organizations have an undefined appetite for change, but when you push for too much change, you will know right away when you've exceeded that appetite.

6) Review the existing agreements

Be sure to review all of your third-party vendor agreements, along with the MSAs (master service agreements) and SLAs (service-level agreements). In the post-pandemic world, much of what you're responsible for won't live inside your ecosystem. You need to know what expectations have been set with your vendors. Their breach is your breach.

Don't be afraid to ask questions

You won't find out everything you want to know by watching and listening alone. Open two-way communication is indispensable, even if it leads to difficult conversations.

Equally important to asking questions is listening to the answers you're given. Allow yourself to be open to new ideas and guided by what your colleagues are telling you.

Karim Zanaty, Head of Software Development at Coboworx, explains the importance of asking questions and adapting accordingly to what is happening around you: *"What I thought about leading people in tech areas is that you should take courses, learn all theories about management, and be guided in a long process before you are even able to do such a job.*

But it is not like that. Of course, I needed time to know how to operate with customers and partners. But I am not afraid to ask if I do not know something. I reach out to people, friends,

former bosses, or, importantly, current cooperation partners! I tell them honestly what I am missing and that I am open to learn. I discuss plans for management with them.

I am adapting from the best in my surroundings to achieve the goals in our company. I am tired of 'easy steps to become the best next lead' or 'courses and certificates that you should not miss.' They help you, yes. Give compact information, yes. But they do not prepare you when it comes to practical things.

If a guide should do something then it is to point out that the most important thing is not to focus on yourself to become better but to focus on your environment and be open to your surroundings' opinion and skills to learn."

Questions to ask yourself

To gather all this priceless information about your organization effectively, you need to be asking yourself questions. Having a clear idea of what you want to find out will help you enormously.

Eric Billingsley, CEO at Guide-Rails, has shared some tips on what to consider, evaluate, and assess as a new CTO:

- Understand the mapping of resources to priorities. For example, is money being spent well? Are people working on what matters? If not, why?
- Drive focus in the organization, initiate a reorganization or reallocation. Shake up the status quo.
- Dive deep into quality. How is testing being performed? Whose job is quality—just the QA team's or everyone's? What's the level of automation? How often do issues affect customers?
- Set metrics around bug fix priority, code coverage (80%), and automation (100%) goals.
- Understand cycle times. How quickly can you deliver for your customers and your business? How frequently do you push changes? Can you push a small change in less than a day?
- Simplify the process, invest in VSDPs (value stream delivery platforms), set goals around delivery time (>50% of changes, <1 week).

Think outside your organization

It's important not to become insular in your approach. Look at the bigger picture. You have to understand the sector your organization works in and take into account external factors.

Sergey Velts, Founder and CTO at Cybertonica, shares this advice: *"What I would advise new CTOs is to always think about stakeholders, the market, and the competition. At the end of the day, it's their job to win against competitors using given time and resources, be aware of risks, including risks of over-engineering, under-engineering, and managing people."*

Look forward, not backward

Your experience has taken you where you are, so it's only natural for you to want to draw on what you've learned from previous roles. But if you're joining a company, be careful not to compare your new employer with your previous job too much.

Try to look at the advantages of your company's current ways of working. And maybe question why those solutions were implemented in the first place—there might be factors that you hadn't considered.

As **Hugh Cumming** puts it: *"Avoid overusing references to past experiences to highlight better practices. We've all said, 'We used to do this at…' And remember people may have thoughtfully put in certain processes without the same experiences. It's important to create a metaphor for the changes you want to see without trying to inject another company's practices directly.*

Focus forward and don't get overly distracted by history—you weren't there when any of the past decisions were made, so best not to try to adjudicate them differently. That can disengage folks that you will need on your journey."

This is not to say that you shouldn't be offering up fresh perspectives. You've been chosen to help drive the company forward in the best way you see fit. Of course you should use your knowledge and experience to create a successful strategy. But focus on the present and the future, rather than the past.

Certainly processes can be improved, tech stacks can be altered, and new ways of working can be implemented—but in time. Every company should be trying to continually improve its working methods and streamline its processes. But you need to understand the bigger picture before you start to execute changes.

Kathleen Moriarty, CTO at the Center for Internet Security, explains: *"In the first few months of being a CTO, listening, reading, and absorbing internal strengths, gaps, as well as relevant industry trends is most important.*

While you may enter an organization with wonderful new ideas, it is important to fully understand the environment and challenges to adjust your approach to ensure success for CTO initiatives and for the organization's technology strategy."

Rushing decisions or acting too soon may well cause bigger problems in the long run. There's a benefit in taking things slowly, which is what we'll touch upon in the next chapter.

Implementing Changes

Once you take your position as CTO, you'll probably be eager to start implementing changes as soon as possible. With a head full of ideas, you may be itching to just get out there and hit the ground running. But is this the best approach?

Start small and gradual

CTOs we have spoken to have warned against making big changes at the beginning of your tenure. **Hugh Cumming** explains: *"Don't overthink it and try to change the world all at once. Make a series of small changes and learn and adjust and work towards the larger changes as the organization gets to know you."*

Rohit Sinha, CTO at SmartPM Technologies, Inc., also promoted a similar idea of a series of small changes in the form of iteration and pivoting: *"What's important for new CTOs, especially early on, is the ability to iterate quickly and pivot. You won't have the right product–market fit from the beginning, and that's okay—it doesn't have to be perfect. But you do need to be flexible so things can be adjusted as you progress and continue to build out.*

It's impossible to anticipate all your clients' needs, and you'll only figure that out once clients are using the product and provide feedback. Ensure that whatever you're building has the capability to pivot fast. If it's too hard to make quick, necessary changes without months of rework, you may have already lost that client.

Also, you'll want to do an incremental delivery where you have small features that make use of larger features along the way, so look at the big picture and plan accordingly."

You need to have a solid understanding of your organization and the industry before making changes that might not be the best for your business.

Gregory Yong, Chief Experience Officer at Convincely, explains: *"Even if the workplace does need a serious overhaul, you need to make sure that your changes are well understood and properly executed by the team. Consider crafting an operations handbook outlining the provisional changes you plan to make within the organization. This helps to standardize communications and ensure that all stakeholders are on the same page.*

After all, the number-one thing you must avoid is a communication breakdown. It can leave employees feeling deflated, disengaged with their work, and potentially resistant to your leadership."

Consider the stakeholders

As **Rohit Sinha** says: *"As a new CTO, it's essential to understand what stage the product is at, and the short-term and long-term vision for the company from the perspective of all stakeholders. All of them have to be on the same page."*

Involve others in the changes you're making

According to **Gregory Yong,** gradual changes should be made with the involvement of your team: *"In your first 3–6 months as a new CTO, make changes gradually. There's a high likelihood that you have a vision for change and a plan in which to make that change happen.*

However, you must make sure to avoid the trap of overproductivity—that is, trying to make operations perfect all at once. Rapid changes can lead to extended lulls in production as employees work to reacclimate themselves with new software and technology. By doing this all at once, you run the risk of leaving the business at a complete standstill while employees get up to speed.

Make sure you create improvements with the guidance of your team—they could offer useful perspectives owing to their extensive experience at the company. With these considerations in mind, your first few months as a new CTO will go smoothly, and should promptly improve day-to-day operations."

Prioritizing & Delegating

Involving others in what you're doing doesn't end at the changes you'll be implementing. Actually, that's only the beginning. What's equally important is letting them help you.

Particularly at the start of your new professional journey as CTO, there's a very real possibility that you're going to be overloaded with all the responsibilities and duties.

During our interview on Tech Leaders Hub, **Aviv Ben-Yosef, Tech Executive Consultant and author of** *The Tech Executive Operating System,* said: *"At the beginning of your work as a CTO, it will feel like every random chat you have unearths more issues that you should be dealing with. Do not become overwhelmed. Instead, diligently keep track of what you're hearing and pass it along to the next steps in the process."*

To deal with the overwhelm, prioritization is essential. You need to set realistic goals and understand which issues should be addressed first.

The shortest path to success

You can't do everything on your own—and no one would expect you to. Using other people's skills, knowledge, and experience is vital if you're to succeed as CTO.

Aaron Podolny explains how effective prioritization combined with collaboration is key to success: *"While it's tempting to focus on solving engineering challenges at first, figuring out how the technology organization contributes to the company bottom line is likely the shortest path to success.*

There's often a web of informal stakeholders in product, design, sales, and success that all care deeply about how engineering allocates resources and bringing them in as early as possible will build trust and enable faster and higher impact innovation."

Don't be a hero—learn to delegate

As well as organizing work and collaborating, you must also learn to delegate. Which may not be as easy as it sounds.

"Many tech leaders come from technical positions. Our responsibility now is no longer to write code, however tempting that might be, but to shape visions and coordinate teams."

—Hendrik Wallbaum,
Head of Frontend at Taxdoo

Dor Zilka, Head of Digital Development at Cal (Israel Credit Cards), knows from personal experience that transitioning from a technical position to a leadership position is something that you need to work on: *"As a tech leader, I struggled with the fact that I had less keyboard time. It took me a while to learn to balance my time and stay up to date with technology without actually taking on a lot of development assignments."*

Even though it may be challenging, delegation is crucial for you and your team.

Eric Billingsley has shared his advice on working efficiently that focuses on the advantages of delegating work: *"Stop doing as much as you can—create space to do what you need to do next.*

Agility is the cure for most issues in a technology organization. If you have fast cycle times, you can create or fix anything that comes your way.

Shift left to find and solve problems as quickly as possible to drive faster cycle times. Issues found later in the process take more time and resources to resolve than those caught early.

Building your own tools is a huge trap—your team will end up automating existing broken processes. Buy a product instead.

Don't let compliance and security become reasons to go slow. Make them part of your process and drive them as early as possible."

Widen your scope

As a CTO, your responsibilities grow and your focus widens. You need to make sure your strategy is not too technology-centric. Technology is a main concern, true enough, but you must allow other factors to be taken into consideration.

As **Matthew Shiner** explains: *"You should avoid the 'technology only' trap. Focus on driving conversations to better enable decision-makers to make informed, data-driven decisions.*

CTOs do more than just managing the technology, they play a critical role in following the organization's strategic path. It's important to communicate this to the wider organization.

Similarly, don't do all the technical work yourself. You need to have time to focus on bridging skills and knowledge gaps by hiring qualified staff and leadership."

Remember your role as a leader

As **Hendrik Wallbaum** says: *"For leaders, the task is to facilitate decisions rather than make them. As a result, looking into how decisions can be made and consciously giving concrete decision-making processes a frame is super helpful.*

When deciding on a component library, I communicated that this would be a process where we don't look for consensus but consent in the sense that not everyone has to agree but strong opposition would halt the process.

Making this clear up front was very helpful in reducing friction related to the process and kept discussions focused on the topic at hand."

Managing Your Team

Giving support

As a leader, your role is to provide your team with the support they need to do their jobs.

Stuart Kelly, Co-Founder and Principal Engineer at Zego, recommends five levels of support that your team should receive. Bear in mind that this support can be provided by multiple people—the responsibility doesn't need to fall solely on you.

1) Visionary support

It's your responsibility as a manager or leader to give your team members a clear picture of the organization's future goals. You need to remind them of the company's long-term strategic vision and how they can contribute toward realizing that vision. This way, they'll all be working to further the company's long-term goals.

2) Product support

If you're taking up a leadership position in a tech company or product development department, you'll need to support the product team.

You can do this by helping them identify the problems that the product they're developing is intended to solve, the end users of the product, and the necessary features. This will help them prioritize and deliver suitable products that match specific needs.

3) Operational support

Every tech team needs core operational support to perform optimally. The operational support required may include support in activities such as looking out for backlogs,

making sure tickets are up-to-date, and ensuring code is reviewed fast enough to get features released on time. This is typically [the role of a Scrum Master](#) or Delivery Master.

4) Career support

As employees remain committed to meeting the company's goals, it's important to give them some form of career support to help them grow in their profession.

According to **Stuart Kelly,** *"Everyone with a career is often looking out for their career development."* Therefore, it's best to ensure that employees leave your organization better than they were when they first joined. This is usually the role of an Engineering Manager in tech companies.

5) Engineering support

Developers typically know what needs to get done, why it needs to get done, and the order of priority in which to do it. However, they may not know the optimal way to go about it and need some technical leadership to figure things out.

This type of support is usually offered by Senior Staff Engineers—good leaders who understand the product's technical requirements and the best way to get things done.

Create alignment

"My number-one tip is to bring your team into alignment. What does that look like? You can imagine a team that is not aligned as two people holding a tug-of-war rope. The two people on either end are pulling it against each other, causing stress and getting nowhere.

An aligned team is two people holding a rope on either end and walking in the same direction. Sure, they will occasionally get hit by an obstacle—they might get wrapped around a tree or they might want to go off in a slightly different direction—but they're always going for the same goal. That's what an aligned team is about. It's happier, and it produces more code."

—Adam Craven,
CTO and Founder at principles.dev

Focus on building relationships

Good management is built around strong relationships across your business as well as with people external to your organization.

Kathleen Moriarty says: *"Relationship building is essential not only with peers but also other influencers in the organization who may be at varying levels of management or individual tracks."*

Furthermore, good relationships are built on a foundation of trust, openness, and honesty. It is also important to be yourself.

"Make sure people get to know you as a person, especially the way you think and make decisions and how you lead others," says **Hugh Cumming.** *"You won't have all the answers and will need the team to be super engaged to help you be successful. Great companies and products are built bottom-up and not top-down. Cover off the basics around security and risk early to avoid any surprises later on."*

Present yourself as you want to be seen

"For me, it's about making sure that as a tech leader you build your own personal brand and you're putting yourself out there on various social platforms so that people get to know you and understand what you're doing.

It's really good personally—you get invited to talks and webinars, for example. It's also good for the company you work for as it's bringing attention to it.

The other thing you attract thanks to your personal brand is talent. Recruitment is one of the hardest things these days, especially in tech. Getting a personal brand and being out there is one way that people actually come to find you."

—**Andrew Beacock,**
Head of Engineering at Canopy

Take a step back

Instead of making quick decisions, it's usually best to take a step back. This will let you reflect and take into account everything you need to act accordingly.

Hendrik Wallbaum has a technique that helps him as a leader: *"A concept that helped me a lot when my authority increased was to 'give it five.' Most things people come running with don't require immediate action and are not as bad as they seem after you have given them five minutes to reflect on what is actually happening. This applies to many aspects of our work, including ideas you disagree with at first.*

As a leader, your own opinion is often not what's important. The important thing is to unleash the group, and that might require you to change your perspective. 'Give it five' is a great trick that works here as well. If you want to dismiss an idea outright, give it five and challenge yourself to find something positive in it."

By giving yourself some space, you're allowing yourself to be active rather than reactive.

"We have to stop and take a breath. And don't rush and react to whatever's happening," says **Aviv Ben-Yosef.** *"I want to see leaders in technology actually leading and not being led by what their calendar's telling them to do, by requests coming in, I want to see you start by taking a breath, thinking about what's coming in, and then deciding how you want to react, whether you have to react right now, what is the right way to approach it, etc. So relax."*

Managing during downtime

It's likely that at one point or another, you're going to experience downtime. This situation is far from ideal and things can quickly spiral out of control if not properly managed.

[During our conversation on Tech Leaders Hub](#), **Junior Godoi, Software Engineering Manager at Scurri,** shared with us three ways in which you can manage downtime effectively:

1) Engage everyone

Get all hands on deck in times of crisis. This way, you can quickly tap into the expertise of different people and solve the problem. Moreover, prioritizing the challenge will ensure that everyone gets back to their tasks quickly.

2) Define roles

Once you've gotten everyone involved, you can assign particular roles to speed up the process. Everyone else can watch and learn or go back to their tasks.

3) Communicate

Constant communication will help your team achieve quick recovery. Therefore, you should ensure that there's a channel for passing information to team members working on the solution and other teams. You should also communicate with clients if the problem affects them.

Avoiding Common Pitfalls

Manage your onboarding

You should be prepared to manage your own onboarding period and be proactive in your approach.

As **Aviv Ben-Yosef** puts it: *"You don't get real onboarding as an executive in most companies. If you're starting a company from scratch, you're not going to get any onboarding because you're creating it. If you're being promoted or hired into this position, it is very rare that you get some sort of real onboarding."*

Learn to be flexible

"Be prepared to learn and adapt. Flexibility, in the world we are living in, is the most valuable skill you can develop. So say yes to everything, take everything as an opportunity and as a challenge to learn."

—Alvaro Moya,
Founder at Lidr.co

Flexibility will be indispensable to your success. **Kathleen Moriarty** agrees: *"It is important to maintain a level of flexibility so that as a new CTO you might adjust your approach or even strategy based on the new information gained as you learn more about this organization, its culture, as well as industry trends and inflection points."*

Kenneth Weesgaard, CEO at GoSimplo, expands on the idea of adjusting to your new role within a potentially new environment: *"Starting off as a new CTO can be challenging. When taking up the position, people assume you already know everything about the tech industry. However, you often need some time to adjust.*

Every company considers different technologies and trends as important to their business. Therefore, it can take some time to get up to speed. It can help to do some research about the companies' software, for example, or the latest technological trends it used. Although you're new to the company, don't be afraid to share innovative ideas while also remaining flexible.

Being a CTO is all about leading a team, and inspiring them to think outside the box and use new technological trends to the company's advantage."
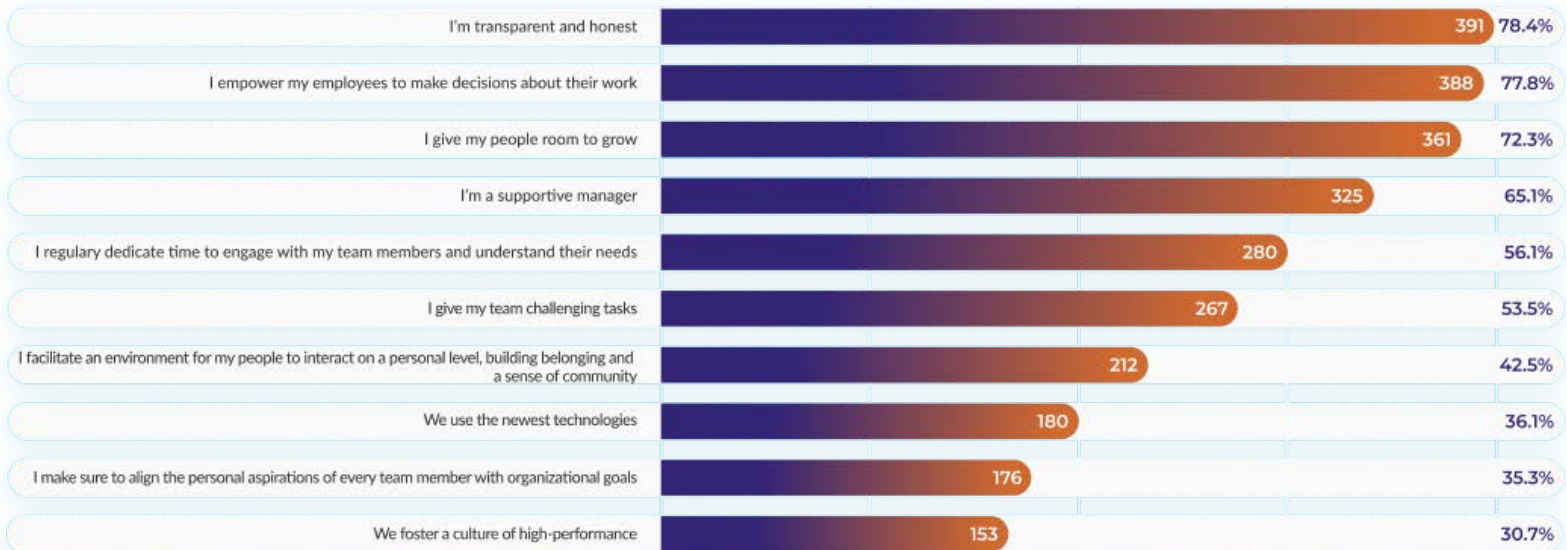
Keeping Your Team Motivated

In our 2021 Global CTO Survey Report, we inquired tech executives how they keep their teams motivated.

When asked about their winning strategies for keeping motivation high, our respondents typically said they focus on honesty, trust, and growth for team members.

Those options won over some interesting alternatives; for example, only 36% said they motivate their staff by using the newest technologies, and only 54% said that they give their team challenging tasks to boost motivation.

Strategy	Count	Percentage
I'm transparent and honest	391	78.4%
I empower my employees to make decisions about their work	388	77.8%
I give my people room to grow	361	72.3%
I'm a supportive manager	325	65.1%
I regulary dedicate time to engage with my team members and understand their needs	280	56.1%
I give my team challenging tasks	267	53.5%
I facilitate an environment for my people to interact on a personal level, building belonging and a sense of community	212	42.5%
We use the newest technologies	180	36.1%
I make sure to align the personal aspirations of every team member with organizational goals	176	35.3%
We foster a culture of high-performance	153	30.7%

Bring harmony to your team

It's much harder to make people feel motivated when there isn't a good atmosphere on your team. Here are five tips to help your team become a place of unity, understanding, and good rapport between colleagues.

1) Define clear objectives

Clear, concrete, and transparent goals are important for every successful project. Even with objectives in place, a huge number of teams don't successfully execute their goals.

Why is that? It's often because teams spend much more time on *how* to do things instead of *why* they want to do them.

This usually results in implementing unnecessary functionalities or causing huge refactoring efforts in a freshly delivered release. It also undermines the team's motivation and engagement.

Productive teams focus on the "why." They ensure that their goals are well-defined, known, and understood by everyone. Productive teams work with their Product Owners and stakeholders to understand the value of the planned features.

How to define clear objectives:

- Share your vision with the team and ensure that all team members understand the reasoning behind the project.
- Visualize your goal using techniques such as the product vision box.
- Identify your stakeholders. Check whose problem you're going to solve with your solution.
- Support your team daily and help them validate their actions against your goals.

2) T-shape your team

In the world of rapidly changing technologies where specialists are worth their weight in gold, a team built on solid cross-functional foundations is instrumental for every successful project.

It's a rare situation where tasks can be ideally mapped to the core skills of your team members. There are always too many or too few tasks for either backend or frontend developers, or the amount of work on test automation or manual tests sometimes exceeds the number of available testers.

This leads to delays in development due to a bottleneck of skills.

The key to solving this is to build teams with T-shaped skills. The T-shape describes someone with deep vertical understanding in one area (such as backend development or test automation) but also with broad, though usually not deep, knowledge in other areas.

Imagine a Python developer who is great at using the Django framework but can also do a little bit of frontend and create basic E2E automated tests—that's a T-shaped team member.

T-shaped employees have a positive impact on the scalability of the team, but more importantly, they can eliminate skill bottlenecks.
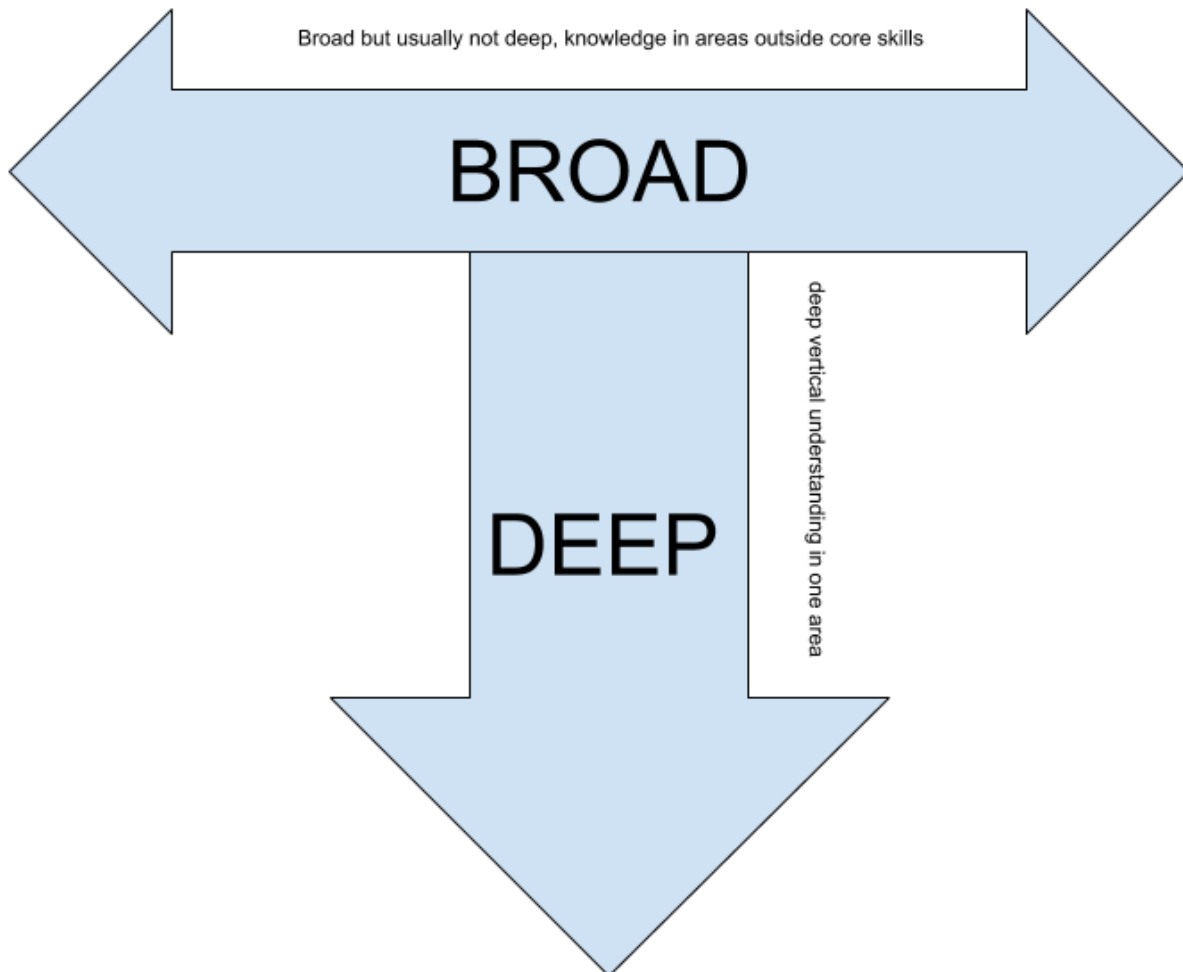
How to create a T-shaped team:

- Reward your team members when they improve their versatility.
- Encourage and help your team work outside their core skills and comfort zone.
- Offer your team a training budget.

3) Stop starting, start finishing

The low effectiveness of any team is often the result of a high amount of work in progress.

High WIP can be overwhelming and detrimental to productivity. Working on more than one task at a time also requires costly context-switching.

Productive teams are aware of the true cost of multitasking and avoid it at all costs. They seek to limit WIP no matter what type of development method they're using.

How to avoid high WIP:

- Limit your tasks in progress to your number of manual testers—the WIP should be equal to or less than your number of manual testers—or try the swarming approach by applying the whole team to one ticket/story.
- Visualize your task board and discuss it on a daily basis.
- Work with the team to prepare your "Definition of Ready" and "Definition of Done."

4) Build trust with your team

Good managers use trust as a basic tool for project development.

"You need to trust your team, but don't take trust for granted. It takes time and it's very expensive to build up. It's also very easy to break it," says **Junior Godoi.** *"Obviously, you have to rely on psychological safety and an open environment, but once you build that trust you don't need to worry about micromanaging people. They know what to do and you are confident they will achieve the results. So, focus on trust and you'll get all the rest as a bonus."*

A no-blame environment creates a spirit of trust. If mistakes happen, use them as an opportunity to work collectively on the best process solutions for future use.

How to build trust:

- Focus on results and account for results.
- Let your team fail from time to time and help them learn from their mistakes.
- Give honest feedback based on facts not opinions.

- Don't provide solutions.
- Give your people your time and be always around to support them.
- Encourage your team to build their own processes.

5) Continuously improve

A development team needs forward momentum. As Johann Wolfgang von Goethe famously said: *"He who moves not forward, goes backward."*

Fortunately, modern development frameworks have improvement processes in place, such as the retrospective.

Productive teams use retrospectives for continuous improvement. They don't limit the "retro" only to the development process; they also focus on products, the process, and improving the attitude of the people involved.

The best teams understand the power of small changes.

Improving your skills as a leader

Honing your skills should be a continuous process. Whatever stage in your career you happen to find yourself at, it's always a good idea to develop your existing expertise and acquire new competencies.

For new CTOs, there are some key skills that will make your new role easier. **Stuart Kelly** listed the top three skills tech leaders (and aspiring tech leaders) should be perfecting:

1) Good presentation skills

While you're expected to have the technical skills required to succeed in your new position, it's important to practice and develop great presentation skills to be an effective leader. According to **Stuart Kelly,** *"Good presentation skills are the number-one skill every tech leader must have, as most of the responsibility will involve communication."*

You will need to present arguments to software engineers and make presentations to the executive team. You will also need to communicate the company's plans to your team and give frequent activity updates. You need great presentation skills to execute these tasks competently.

2) Strong mediation and conflict resolution skills

Apart from making presentations, a substantial part of your role as a new tech leader will be managing people, mediating disputes, and resolving conflicts.

You may also have to mediate and resolve disputes arising within your team or between your team and other teams within the organization.

3) Great interpersonal skills

You may be in a situation where you'll need to request that a new leadership role be carved out for you or present other requests that generally aren't easy to make.

Having a good relationship with the executive team members and your colleagues will make putting such requests forward easier. Thus, it's advisable to develop great interpersonal skills to build and maintain good relationships both within and outside your organization.

Look for a mentor

One way to improve as a manager is through a mentor. **Sergey Velts** promotes the idea of assigning a mentor: *"For new CTOs, it's good to have a mentor or at least communicate with other CTOs in a particular industry. There are many professional associations, such as CTO Craft. Conferences are also a good source of contacts and they offer space for discussions.*

It's also helpful to pay attention to how other CTOs present themselves. The role of a CTO is not only about tech but also about tech marketing and advocacy, HR brand, product marketing, and investor relationships. Consequently, it's important to develop public speaking skills."

Final Thoughts

As a new CTO, your journey is just beginning. We hope that the information in this ebook will help you navigate your way through the challenges you will inevitably face.

There will also be many rewards along the way, so don't forget to acknowledge the successes you will encounter. Continuing to seek insights and inspiration will give you a competitive advantage among your peers.

With that in mind, be sure to subscribe to the STX Next newsletter and follow us on LinkedIn where you'll find out about upcoming interviews with industry experts as part of our Tech Leaders Hub series.

You can also find a range of useful resources on the STX Next website, such as:

- The Ultimate Guide to Hiring Software Developers
- Tech Leaders Hub: Management & Growth
- A Practical Guide to Outsourcing Software Development

About STX Next

At STX Next, we provide external talent to software projects through comprehensive end-to-end development, instant team extension, or expert consulting services.

Specializing in Python, JavaScript, and React Native, our 500+ professionals include Scrum Masters, Product Owners, DevOps Engineers, Machine Learning and Data Engineering Specialists, Service Delivery Managers, and a dedicated Product Design team.

To learn more, visit our website or reach out to us directly!

With special thanks to...

Aaron Podolny, Adam Craven, Alvaro Moya, Andrew Beacock, Aviv Ben-Yosef, Dor Zilka, Eric Billingsley, Gregory Yong, Hendrik Wallbaum, Hugh Cumming, Junior Godoi, Karim Zanaty, Kathleen Moriarty, Kenneth Weesgaard, Matthew Shiner, Patrick Kelley, Rohit Sinha, Sergey Velts, and Stuart Kelly—for all of your insightful contributions.

Hire an exclusive
Python development team

Accelerate your software project with Europe's largest Python software house.
For companies with big projects and fast deadlines.

Team Extension

Additional developers or experts supporting your development efforts within 14 days

End-to-End Development

Full development team taking your project all the way from discovery to deployment

Consulting & Expertise

Solving your problems or improving your product with the help of subject matter experts

Administration
Recruitment
DevOps
Management
Product Design
Agile & Scrum
Testers
JS & Mobile Developers

Python Developers
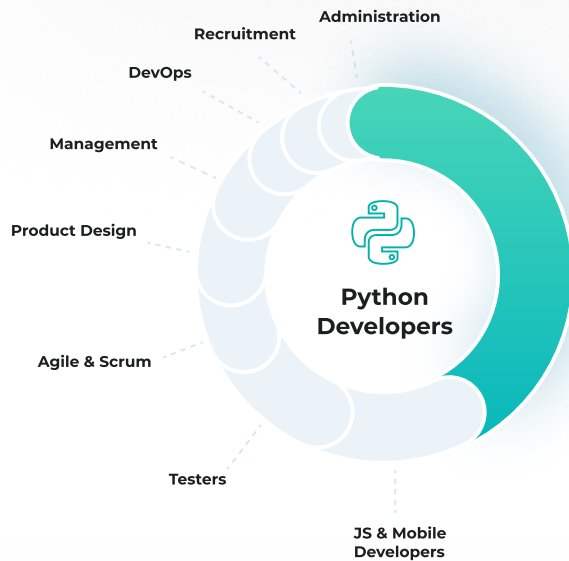
Over 400 developers

Ready to empower any project with well-reviewed code and a results-driven Agile process

17+ years
market experience

750+
projects delivered

3.5+ years
average partnership

300+
clients served

550+
professionals on board

6.5+ years
average experience of our developers

Locations

Poznań (HQ) ◉

Mostowa 38
61-854 Poznań, Poland
+48 61 610 01 92

Wrocław ◉
Olsztyn ◉
Katowice ◉
Łódź ◉
Hague (Netherlands) ◉

Resources

Arm yourself with the expert knowledge you need to successfully deliver software projects.
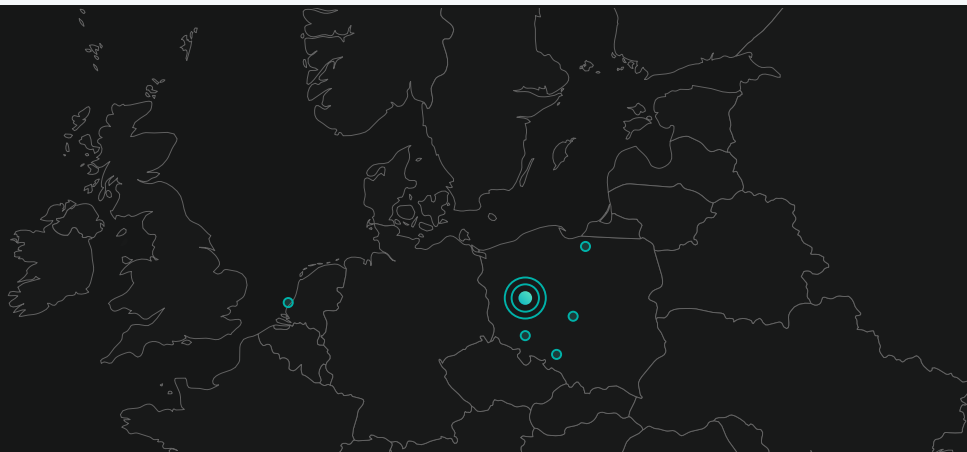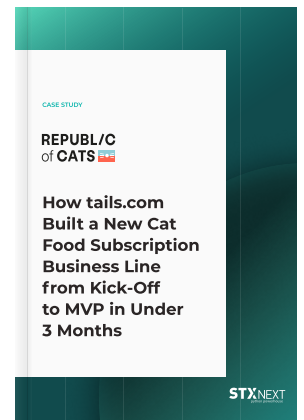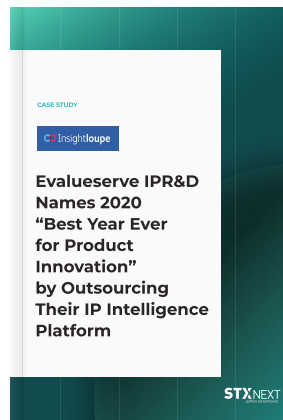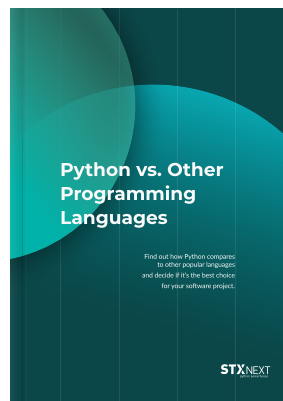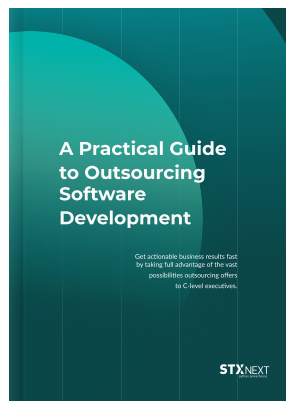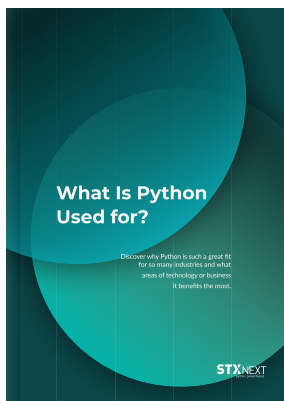Get free in-depth resources, templates, and checklists—all based on 17+ years
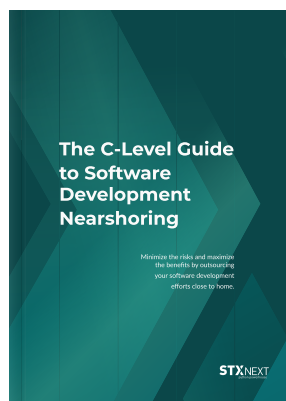of software development experience.

Reports and case studies

TheGlobalCTOSurvey × STXNEXT

SURVEY RESULTS & INSIGHTS

The Global CTO Survey 2021 Report

CASE STUDY

LUMICKS

How LUMICKS Validated and Built a Platform to Help Researchers Analyze Diseases at the Single-Molecule Level

STXNEXT

CASE STUDY

Insightloupe

Evalueserve IPR&D Names 2020 "Best Year Ever for Product Innovation" by Outsourcing Their IP Intelligence Platform

STXNEXT

CASE STUDY

REPUBL/C of CATS

How tails.com Built a New Cat Food Subscription Business Line from Kick-Off to MVP in Under 3 Months

STXNEXT

Guides

What Is Python Used for?

Discover why Python is such a great fit for so many industries and what areas of technology or business it benefits the most.

STXNEXT

A Practical Guide to Outsourcing Software Development

Get actionable business results fast by taking full advantage of the vast possibilities outsourcing offers to C-level executives.

STXNEXT

Python vs. Other Programming Languages

Find out how Python compares to other popular languages and decide if it's the best choice for your software project.

STXNEXT

The Ultimate Guide to Hiring Software Developers

Everything you need to know about growing your software development teams—both onsite and remote.

STXNEXT

Ebooks

The New CTO's Handbook

Start your new role as CTO the right way with practical advice from senior tech executives. Learn how to prepare and what to expect.

STXNEXT

The C-Level Guide to Software Development Nearshoring

Minimize the risks and maximize the benefits by outsourcing your software development efforts close to home.

STXNEXT

TECH LEADERS HUB

Tech Leaders Hub: Management & Growth

Become a great leader by leveraging the huge experience of tech leadership experts who have spent years managing and growing teams.

STXNEXT

The True Cost of Hiring In-House Developers

From training through benefits to paid leave, the cost of adding new members to your team is never just the salary.

STXNEXT

DISCOVER MORE →

Services

Your project is all that matters. We'll build it like it was our own. Whether it's team extension, end-to-end product development, or expert consulting you're after, we'll do everything in our power to meet your needs.

Python Development	JavaScript Development	.NET Development
Web Development	Software Testing & QA	Mobile Development
Django Development	Node.js Development	React Native Development
Fintech Development	Machine Learning	Data Engineering
DevOps	Product Design	Discovery Workshops

REVIEWED ON
Clutch ★★★★★ 82 REVIEWS

python SOFTWARE FOUNDATION

TOP PYTHON & DJANGO DEVELOPERS 2021 — **Clutch**

TOP DEVELOPERS — **Clutch** POLAND 2021

TOP CUSTOM SOFTWARE DEVELOPMENT — **Clutch** COMPANIES 2021

Tell us about your project

Speed up work on your software projects and outpace the competition.

HIRE US →

Marta Błażejewska
DIRECTOR OF SALES
marta@stxnext.com
+48 506 154 343

Sebastian Resz
HEAD OF SALES
sebastian@stxnext.com
+48 690 433 578

Follow us