TECH INCLUSION

IN SOUTH AFRICA

2019

OFFER ZEN

ABOUT THIS REPORT

This is a compilation of experiences from stakeholders in the tech community and the ideas of those who are working to create a more inclusive and diverse tech industry. In relation to these efforts, we are focusing on three themes, namely: teaching technical skills, preparing people for work, and helping people thrive in the workplace. We aim to explore the thinking required by hiring managers and employers when they are trying to implement or scale diversity and inclusion within their organisations.

This report is an inaugural body of work from OfferZen Foundation, OfferZen's not-for-profit organisation. Our mission is to help underserved South Africans thrive in their tech careers by acting as a rally point for discussions and actions that help combat the inclusion and diversity challenges that are pervasive in the tech industry.

As an organisation, we recognise that it is important to tackle these complex challenges. We hope that this report sparks further discussion about the challenges that are faced by underserved people within companies in South Africa. In turn, we can all start working together to create systemic opportunities and solutions that drive inclusion and diversity in the tech community.

If you would like to get in touch, email foundation@offerzen.com

CONTENTS

Click on the titles below to go to the section

INTRODUCTION

South Africa's tech industry is seeing a disconnect between people from underserved communities looking to get into a career in tech and companies looking for people with software skills. While a growing number of companies are looking to scale their existing tech teams, and others are bringing software development into their businesses for the first time, people with unconventional backgrounds still struggle to enter and thrive in the tech industry.

Currently, the majority of people in the South African tech industry are white and middle class. At the rate at which demand is growing for software roles, this group of candidates alone won't be enough. That's why the only way to sustainably grow our tech companies, and thus South Africa's tech industry, is through diversity and inclusion. We need to broaden our view on what someone in software development looks like, where they come from, and how they work.

Oftentimes, diversity and inclusion are used interchangeably. Indeed, while closely related, they are not the same thing:

Diversity speaks to the range of differences within a group, such as race, educational background, gender, sexual orientation, or religious beliefs.

Inclusion, however, describes *whether* and *how* individual people within this diverse range are included in a work environment. It is not necessarily only about ensuring that you have black women in your company, but goes a step further: Is your company a place where black women are respected, have a voice, and are supported to do their best work?

Within the South African context, people from non-traditional backgrounds are most often non-white. As a result, the focus of this report is on diversity and inclusion as it relates to race.

We also use the term 'underserved'. This relates to, for example, having a lack of access to good quality education, transport, and healthcare options. These factors impact what people from those environments learn, and what opportunities are immediately available to them. In the work context, not understanding that reality, or acting under incorrect assumptions of the underserved community, can be alienating. It can make people from underserved communities feel unwelcome.

So far, the South African conversation has centred around diversity, with a specific focus on training. Funding from corporate sponsors, as well as national initiatives like YES and The Jobs Fund, has helped create the boom around coding programmes, or bootcamps as they are commonly known. B-BBEE also incentivises this focus on training, by awarding points to companies on the Scorecard for skills development.

As OfferZen Foundation, we believe that, in order to realise the potential of people from underserved communities in tech, training alone is not enough. In addition, these two things need to be understood:

• What support they'll need as they enter the workplace
• How to ensure that our work environments are spaces where they feel included

Given South Africa's history of Apartheid, and the resulting socio-economic challenges, addressing diversity and inclusion means having uncomfortable conversations and acknowledging our biases.

Our report aims to uncover these issues by exploring what companies can expect from people with software skills as they're coming into the industry. It explores what companies should focus on when helping people from non-traditional backgrounds, and discusses why it is unlikely that they will see results of their diversity and inclusion efforts without a fully inclusive workplace.
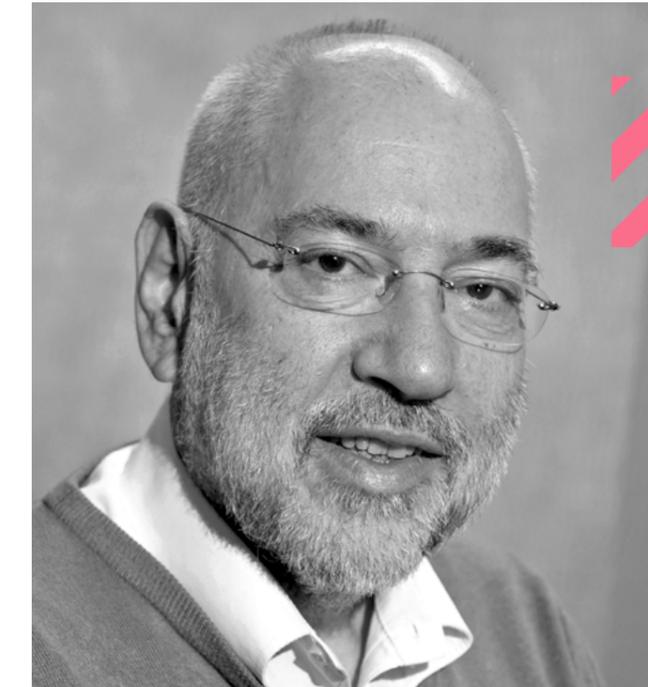
We're excited to share these ideas with you.

Tumi Sineke
Head of OfferZen Foundation

TEACHING THE SKILLS

When growing a tech team, companies need to know what they can expect of people from non-traditional educational backgrounds. In this theme, we explore why this is important, and provide examples of what non-traditional educational backgrounds may look like.

THE IMPORTANCE OF TECH INCLUSION IN SOUTH AFRICA

Q & A with Professor Barry Dwolatzky,
Joburg Centre for Software Engineering

Demand for software makers is outgrowing the available talent pool. That's why it has become more important for companies to leverage candidates from more diverse backgrounds and educational paths. With more than 30 years of research behind him, JCSE Director Prof. Barry Dwolatzky has a clear idea of what companies can expect from these candidates and how they can develop their full potential. He sat down with us to share what he thinks the current problems are with creating these opportunities for people and how we can begin to address them.

TEACHING THE SKILLS

Who is Professor Barry Dwolatzky?

When Professor Barry Dwolatzky started his PhD at Wits in the 1970s, the age of computing was beginning in South Africa. Despite his focus on electrical engineering, he had a keen interest in the developing digital realm.

During the Apartheid era, the South African IT industry functioned very much in a bubble. Despite the lack of international companies operating in the country, and thus a lack of external competition, developers were producing technology that was at the cutting edge of IT. Then, with the dawn of the new democracy, came the opportunity to build on South Africa's existing expertise and open it up to more people. This growth was something that Professor Dwolatzky became increasingly passionate about.

He wanted to figure out how to bring underrepresented groups, such as young people, black people and women, into the fold to develop new talent that could help cement South Africa's presence on the international IT stage.

To pursue this mission, he founded Wits University's Joburg Centre for Software Engineering (JCSE) in 2005 and currently acts as its director.

What is the status quo in the current South African tech industry that makes this topic so relevant?

I think that, for a number of reasons, the great opportunity that the South African tech industry was presented with wasn't realised.

We didn't capitalise on what we could have done, but I passionately believed then, and I still believe now, that we do have an opportunity. I think the next wave of opportunity is in the so-called fourth Industrial Revolution, and digital transformation of everything.

In order to really grab this opportunity, we need to grow the number of digital skills that we have in this country. Despite what people may say about machine learning and AI taking over everything, tech is still very people-centric – it's a human-centric activity to develop software – and so, we've got to pull out all the stops to grow the number of people that we have doing this if we want to make a global impact.

"What feeds into this problem is that our universities and other formal education systems cannot do enough anymore. They're simply not big enough to grow the skills we need: They do not have the space or the resources to cope with the demand, so we have to look at alternative ways to grow these skills."

What skills can companies realistically expect students from non-traditional backgrounds to have when they arrive for their first day of work?

Depending on which training programme they come from, most students from less-traditional streams will arrive at work with some basic knowledge of a modern language's syntax. They don't come as a ready-wrapped package, ready to be deployed, because the time and energy that it takes to set someone up to be "work-ready" has not been invested in them.

Many of the coding bootcamps that are out there focus on churning out large numbers of candidates who have three or four months of coding training and that's it. This is not enough. What I believe students really need, and is something that good organisations really focus on, is **practical experience.**

When you are working with people who have never been exposed to coding or programming before, don't have role models, and haven't ever seen or interacted with other developers, there is going to be a gap.

Too many programmes are not looking at how to give students the opportunity to practise a practical application of the technical skills that they have learned to real-world problems. On top of this, when students are not getting this opportunity to experience what they might face in 'reality', they are not being exposed to important soft skills, like communication, problem-solving and interacting with their peers, either.

This does not mean to say that the student, or candidate, will never have the potential to be an excellent or 'world-class' developer: If they can demonstrate an aptitude and the right

attitude, then they will get there. How long this takes, I think, is largely dependent on the company and how determined they are to set the candidate up well.

How different is this to someone who comes from a more formal stream of education?

Someone who has studied a computer science degree at UCT or Wits will actually have less practical IT skills than the type of candidate we were just discussing. They will have done some programming courses as part of their degree, combined with other subjects like maths and physics.

So, their background will be largely theoretical, which I've seen frustrates a lot of companies. They say that these students aren't work-ready and that universities aren't doing their jobs because they aren't producing the right level of talent.

However, the difference that better sets university graduates up for success in the long-run is that they have three years of learning behind them. They were taught to problem-solve, communicate, and gain confidence in their ability to learn more by themselves.

If a candidate has done a good degree, they will come ready to learn and to move forward very rapidly in terms of acquiring new skills, whereas a bootcamp graduate has not been primed to do this through their training.

Why then should companies proactively think about including people from more diverse backgrounds in their teams?

As I just mentioned, there aren't enough graduates feeding the industry from the more formal streams.

I mean, even the number of people passing matric, and then getting into a computer science degree, is too small.

We need to look at other ways of doing it. I kind of see skills as a pyramid: I see that at the base of the pyramid, which is very wide, you need a lot of skills at the level of

expertise lower than a graduate. As you go up the pyramid, it gets thinner, and you need fewer people of higher qualifications. At the peak of the pyramid, you have graduates, which is a very small number. This is not to say that graduates are the most important people in a team. Through the analogy of a pyramid, I am not demonstrating hierarchy but rather the importance of numbers; if you do not have a broad, solid base, the pyramid cannot exist.

Essentially, if you only have graduates, you can't build big teams. If we don't have big teams, we can't build a very big system, which means we will never succeed in the IT industry.

This is not just lip service, but I believe that diversity in the workplace is an organisation's greatest strength. If you have a sort of monocultural system, where everyone is a white person from a Model C school in the suburbs, then the thinking, the creativity, the social mapping that happens is very narrow, and companies are missing out. If they embrace diversity, then they are benefiting from the amazing opportunity we have living in such a diverse country.

With this understanding, how can companies begin to think about working with candidates from less-traditional education streams?

In my opinion, a very useful place to start is with the practical experience that is missing from the initial training programmes.

I think that it's important for companies to understand the power of an internship. When a company takes on a person, they should not have huge expectations about what they can do, but trust that they have the attitude and aptitude to learn.

If you look at the Indian IT industry, a lot of the large companies recruit students from great Indian universities, and then send them off for six or nine months onto a graduate internship where they learn the company's way of working and the practical skills they need. This would be a fantastic model to apply here.

I think a lot of South African companies are lazy. They want to take someone from a training programme, put them to work, and start making money from them. I think it needs investment on the part of the company to grow the talent, to round the person off, and make them job-ready. This has to be done at the company's expense, which a lot

of companies don't do. We need to address this thining, and change it from technical team leads grumpily refusing to work with "raw" individuals when they expected someone with five years of experience, to wanting to foster growth.

What practical steps can companies take to actually win at this?

On a base level, we need to show companies that having more diverse teams can benefit them. As an entry point, perhaps we could leverage B-BBEE incentivisation. A lot of companies recruit people with one eye on their B-BBEE scorecards, and so we should look at how we can make use of this to benefit everyone involved.

> **Companies should start thinking about how these intakes can help their entire team grow.**

If they can do this, we will see less of these new candidates being dumped in a corner and given rubbish work to do.

By putting a newbie in a team, and focusing on helping them get to where they need to be to succeed within that team, a culture of sharing and learning amongst peers is encouraged. This is beneficial to everyone as you begin to break down barriers of awkwardness and include every member of the team.

> **Team leads should take the time to learn what it means to build a solid base for their team.**

By doing this, we will see more candidates from less-traditional backgrounds being welcomed in.

When HR managers are the ones driving job searches, it is easy for a disconnect to happen and team leads to be frustrated when their hiring expectations are not met. If team leads are the ones driving recruitment, then we start to see more of a team ethos developing.

> **The focus should shift from what the candidate can do in the short-term to what they can do in the long-term.**

Similar to what I previously mentioned about university graduates, if a candidate can develop skills that will help them in the long-term, then they will add more value to a company indefinitely. Instead of viewing them as immediate money-makers, companies should focus on training them to be problem-solving individuals. A large part of how they can do this is by encouraging soft learning.

To get a person into a mindset where they have to go and find answers to questions themselves, you have to stay away from spoon-feeding and laying everything on a plate. An effective mentoring system should focus a lot on encouraging people to think for themselves: They come and sound stuck, and you point them in a way that they can get themselves unstuck, rather than you unsticking them.
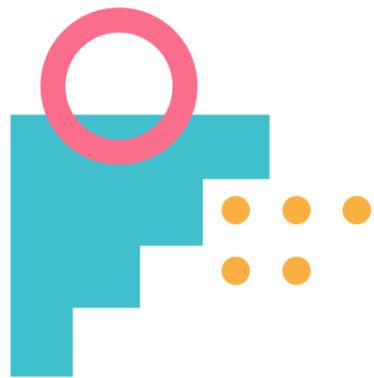
By focusing on these kinds of ethics when it comes to working with other people, development becomes a really communal experience, that everyone can benefit from.

Reflecting on everything that we have discussed, is there anything else you would like to share?

I think, and it's something I said right at the beginning, that the problem that we face when it comes to inclusion is a national question, and therefore something that we should all care about. As I said before, South Africa has had a huge opportunity to become a key player in the digital world, which we lost out on before but have been lucky enough to be presented with again now.

We should grab this opportunity as a country, because if we don't, we'll be all the poorer for it. I think South Africans are very, very talented; we are creative and innovative, and have all sorts of amazing attributes.

We need to turn our attention away from the nonsense that comes with people competing for skills and the wrong incentives being put in place, and rather focus on growing this huge cohort of skills we need to embrace the digital age.

———

EXPLORING CODING BOOTCAMPS IN SOUTH AFRICA

by Tumi Sineke

Many people who might have an aptitude for a career in tech either can't afford to study formally towards a qualification, or are unaware of the opportunities that are available to them, such as coding bootcamps. That said, it's also hard to get an overview of the coding bootcamps in South Africa and what they offer. Here are the most important facts on costs, time and opportunities of **tech bootcamps offering the potential of a career in the tech industry.**

What are bootcamps in South Africa?

The concept of a coding bootcamp is said to have originated in the USA in late 2011. Since then, it has taken off across the world and now, bootcamps are largely paid-for programmes that offer people an intensive period of upskilling to help them move in tech careers.

In South Africa, bootcamps offer the same promise of setting students up for a successful move into a tech career. The thinking behind establishing bootcamps in South Africa seemingly aims to address the skills shortage that is described by companies in this industry. You can read more about this here: Johannesburg Centre for Software Engineering's report that surveys ICT Skills shortage in South Africa.

Who are bootcamps aimed at?

Coding bootcamps in South Africa tend to attract young people from underserved communities. CodeX, for example, has an 87% black student enrolment rate. Their sell is a career in a high-paying, in-demand job, so this proposition seems like an attractive option to those who are interested in tech but don't have the means to get into it through more formal streams.

What are the entry criteria?

Many of the bootcamps require students to have completed matric. Students also need to show an aptitude for software programming, which is determined by an online assessment. Thereafter, their attitude and motivation for learning is assessed, with some programmes taking students through an in-person interview, or, in the case of WeThinkCode, a selection bootcamp from which the final cohort is selected.

Umuzi is working on new psychometric-based assessments that they believe will give them a better idea of whether an applicant will be successful in tech, strategy or creative careers.

How much do bootcamp courses cost?

Free courses

In most cases, attending coding bootcamps is free. This actively addresses the challenge of providing access to those who might have an aptitude for a career in tech but can't afford the education, even at a price-point lower than a formal university degree.

In the cases where the bootcamp programme is free to students, the cost of running it is sometimes covered by companies. These companies will often either sponsor students by covering tuition and living expenses, or host them as interns. Some companies also make a pledge to hire a number of students at the end of the programme.

Paid courses

Bootcamps that charge their students fees also exist. In most cases, though, payment plans have been worked out to keep these bootcamps as inclusive as possible. HyperionDev, as an example, accepts both monthly and upfront payments. They also make an effort to offer discounts where possible.

Pay-it-forward

CapaCiTi seems to be the go-between of the two options. While the bootcamp programme itself is free while you're studying, their model is "pay-it-forward". This means that students are required to pay back their tuition once they've found work. The amount students need to pay back is tiered, depending on their income. For example, if a student earns R12 000 per month, they could pay back 10% of what they owe per month.

How are bootcamps funded?

As mentioned before, bootcamp programmes often receive funding from companies. This may come in the form of donations, or companies might opt to sponsor students throughout the programme. Alternatively, some bootcamps also receive grant money.

How do bootcamps work?

Many local bootcamps fall into the formal classification of a learnership. Learnerships are part of the South African Qualifications Authority (SAQA) framework, which means that they are made up of theoretical and practical applications. In a coding bootcamp, this could mean being taught a Javascript framework, then working on a project where you use that framework practically.

Getting into the bootcamp

Typically, bootcamp programmes begin with an online assessment, which seems to look at reasoning ability and maths. This is then followed by an in-person, intensive pre-bootcamp that tests skills, before the final student group is chosen to begin the formal programme. This is how WeThinkCode, Project CodeX and Umuzi operate.

Other programmes, such as HyperionDev, Quirky30 and CapaCiTi, do not include an intensive selection pre-bootcamp before selecting their final cohort. The reason CapaCiTi and Quirky30 don't include this session is to reduce barriers to entry for people who want to learn; HyperionDev is a for-profit company, and so has more of a financial incentive to allow as many students as possible to enrol.

Teaching approaches at the bootcamp

Once the selected cohorts begin their programmes, different bootcamps approach teaching the skills in a variety of ways. For instance, Umuzi begins with teaching technical skills before placing students in cross-functional teams. Because Umuzi accepts students into web development, UX, UI, data and copywriting streams, the teams work together to create solutions that make use of this wide variety of skills.

Duration of bootcamp courses

In terms of duration, most programmes run for three to nine months. There are exceptions, of course, such as WeThinkCode. Here, students embark on a two-year long full stack development programme, which includes two internships. HyperionDev offers the shortest programme, where students spend three months working on a web development course.

How do bootcamps help set students up for employment during the course?

Partnerships

In terms of how they help guide students towards employment opportunities, many bootcamp programmes partner with technology providers, or other members of tech ecosystems. For example:

- **HyperionDev** has, since launching, advertised their partnership with The Python Software Foundation, which helps give credibility to their Python course.

- **WeThinkCode** has recently partnered with a range of organisations, including Harambee, Tshimologong Precinct, Explore Data Science Academy and CapaCiTi to focus on creating better opportunities for graduates to find work. You can read more about this here.

Work-readiness and interview prep

Learning how to code is not the only training that students receive at local bootcamps. All of the programmes that I looked at offer either work-readiness or interview prep as part of their courses. CodeSpace, for example, offers an extended programme that runs two weeks longer than their core programme to prepare students for the office environment.

Giving students opportunities to practise skills that companies find valuable, like communication and teamwork, is something that Project CodeX and Umuzi also offer:

- **Umuzi's** cross-functional product teams provide a good opportunity for their students to practise skills like collaboration and communication.

- **Project CodeX's** students work in teams on a week-long hackathon at the end of each semester.

How do bootcamps help set students up for employment after graduation?

Once students leave the programme, there are few options for the next steps.

Employment

For those who are looking for employment, a few programmes have industry partners who absorb interns from the programmes into full-time positions. These industry partners vary, from large corporates to SMEs. Umuzi, which offers programmes that are popular in creative agencies, has partnerships with the likes of King James, HelloComputer and VML.

Entrepreneurship

For those who want to use their skills to start their own businesses, bootcamps like mLab provide support for ventures that have been founded by their graduates.

How are employers encouraged to hire bootcamp alumni?

When bootcamps are SETA accredited, they provide the incentive for companies to sponsor students and have this sponsorship count as points towards their B-BBEE scorecard. While this benefit means that coding bootcamps are more likely to secure funding from companies, it doesn't guarantee anything else for the student.

There are many options for young people who are looking to enter the tech industry. At the same time, the magnitude of the challenge to get young people, specifically from underserved communities, to a place where they can succeed in the tech industry is striking. While there are a lot of exciting programmes teaching tech skills, there is more work to be done to understand how we can improve access to full-time work opportunities to help young people reach their potential.

> **What still remains unanswered is how to encourage companies to employ graduates from these programmes.**

Even though some companies may invite students to join them as part of an internship programme, they are not required to hire them on a permanent basis at the end of the course. This is part of the challenge with creating work opportunities:

Companies are measured on the percentage of black people that they take through internships, learnerships or apprenticeship programmes. There are some bonus B-BBEE points for the number of black people absorbed into a company after a learnership programme, but what is not specified is where those young people go within a company.

With this understanding, it seems that even though graduates from bootcamps have received some training in tech, there is still a gap that needs to be filled before someone can deliver real value in a tech team.

LAUNCHING A CAREER AS A SELF-TAUGHT DEVELOPER

Conversation with Tumiso Marebane

When you have limited resources available, deciding to make a career pivot can be tough. Tumiso Marebane discovered this when he dropped out of his previously sponsored mechanical engineering degree. He quickly realised that what he really wanted to do was code. With no qualification behind him, and the pressure of needing to make ends meet, he taught himself the basics of programming between tutoring and truck-driving. Here's how he overcame the challenges of his situation in order to pivot into the reality of becoming a developer.

Tumiso grew up in a village in Limpopo. When he finished high school, he was offered the chance to study at Wits and, in his second year, he was offered a full bursary to study mechanical engineering. Even though this was not what he had wanted to do, this bursary would offload a huge amount of financial strain from his parents. In order to set himself up for something more progressive, he decided to take the opportunity.

"I was a horrible student," Tumiso says. Having not wanted to study engineering in the first place, finding the motivation to apply himself was hard. "My performance was nothing to be noted and, eventually, in my third year, I was kicked out." Failing classes had excluded him from writing exams and the company that was sponsoring him eventually revoked his bursary.

Tumiso knew that he had to contribute to his family financially, and so, with no qualification behind him, he went out looking for odd jobs.

Initially, he tutored highschool students and drove a truck for a bottle recycling company. During his degree, though, he had briefly studied mechatronics and really enjoyed it. He started thinking about programming and how he could potentially make money from it. He decided to try his hand at building basic websites and it was at this moment that a lightbulb went on:

> **"I'd always liked the fact that you can write something and it stays there, and does what you say it should. I decided I wanted to be good at this, so I started reading, creating, and then recreating."**

For years after this, Tumiso dedicated hours every day to teaching himself how to code. "I recognised that this could be my chance to be happy and to contribute to my family at the same time – I didn't want to suffer anymore and I found that desperation to be a great motivator," he says.

Today, Tumiso is a mid-senior developer at Superbalist, one of South Africa's most prominent tech companies. "It took a lot of hard work for me to get here," Tumiso shares. Reflecting on his journey, he attributes his success to three key factors:

- Committing to his passion
- Making smart use of what was available to him
- Learning continuously

Here's how he went about this:

Committing to his passion

Thinking about his experience at university, Tumiso realised that he couldn't put his head down through the tough parts of mechanical engineering because he didn't have any drive for the subject matter. "I think, in essence, I didn't have the aptitude," he says, "but I also struggled very hard to do something that I didn't enjoy."

Coding was different: The challenges it presented really fascinated Tumiso, and working through them made him feel incredibly satisfied. Once he decided that coding was what he wanted to do, he set up a rigorous plan to make learning a priority. "I had given up on something once and I didn't want to do that again," he says. "I had to really take the time to be sure about this because it was going to take a lot from me. Once I decided I was sure, I had to work out how I could keep myself motivated."

Setting aside time

In between all of the odd jobs that he worked, Tumiso set aside time every day to practice coding. "I would go and sit somewhere quiet at night, take my laptop and just code. I wanted to get good."

This level of commitment meant that Tumiso lost many hours of sleep and often missed out on having a social life. He had to be strict with himself because his other jobs were physically demanding and it would have been easy to give in and go to bed early every night. Simply promising himself to look at his computer would never get anything done, so establishing a routine was essential.

Not giving up when things got tough

Sticking to his daily plan – get up, go to work, come home, code for x number of hours – took a lot of willpower, and there were many times that Tumiso wanted to give up.

"It's an odd space to be in when you don't have a fallback," he says. "There are days when you feel on top of the world and excited about what you want to do, and then there are the days when this decision seems like the worst one you could have made. Because I had changed my mind once already, I knew I had to stick this decision out."

> **Tumiso found that jotting his thoughts and feelings down on paper helped him to make sense of things.**

In fact, this writing often turned into poetry. "I liked to expose myself to new ways of thinking, and consider how I could do things differently. This really helped me get out of my own head when things got tough. I found that taking this sort of break helped me reset my sense of motivation and I would come back to the problem I was facing feeling refreshed."

Making smart use of what was available to him

Initially, Tumiso started building basic websites with just the base knowledge he had picked up online, using WYSIWYG tools, and from his brief interaction with mechatronics. Then he stumbled upon the book JavaScript and Ajax for Dummies by Andy Harris. "I knew I liked to code from what I had learned at university, but then I found this book, and I can't explain it… The way he described coding made me sure that that was what I wanted to do with my life."

> **While Tumiso had found a valuable source of information, one of the problems that he faced working on an ad-hoc basis was Internet access.**

"I was working with a dongle most of the time and didn't have a lot of data," he says. "While I knew where to go as a starting point for answers, I would have to be smart about how I accessed these resources because using the Internet was expensive."

To get around this, he took a lot of screenshots of webpages and kept careful notes. "If you looked at my computer, I had so many folders filled with my own notes and all of those screenshots – I screenshotted everything, including comment sections on pages so that I could read what people had to say without wasting money."

Asking the community for help on specific problems

Being sure to get as much detail as possible when he accessed anything that would help his learning, Tumiso got a real sense of how helpful the tech community could be. The comment sections showed him that a lot of people were willing to spend time helping others who were new or struggling with a particular topic. Reading through comment after comment, he felt more and more comfortable with the idea of reaching out when he was stuck on something.

However, because he had limited time each day to focus on coding, he realised that he couldn't spread himself too thin. When he was focusing on a particular topic, he needed to be disciplined with how he approached it. He couldn't learn everything at once, because of time and mental overload, so he decided to only **reach out when he needed help on a particular challenge:** "I'd think to myself: I don't want to know everything, I want to know this so how do I learn about this?"

> **"The programming community as a whole is very welcoming. If they can see that you're taking initiative, they will help you a lot."**

This really paid off. Focusing on exactly what he wanted to know at a particular moment helped him not to get overwhelmed with information, or distracted by advice or insights that were interesting but not useful. Tumiso found that this was an important foundation to lay for anyone with limited time.

Learning continuously

Committing to the grind

In January 2018, five years after he started to teach himself to code, Tumiso joined Superbalist. He quickly saw that levelling up in a fast-paced corporate environment was very different to what he was used to.

"When I started the job, a lot of what they did was new. I didn't know the kind of tech they were using and I'd only had limited experience in a professional environment," he says.

There was a lot more to learn than just practical skills.

Joining the Superbalist team, Tumiso quickly saw the base that he had built for himself would not hold up against the mammoth requirements that the job had in store for him. He had to learn a host of new technical skills, get to grips with the complex processes that e-commerce environments work with, and adjust his mindset from working alone to working as part of a team.

To do this, Tumiso put his head down and committed to going above-and-beyond the 40 hours a week that were expected of him. "I admitted to myself that I didn't know things and I decided that I wanted to know them and would do whatever it took to achieve this," he says.

While this took a lot of time and energy, it was something that Tumiso was excited about. He had worked incredibly hard to get his foot in the door, and now that he was there, he wanted to keep proving that he had what it took.

That's why he decided to stay late into the night, come into the office on the weekend and use the time to practise what he had learned, do further research, and plan how he could share ideas and questions with his team when they were all together.

Constantly seeking to level up

Because reaching out to the community was something that Tumiso had become comfortable with when he was teaching himself, he found it easy to now approach his colleagues.

Sticking to the approach he had taken before when interacting with more experienced developers, Tumiso made an effort to build a base understanding for himself first and then nail down specific questions he wanted to ask.

Showing this sense of initiative helped Tumiso build a network of people that he is now able to call friends. Because he knew what he wanted from them, and because his focused questions helped them help him straight away, the dynamic between his colleagues and himself was good: Approaching each other was easy and they were excited to share what they knew.

"I was very lucky – they were the friendliest people ever," Tumiso says. "They really would go out of their way – even sometimes missing their own targets to help me. I'm very grateful for them. They are my most valuable resource and knowing that they are 'on my side' makes me look forward to coming into work every day."

Setting up projects to keep learning

Just over a year into his role as a developer at Superbalist, Tumiso is still excited about pushing the boundaries. "I've found that I can't enjoy my life unless I'm moving forward. It's hard work because it means I'm always working, but it's what drives me," he says.

While he is learning a lot on the job, Tumiso uses his free time to practise what he's finding hard, or else just exploring things that interest him. At the moment, he is working on building a real time chat room, similar to WhatsApp, because he is curious about what it takes to make an application like this work.

Making the time to focus on side projects can be difficult, especially when his workload gets demanding, but Tumiso has realised how important it is to remind himself where he came from and why he decided to do what he did.

> **"I don't ever want to feel like I have learned everything because I know that is not true. I know that there is so much out there and I want to keep discovering. Coding got me out from behind the steering wheel of a truck, and into a chair in front of a screen at Superbalist – I owe it to this opportunity to keep exploring it."**

PREPARING FOR WORK

Even with the necessary technical skills, it's important for companies to know what it means to hire people from underserved backgrounds, and how to help them level up in their new jobs. In this theme, we explore what support they'll need as they enter the workplace.

DEVELOPING WORK-READINESS WHERE IT REALLY MATTERS

Q & A with Alwyn van Wyk,
Founder of Younglings Africa

Many people from non-traditional paths aren't always immediately ready to add value to companies. Alwyn Van Wyk is involved in Younglings Africa, and has developed an internship programme that tackles exactly that: Training graduates to be 'work-ready' devs. Here's what it looks like, and how his lessons in doing so can help companies understand and grapple with 'work-readiness' more successfully.

What does the Younglings Africa programme aim to achieve?

I think it's all about the human capacity to be incredible. That's the part that has always been intriguing to me. About ten years ago, I heard a talk by a representative of an Indian company that pointed out a vastly under-utilised group of people in South Africa. They actually went into different metros, saw potential in software developing, trained those people in India, and when they came back, they found them jobs. That blew my mind.

> **I couldn't understand why South Africans weren't doing this for our own people. We're always moaning about skills shortage, but no one is doing anything about it.**

That's kind of where I'm coming from with Younglings Africa: We want to help those who couldn't get into university, because they encountered financial problems or their marks got pulled down because their passion didn't sit with English or the other science languages. Even though they have the brain for computers, they struggle with the rest and might never get the chance to study something like computer science.

Where we come in is to support them, and help them to be more than just a 'factory-worker' kind of software developer, typing away at useless stuff. We train them to be somebody with a creative mind, who can actually figure out problems and is working on something that they're passionate about. **We help them own that skill and accelerate their career.**

You currently run the programme in three stages, each over six months. What are some 'key indicators' of work-readiness that you focus on at each stage?

The first six months are a guaranteed Learnership, where you have an opportunity to prove yourself and specialise in something. For the first bit, we are training our Younglings in their actual coding ability, to get them on the same level. We are looking at their aptitude, attitude, and aspiration: Can they develop incredible software?

The next six months are called the Apprenticeship: This is where the 'pressure-cooker' happens: We ask our apprentices to build internal products, so they can focus quite intensely on understanding what it means to work in a team. We're also making sure their skill set is honed so that, at the end of that six month period, we can expose them to the outside world and say: "Alright, you've survived six months. Now we think you are ready for six months 'controlled in the wild'".

That, then, would be what we call the Internship, which is centred around working with external customers.

How do you define 'work-readiness'?

We obviously teach them the technical skills they need to be a software developer, but what I consider 'work-ready' comes down to simple things around behaviour: Things like promptness, taking pride in wearing appropriate clothes to work, attending meetings, being able to book meeting rooms, and knowing how to greet colleagues in different roles. It almost sounds silly, but those kinds of things are what we're trying to work into the programme so interns get exposure to that throughout their time with us.

> **If the people you're trying to empower don't understand the rules of the work environment they're trying to get into, then you can't expect them to thrive as well as they should.**

They need to be able to come into any job, any workspace, and be able to adapt. I've noticed that those rules have become a lot more 'loose' over-time – things like what you can and can't wear, for example – and that makes it a lot trickier too. Being able to understand those subtleties in behaviour, and to know that what's appropriate in one place over another, is incredibly important to make someone really 'work-ready'.

What about those skills is so important?

If you don't have the guts to start your own business, which is very likely when you're still young and finding your feet, you're going to have to fall in line with other people from diverse backgrounds that are older than you and come from different cultures.

For you to fit into that successfully, you need to be ready to face the fact that the way people judge your 'fit' will be 70% your behaviour, and 30% your actual skill set. It will take quite a while before they can look past this and focus on your technical ability. It doesn't matter how well you code if you can't adapt to the environment in which you work.

How do you create a space that enables your Younglings to get to 'work readiness'?

It goes back to purpose, autonomy and mastery: I can give you purpose, and I can give you an environment where you have a purpose; I can give you supported and nurtured autonomy as well, but you have to find our own mastery.
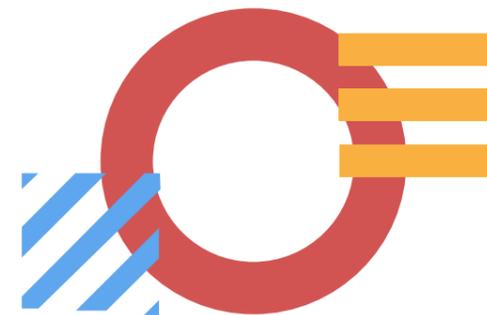
I have seen success in that every single time. And not 'normal' success; it's like exponential success, where somebody just flies because they finally found their niche, the bit that makes their head spin and their fingers tingle. It's difficult for someone to find that mastery for themselves at 18 or 19 years old, or even at 29. But once they find it, it's fireworks.

What lies in the future for initiatives like Younglings Africa?

There's money, and there are people out there who need that support – in other words, a purpose and an environment – but now it's companies that have to lead by example and find their own mastery in order to help others find theirs.

I'm currently only working with a small niche of people: Those who got 50% average and a Bachelor's pass. But there's the potential to start exploring other niches too. Even though I'm looking at an intake of 400 people in the next two years, that doesn't even begin to scrape the surface of how many people could easily be helped with just a little bit of structure and nurturing.

It does take a lot of money, but the government is already encouraging big corporates to spend on CSI. Companies actually just need to take that 'carrot', go out there, find niches that they can tap into, and empower our people.

MENTORING DIVERSE GROUPS BY BUILDING STRONG RELATIONSHIPS

Conversation with Thabang Magaola

Mentorship can be a really valuable tool in helping those who follow non-traditional paths into tech to be 'work-ready'. Working with diverse groups, however, can make this challenging, as everyone brings along unique struggles that need to be addressed. Thabang Magaola has had first-hand experience on both sides of mentoring, and shares his insights on what it takes to mentor 'work-readiness' effectively.

PREPARING FOR WORK

PREPARING FOR WORK

Coding is something that Thabang has always been passionate about. After enrolling at CodeX, he quickly discovered that learning new concepts came easily to him. However, it was the collaboration aspect offered by the bootcamp that really excited him.

Since he had always been interested in helping people, he was a big fan of the mentorship programme that CodeX was based on: "When I started at the bootcamp, I was told that we would be coding, but also spending some time learning how to communicate with people, collaborate, ask questions etc. They didn't want us to move on without understanding all of these concepts, and I really liked that".

Becoming a mentor

After graduating from the programme, Thabang was offered a developer position at a startup in Cape Town. Still wanting to give back to the community, he worked out an arrangement with his new CTO that allowed him to mentor at CodeX once a week. And after sitting with students for a few hours, as his mentors had done with him, he realised that he could make a real difference in their lives: "The more I gained mentoring experience, the more I felt the pleasure that comes with people saying 'I did this because of you'".

Because he was dealing with large numbers of students from different backgrounds, who had varying levels of experience and motivation, he found that he was continuously stretched, both intellectually and emotionally. Setting others up for success in a relatively short period of time proved to be both enormously challenging and exciting.

Having now worked as a mentor for about two years, Thabang has learned a lot of lessons about what good mentorship looks like. Here are some of his learnings.

Listening to understand the other

As a student, something that Thabang really valued about his mentors was their level-headedness: "They were always very calm – even when someone was all over the place. They would take the time to calm you down first, and then listen to what your problem was. Once they understood that, they'd go through everything with you step by step".

He personally found that this made him feel a lot more confident in reaching out when he had a problem or didn't understand something. "If I felt like the mentors were judging me and getting annoyed with me when I didn't know something, then I would've stayed away from them."

Now a mentor himself, Thabang recognises that many of the students who enrol at bootcamps face challenges that make it harder for them to feel comfortable enough to reach out. Something as seemingly simple as the language that a course is taught in can make new technical information exponentially more overwhelming.

Thabang sees that these factors often cause students to put up mental and emotional barriers that are hard for mentors to break-down.

> **"You need to know the level that they're at. Some come with experience and some don't. So, I think you need to be open and explain to them that you're there to help and then, hopefully, they will open up to you."**

Establishing trust through listening

In Thabang's experience, demonstrating a willingness to hear someone's concerns can establish a solid level of trust. If a student feels that they can trust their mentor, they will be more likely to share what is really bothering them, and not just what is difficult for them at surface level. This, in turn, helps in the long run: "You help people by motivating them to want to help themselves".

In addition, Thabang thinks that the more attention he pays to those around him, the more he can broaden his own perspective. He does this by always seeking answers: "I never like to get stuck on things, so I ask a lot of questions. I want to know everything about a situation before I continue".

Whenever he's presented with someone's problem, Thabang does the following before immediately responding:

- **Understand the problem:** First, he tries to understand the problem at hand. If he cannot establish this understanding for himself, he knows that he needs to take the time to listen.

- **Follow-up with students:** He asks the student to explain what is wrong and continues to check that they are on the same page. He never wants to mislead someone by not understanding them, and so he makes an effort to get every detail.

- **Reassure students**: Once this understanding has been established, Thabang makes an effort to reassure the student, by saying, "There is nothing that you can't tackle and I am here to help you with whatever struggles you face".

- **Set up for the first step:** Next, he encourages them to start by acknowledging that it can be hard to do, without giving away any answers. By making the effort to set the student up well, they feel confident enough to hit the first key on their keyboards themselves.

- **Assist continuously:** Finally, Thabang makes it clear that he is always available for his students to check-in with him, so that they continue to feel confident.

Being open to differences

Interacting with people who have various degrees of experience, both as a student and mentor, Thabang has recognised the challenges that this can bring: "Some people don't have confidence and some are very defensive with their code. This can be quite hard".

Since he knows that he will not always be on the same page as everyone that he meets, Thabang has started taking the time to understand people's backgrounds.

Forming solid connections through curiosity

He has found that being more curious and open has helped him form more solid connections and establish a level of mutual respect and trust. He practises this by simply talking to people, "I like to learn from others and I can only learn if I know their stories.

By asking, I can find things out and then I can help".

By doing this, he's seen that ideas become easier to share, feedback becomes easier to digest and the learning experience becomes that much more enriching for everyone involved.

In order to actually build strong connections, Thabang has found it helpful to consider the following:

- **Every person comes from a different background:** This means not everyone starts on the same footing or for the same reasons.

- **People may lack motivation:** This could be because of what they have experienced in different education systems, or what they have seen happen in their families. It is more likely to be a learned blockage than an inherent attribute of the person.

- **Frustrations can easily be transformed into opportunities to learn:** Thabang always reminds himself, "Remember, things only become a problem when you allow them to be one".

Being vulnerable is a strength

Thabang used to think of himself as an anti-social person who liked to find his own solutions, so he never felt like he needed to ask for help. However, when he started at CodeX and had to work with groups to figure out new concepts, he found that he couldn't just stick to himself. Putting himself out there was the quickest and most efficient way to make an impact.

This has only been amplified since he started mentoring. It is impossible for any one individual working with hundreds of different people with different problems to know everything. That's why Thabang knows that reaching out is often the only thing he can do: "Being a mentor has taught me that sometimes I just need help".

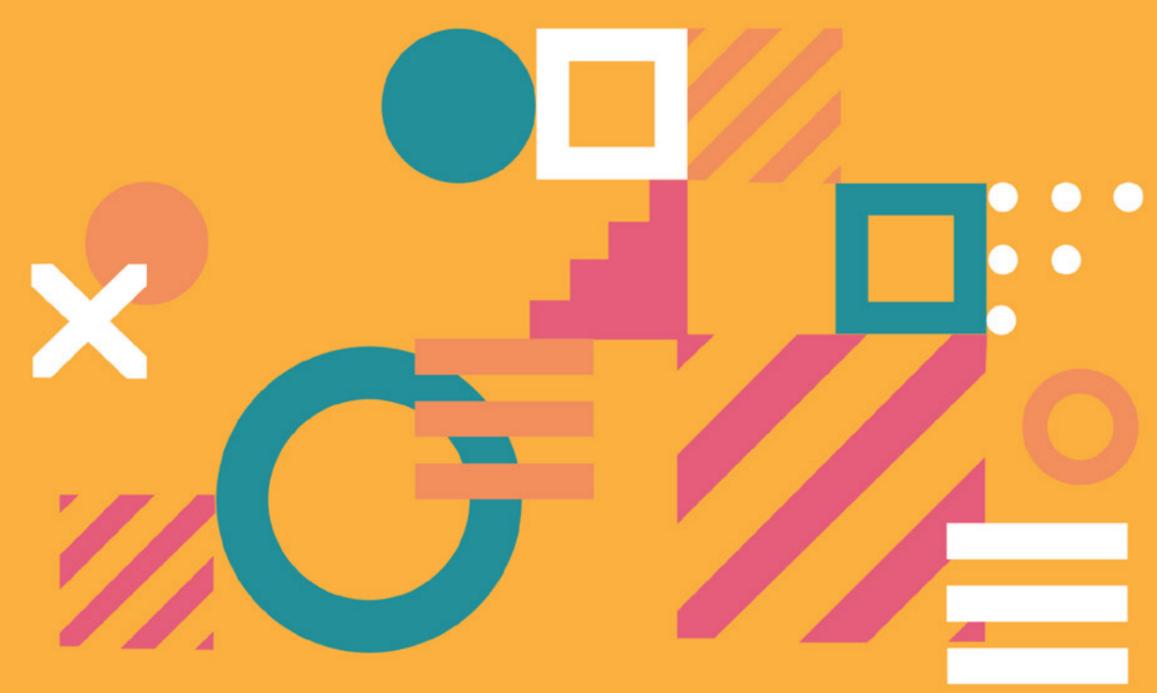**Being vulnerable helps overcome hurdles and build trust.**

By now, Thabang is not afraid to put his hand up and make it clear when he doesn't know what is going on. This has really helped him overcome a lot of the hurdles that are set up by innate feelings like pride and self-doubt. As he says, "Being a mentor is not just about helping someone. It can also help you as well".

Admitting his shortcomings as a leader also hasn't cost him any respect from his students. In fact, he's found that his students can better relate to him on a personal level and thus trust him enough to reach out when they need help. His students also increasingly want to contribute and offer support to others because they have experienced someone doing it for them. This is what brings Thabang the most joy: "When the students say, 'You are my role model,' I feel like I have really made an impact".

When Thabang is faced with helping a student with something that he doesn't know, he tries to put himself on the same level as the person that he is helping by doing the following:

- **Shares his story with them:** "These people (the mentors) have been in your situation, so they know what you're going through!". Providing this context helps to remove power dynamics from difficult situations so that everyone feels comfortable enough to talk at eye-level.

- **Shows students that he is excited to learn with them:** By acknowledging that he sometimes doesn't know how to answer a question, but is keen to work through it with the student, Thabang has seen that any judgement is quickly dissolved.

- **Swallows his pride:** By being curious about problems, instead of frustrated by them, Thabang has found that it is easier for him to address challenges. He has seen that admitting his deficiencies as a leader has opened him up to the possibility of being better while simultaneously encouraging others to do the same.

When asked about how he feels about making mentorship his full-time career, Thabang said that the decision has made him feel thoroughly fulfilled. He is excited to push himself to keep learning so that he can continue to make a difference to the people that he meets.

THRIVING AT WORK

Making the decision to prioritise diversity requires an inclusive environment where people from underserved communities are supported to reach their potential. In this theme, we'll discuss how companies can help a diverse group of people navigate their workplace.

WHAT AN INCLUSIVE WORKPLACE SHOULD ACTUALLY LOOK LIKE

Q & A with Zimkhita Buwa,
COO at Britehouse

Research shows that people who feel excluded at work are more likely to underperform or resign. This is particularly true in the tech space, where demand outstrips supply and software engineers have plenty of opportunities elsewhere. As a result, workplace inclusivity is not just essential in creating high-performance environments where people feel valued, but is also a question of retaining highly talented team members. Zimkhita Buwa lives and breathes inclusivity in the tech space, and has some practical tips on where to start and why.

Britehouse COO, Zimkhita Buwa, is deeply invested in, and passionate about, workplace inclusivity in the tech space. She has learned a lot through championing a host of tech inclusivity initiatives such as:

- SiliconCape's Students & Careers as well as Women portfolios,
- Techpearls, and
- Britehouse's own grad programme

She knows what it's like to feel excluded in the workplace, and these are some of the lessons she's gathered along the way.

Why should company members care about workplace inclusivity at all?

A lot of the younger talent that's coming into the market now aren't just looking for a high salary; they're looking for a space that can celebrate the value that they bring into the workplace. If you don't have an inclusive workplace, you might be able to hire those people, but **how are you going to retain them?**

The argument will still be: "We've always done things this way". But the new workforce that you're engaging with expects you to focus on purpose just as much as profit. They expect cross-functional teams, and they expect you to care about creating safe spaces for them.

The kind of creativity and ideas that you get from people of different backgrounds, cultures, as well as ways of thinking and working, is mind-blowing.

If you don't want to be disrupted by the market, if you want to innovate, and if you want to remain sustainable, you're silly to not consider an inclusive workplace.

With that said, what should an inclusive workplace actually look like?

A misconception, which I've seen in most of the environments that I've worked in, is that an inclusive workplace strategy is owned by the HR team. I truly believe that it needs to be **owned by leadership**, and then needs to be **executed by everyone in the organisation.**

THRIVING AT WORK

THRIVING AT WORK

You have to be intentional, and invest in it.

I also don't think it's something that lives in strategy maps and annual reports. I think it's **something that an employee feels when they walk into a work environment that accepts them for who they are.**

In the same way that leaders are obsessed with 'their bottom line' and profits, that's how obsessed we need to be about an inclusive culture.

> **Quite simply, from an employee's perspective, it's feeling like you're seen, you're heard, your voice matters, and that leadership is taking your thoughts seriously. I've felt the other side of that, and it's demoralising.**

For a company, it's firstly about recognising that you do – and always will – have different people from different cultures working with you. Then, I think it's about accepting and celebrating the different backgrounds that you have, and **creating a space that allows for people to confidently and comfortably own the fact that they're different.**

From your experience, what are the most effective ways of making workplaces more inclusive?

First-of-all, if it isn't top-of-mind all the time, and we aren't waking up feeling like "Let's actually do something about it!", then, to me, it will always be ineffective.

It also has to be authentic. It has to be something you implement all the time, across your entire company, not just in one or two areas. We don't profess to have all the answers, but we do know it is an essential conversation to be having in the boardroom, in order to set our company up for success.

Here are some examples:

Invest in leadership first

As I mentioned earlier, **inclusivity starts with leadership:** A solid investment into an organisation's leaders is crucial. It's not easy to deal with people from different backgrounds, and it's not something that's going to happen on its own.

Leadership has to be intentional, but it also has to be supported.

We've invested into a **tailored leadership training programme** for the past five years. Our CEO spent weeks crafting the programme according to what Britehouse needed, so that our leaders, team leads, and managers could implement its value into their immediate roles. They focused on making sure that the leaders knew:

- What our culture needs to be,
- What our values are,
- What the essence that makes us unique in the market is, and
- How we should make each other feel.

It's a hard one, but how I approach it is to ensure that **people see me living the values that I'm expecting them to embrace.** If they see me having an open door policy, it's not just 'something on a slide deck'. It's that I'm here chatting to everyone in the kitchen, walking the floors, and asking people how they are. I genuinely care about how their children are doing at school, and that makes a huge difference.

> **If you're not engaging, not caring, not living the value itself, then I don't know how you can expect other people to buy into it.**

Make people feel welcomed

On the very first day that people join, we clear our schedules for a **new joiner breakfast,** where the CEO and some of the other team members sit together with all the new joiners. We want them to feel welcome, and like they're part of the team – to the point where the most senior person in the organisation will make absolutely sure that he or she is there. It's so important that they get that sense from day one: "Wow. I feel important. I feel valued".

We're also trying an **onboarding app** so that, when new hires start, there are already things about Britehouse's culture and mission that they can read, and familiar faces that they can recognise. By the time we meet the new hires in person, they'll hopefully say "Oh, I saw your face on the app! You've done this, this, and this!".

These things, to me, already create this element of "We're excited you're coming!". After all, they chose to work for you, and you have to show them that you're excited that they did.

Make people feel understood

We run regular **online surveys** which people can answer anonymously, just to understand whether our employees really think and feel that we're an inclusive workplace. We ask things around:

• "Do you feel comfortable?"
• "Do you have a space where you can air your concerns?"
• "Do you feel like your input is being listened to and discussed?"
• "Are your day-to-day needs being met?"

These surveys are run by an external party to keep things very objective, and we only get the analysis of the results. But it helps to inform how we think about inclusivity in our workplace and whether we're doing it effectively enough.

Then we also run **one-on-one, in-person forums** where people can come and talk to committee representatives from within the company about how they feel within their teams, how they feel within the organisation, and what things we should be doing or looking at, for example. It's an avenue to share ideas or concerns, and to show that we care about the well-being of the people who work with us.

For very specific concerns that need more immediate attention, we've set up **internal committees** that are made up of people from Britehouse to deal with certain portfolios. For example, we have a remuneration committee that evaluates all the new hires and ensures that salaries are in line with the rest of the organisation. There's also a diversity and inclusion committee that looks at issues relating solely to diversity and inclusion across our organisation. This just creates more avenues that people can take, and gives them as many platforms as possible to raise certain issues.

Show people that their voices matter

Monthly 'Connect' sessions are another way that we drive inclusivity. The CEO, myself, and a group of about 30-40 people in the organisation – whoever wants to be part of the discussion – will come together for an open, honest conversation around what we should be doing more. We note the feedback, take it to board meetings, and discuss those ideas. Then, we close the feedback loop by communicating how we reached our decisions.

This **feedback loop is so important** because it's one thing to create an environment where people can share their thoughts, but **you also have to act.** You need to provide a platform where people aren't only sharing their thoughts, but where their input can truly be considered and included in the strategy – and be told why it isn't, when it isn't.

What can we do to address inclusivity, not only at an organisational level, but within the tech industry as a whole?

There's another conversation to be had around the tech industry. For instance, the number of women that are studying computer science, that then don't actually make it into the industry, is likely because they don't see the industry as a whole as being inclusive. But this is not necessarily only one or two companies. **And then there's diversity in religion, in age, and in gender as well.**

I think the industry is missing a critical ingredient by not having a governing body that actually looks at these things. Just like we have B-BBEE, we should have a diversity and inclusion index in South Africa that looks at what IT companies are doing to make sure that their work environments are inclusive. **We need advocacy and we need visibility in order to have some kind of accountability that we can hold companies to.**

More importantly though, we need to ask: "**Who is speaking on behalf of the younger generation?".** When people talk about the fourth Industrial Revolution, it's not lead by the people that need to be buying into this movement. It's the young innovators and women that should be doing that – not the older generation. So, why are we not putting them on the stage?

HOW THE RIGHT ENVIRONMENT CAN HELP YOU EXCEL

Conversation with Muchenja Namumba

The environment you work in can have an enormous impact on you and your career. Its routines, power dynamics and communication networks play a big role in determining how successful and happy you will be. Muchenja Namumba shares how her current company has set her up to succeed as a software developer. Here are her experiences and learnings.

After finishing school in Zambia, Muchenja decided that she wanted to study computer science at UCT. She secured a study visa and graduated a few years later. However, when it came time for her to do her Honours degree, she struggled to secure funding as a foreign student and had to start looking for a job.

Most companies, however, aren't willing to employ someone at that experience level with visa issues. Some of those that did employ Muchenja said that they could only pay her for half a day's work even though they wanted her to work full-time.

> **"The companies I worked for that year were hectic, 'prove yourself' kinds of environments. I never felt like I was good enough. I just had to work ten times harder than the people around me and, even when I did, it felt like I wasn't getting anywhere."**

When she had made enough money, she enrolled for an Honours degree in Information Systems at UCT. It was during this year that she first came into contact with the company where she is now working as a full-time software developer. "They didn't even blink an eye at the fact that I was a foreigner," Muchenja says. "They were so supportive with everything — they helped me secure my work permit, welcomed me into the team and set me up to really grow in this role. From the get-go, I knew that it was the best possible environment for me to do well."

Reflecting on her experience, Muchenja thinks that a great work environment, where employees can thrive, is made up of three key things:

- Supportive leadership,
- Belief in employees and their abilities, and
- Unlimited learning opportunities.

Here is how she feels her company has made use of these things to her up to win.

Supportive leadership systems

Relatable team leads

Having the right leader can be a game changer. When Muchenja joined her company, she found herself working under a young developer who had been catapulted into the role of lead developer in the early stages of the organisation. Having to teach himself things on the fly meant that he ended up being a great support for new developers who were struggling to find their feet.

"Honestly, he was the best person I could have had to help me settle in when I first joined the tech team. Since he had to teach himself so much, he understood what it was like to be uncertain, and this made approaching him with difficulties easy," Muchenja says.

> **Having a leader who is not afraid to talk about their experiences, or share their challenges, goes a long way towards helping people feel comfortable.**

Too often, the title of 'team lead' establishes this person as untouchable, and other team members feel too intimidated to reach out when they need help. Muchenja doesn't think this should be the case. "With this team lead, I never felt like I had to be perfect. I knew that he had struggled with many of the same things that I struggled with, and so, because of how honest he was, it became pretty instinctual to turn to him when anything hard came up."

Regular check-ins

Knowing that her company is keen to have its employees discuss their well-being and experiences with team leads every other week is something that Muchenja finds really encouraging. "I really like the fact that I regularly get the chance to sit down with my team lead to check in on where I am at and how I am doing," she says. "The radically candid feedback I regularly received helped me spot my weaknesses and areas needing improvement."

> **Because she has a safe space to share what's on her mind with someone that she trusts, Muchenja has found that she is no longer scared to voice her ideas and opinions.**

Now, when she thinks of something new or can see a path to take with a project, she knows that she can share what's on her mind and have it be taken seriously. This leaves her actively wanting to contribute to the team.

In addition to this, Muchenja really felt valued when the CEO included her on a big overseas project. "We had never done anything like this before and it was going to be a huge learning experience, not only for me but for everyone that was involved." Working on this project was an opportunity to work alongside the CEO and have check-ins with him. "I really got to see how much he cares about the people who work for his company and how much confidence he has in our various abilities. That's when I knew that I had made the right choice when I signed my contract here."

> **When a company takes the time to make sure that their employees are doing alright, both on a professional and a personal level, a safe environment is established.**

Believing in employees and their abilities

Providing opportunities that stretch

Early on in her new job, Muchenja found herself working on or even responsible for projects way before she felt like she was ready.

"My team lead would tell me, 'Hey, that new project that we just signed with that international client has been assigned to you. Congrats and I believe in you!'. Suddenly, I'd find myself in charge of work that had extremely high stakes and, more often than not, I would stress."

All the fears from Muchenja's previous experiences would come rushing back, and it was hard for her to feel like she was good enough to take on such important work. "When you

come from a place where people assign only the simplest tasks to you, it takes a lot of self-confidence to accept that you can actually take on something bigger."

As time went on, Muchenja came to realise how good this level of responsibility was for her own confidence and growth. "As I got more and more of these projects, I started to realise that giving this work to me meant that the company trusted me and valued my contributions. It took a while, but now, when something new lands in my lap, I have the confidence to grab it with both hands and run with it."

> **Making people feel like they are trusted, by handing them projects to run can boost confidence levels ten-fold, which means that they believe that they can reach new heights, both personally and on projects.**

Letting people own their work

Something that came with owning the work on big projects, was the opportunity to present them to clients. "The mindset where I work is that it's important to have the person who has done the work demo it to the people that they have done it for."

Instead of relying on account managers or other senior members of a company to be the sole point of contact with clients, letting the people who actually do the work interact with customers is an excellent way of establishing a sense of accountability and ownership amongst employees.

"It's also a great way to establish relationships, build trust and open up communication channels," Muchenja says.

Setting up unlimited learning opportunities

Providing access to resources

When it comes to the more technical side of the job, Muchenja has found that every resource she needs to upskill has been made easily accessible. "I arrived on my first day only knowing a couple of languages that I had worked with at varsity," Muchenja recalls.

"The tech stack my new team used was a lot broader, and I had no idea how to use most of it."

Knowing that she had come into the role as a graduate, with minimal experience, her team lead worked out a plan to help Muchenja quickly level up. He combined online courses and practical work to help her practise what she was learning. Importantly, he made sure that the practical work was relevant and actually addressed a need that the company had.

By assigning her a task that had an important outcome, but only an internal deadline, Muchenja could push herself without feeling too stressed. More importantly, it helped reminded her that the work she was doing was valuable.

> **Investing in tools and courses to upskill employees helps with continuous growth.**

Encouraging team interactions

"Something that really stands out for me, working at my company, is the opportunity to work closely with so many different members of my team," Muchenja says. "I am not kept in a box – I can reach out to anyone and pull anyone in. By doing this, I get to help others and learn a lot at the same time."

Muchenja particularly enjoys working with the younger developers who join the team: "I felt like I was so supported when I joined, and love being in a position now where I can help people who might feel like I used to – it's really rewarding."

When companies allow employees to collaborate with each other and share ideas, learning opportunities become unlimited. Not only do people get to learn new, practical skills, but they learn interpersonal skills, which are important for growth in other areas of professional and personal life.

"I feel really lucky to be where I am now," Muchenja says. "I have the opportunity to grow and learn new things every day. I feel valued and respected, and I know that I have the support, resources and opportunities I need to do the job, and do it well!"

For most people, mentoring an intern — on top of everything else that one has to do — is a daunting thought. It's a resource-intensive endeavour that takes time, energy, and patience, but Arneau makes it look easy.

He says that ineffective mentorship puts you at risk of losing high-quality developers: "Your hesitancy to really spend time with the guys working with you leaves them feeling isolated", If someone feels unsupported, they start questioning the value of their job. Arneau wants to prevent that thought from even crossing their mind. He wants his teams to say: "I'm learning even more than I did at school, college, or university. And yet, I'm being paid to do it".

Arneau has been a mentor for the past two years and has applied this thinking to the way he mentors interns. He doesn't think that an internship can be really valuable to an intern, a developer or a company if mentorship isn't the focus. Just "giving someone a fish" doesn't help anyone. "And the irony is," Arneau says, talking about internships, "often thinking 'Hey, this is throwaway time' means it will end up being throwaway time after all".

> **The results of his approach to internships speak for themselves: Every one of the interns he's mentored so far went on to become junior devs at OneDayOnly.**

USING MENTORSHIP TO POWER CONTINUOUS LEARNING

Conversation with Arneau van der Horst,
Mentor at OneDayOnly

Being a mentor, and simultaneously getting your own work done, can be really hard — but Arneau van der Horst from OneDayOnly considers mentoring to be his full-time job. His approach with interns is so useful that they consistently use his methods to implement continuous learning throughout their careers. Here's how you can do the same in your company.

Arneau has primarily mentored interns, but the number plate dangling from the back of his chair — which reads "The Mentor" — says a lot about the value his colleagues attribute to his role and success as a mentor in general. Aside from the specific rhythms he applies to his mentoring schedule, his philosophy can be boiled down into two main drives:

- Every time he works with someone is an opportunity to teach them something new, as well as an opportunity for him to learn something new.

- He treats an intern as if they've already been hired full-time, so he invests as much time into an intern as he would into any other employee.

His philosophy forms part of every step of a new team member's path: From the first interview he has with an intern, all the way through to their career at OneDayOnly.

THRIVING AT WORK

THRIVING AT WORK

Every single person is an opportunity to teach

Whether he's working with a colleague, a junior dev, an intern, or even someone who's just come in for an interview, Arneau treats each and every person as a chance to teach, and to learn.

With people he's interviewing for OneDayOnly's internship programme, he spends a portion of the chat having a feedback session on how the interview went. In other words, he's mentoring that person to be better at interviews wherever they go, no matter whether they will be hired or not. In Arneau's own words, "the worst thing is to tell someone after an interview 'Thanks for coming through, we'll be in touch' – and they never hear from you again".

Instead, immediately after the interview, Arneau shares his initial thoughts. He tells them what they did well, where they could improve, and talks about the technical assessment to figure out if "wrong answers" were just misunderstandings.

By spending that time on feedback after the interview, Arneau achieves two things:

Firstly, he sets someone up to succeed at future interviews – regardless of whether they get hired at OneDayOnly or not. "By extension," he says, "that means we're empowering guys for the entire industry". In his experience, people aren't taught to ask "Why didn't I stand out? What should I have done better?". He wants them to adopt that mentality no matter where they go in their career, because it's something everyone could do more of.

Secondly, he primes someone for the culture of mentorship he fosters at OneDayOnly, should they be offered the internship. This will help them onboard and level up a lot faster.

Treating an intern like a junior dev

Once an intern is hired, Arneau uses mentoring 'rhythms' that are a big part of OneDayOnly's general culture. He treats them like they've already been hired full-time, rather than "testing" them to see if they survive.

He sees his role as training them for the positions they're going to get:

This makes his job easier and their lives less stressful, because instead of an intern trying to prove themselves, they are working towards something as a team.

To create this relationship from day one, Arneau integrates his interns into company rhythms right away. He emphasises that he intentionally uses a mixture of group and individual dynamics to simultaneously develop team skills and a communication channel that's a "safe-space".

He wants his interns to be able to work in groups, but have a place where they can voice concerns they might not be comfortable sharing publicly. Some of the rhythms Arneau uses include:

As an individual:

- **Weekly catch-ups:** This is an informal chat to see how the intern is coping and if they're still enjoying their time. Arneau deliberately makes these a one-on-one meeting, and leverages that engagement towards creating a real relationship with the person he's mentoring. Arneau has found that problems get addressed much sooner, which means an intern is less likely to steam-train down the wrong path, and Arneau can help them change course early, and learn.

- **Quarterly personal development programme:** During an internship, Arneau sits down with each intern individually, checks the progress on the goals from the previous three months, and sets goals for the next quarter. They also discuss how OneDayOnly as a whole can help them achieve each of the goals they set. Each intern's goals becomes a team-effort, and not a 'solo mission'.

- **Daily one-hour learning session:** Interns are encouraged to dedicate at least an hour of their day to learning, whether it's a programming language, or a coding problem. The company even pays for online courses to help interns level-up quicker. Rather than do things for someone, Arneau would prefer they acquire what they need to do things on their own. This way, they're more likely to realise when they go wrong, and ask for specific advice when they do run into obstacles.

As a group:

- **Daily sit-downs:** All interns get together and discuss how they will achieve the things on their schedule that day. This allows Arneau to give initial guidance, and interns facing similar tasks can hear how someone else did it first. The extra time is spent going through code, looking at case studies of a specific problem, or discussing techniques.

- **Monthly secret lunch appointments:** These are monthly lunches, which are limited to his interns. They get together for lunch on a random day and chat about everything non-work related, with the intention of building camaraderie within the group.

- **Monthly team lunches:** These are integral at OneDayOnly, bringing the wider team together and giving interns the opportunity to form relationships with people in the company that they don't normally work with on a day-to-day basis. Arneau encourages this as a way for the interns to integrate more holistically into the company and its culture.

Making mentoring easier through daily practice

These rhythms not only integrate interns into company cadences, but also help track an intern's happiness and the value they're getting out of the internship. They are definitive check-in points that Arneau can use to make sure he's doing a good job.

Continuous learning is something he instils during the internship, and practises on a daily basis:

> **"If you treat everybody you work with as an opportunity to teach, an opportunity to share, collaborate, help out… it's invaluable. It develops an environment where everybody's helping everybody out. Nobody feels like they don't have assistance. Your barrier to entry is so much lower just because of that shift in mindset."**

The way Arneau gets his own work done as well is by giving interns tasks from his to-do list that he thinks they can learn from. This achieves a number of things:

- He gets his own work done,
- He helps someone learn through practical experience, and
- There's organic investment from both sides because it's a task that both parties care about.

The intern learns through practice, and Arneau can put sufficient time into both his own work and his mentoring, without making the trade-off. In his own words, **if he's sitting at his desk, then he's doing something wrong:** "It means that somebody's out there who might need my help, and who I'm not helping right now".

———

LOOKING AHEAD

The disconnect between people from underserved communities looking to get into a career in tech, and companies looking for people with software skills can only be addressed if the tech community works together. Part of working together will mean that we first have to acknowledge that the industry needs to change. This change will require us to re-evaluate how we work with training providers, our existing tech teams and the underserved community as they come into the tech industry. This will determine how we can support these people to succeed in their tech careers.

That is why we started OfferZen Foundation. In the short time that we've been working on this mission, we've come to understand that effectively helping people from underserved communities cannot be solved with a single solution.

As this report shows, there are already many people and organisations who are dedicating their time and resources to solve this challenge, yet these challenges still persist. Reflecting on the conversations with stakeholders in tech inclusion so far, there is a common feeling that we need to work together to bridge the divide in the tech space. People from underserved backgrounds looking to thrive in a tech career and companies looking for people with software skills are currently misaligned.

Keeping in mind that bridging this gap will require more than a single solution, we acknowledge that each stakeholder in the community needs to determine what knowledge they can share to move the conversation forward. For OfferZen Foundation, this report is that offering. It is our belief that there is a lot more work to be done to fully unpack what making the industry more diverse and inclusive means, and in what ways we're failing. Our hope is that this report provides an opportunity to continue this conversation with members of the tech community to explore what an inclusive eco-system could look like.

However, we are aware that this kind of change cannot happen with conversation alone. We are therefore looking to engage specifically around the actions we can take as Foundation, along with others in the ecosystem, to ensure that we can grow sustainably and inclusively.

We're inviting members of the community to join us as we unpack these issues. If you would like to join or share feedback, please sign up here.

ABOUT OFFERZEN

Malan Joubert
OfferZen Co-founder

We founded OfferZen because talent is universally distributed, but opportunity is not. There's an incredible amount of human potential that is not fully realised, because talented people don't always have access to great jobs or learning opportunities. At OfferZen, our mission is to close this gap by helping people unlock their potential. But, in order to do this, we realised that we need to better understand why the gap between talent and opportunity exists.

When my co-founder Philip and I were deciding how to approach this mission, we eventually chose the software space because we see it as fundamentally democratic. It's cheap, meritocratic and has the power to be the primary economic driver for a long time to come. And because software is levelling the playing field, we believe that helping people unlock their potential through better tech jobs, upskillment opportunities and community engagement has the potential to reset inequalities. But that's in an ideal world.

In reality, while helping thousands of software makers get access to jobs and learning opportunities, we've seen how many people just don't get a fair shot at these same opportunities, even if they have the right software skills. This means that a lot of human potential remains untapped, which ultimately stops us from achieving our core mission at OfferZen.

When it comes to unlocking potential, we've realised that it's a lot easier to do this in a diverse space where you feel included, respected and able to set yourself up for a thriv-
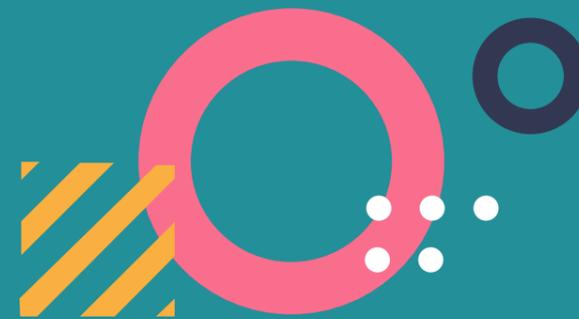
ing, fairly paid career. At the same time, as a company that employs developers, we've realised that we don't know a lot about diversity and inclusion ourselves. We know that there's a lot more we can do for inclusion and diversity in our own team, and we need to become better at it in our own capacity. Right now, we don't have good answers, or perhaps even the right questions. So, as a starting point, we've taken steps to put together this report. You could say that it's our attempt at learning about the right questions.

We took awhile to get here, because inclusion and diversity in the tech space is a complex challenge. It's easy to think that you understand it, when you actually don't at all. As an example, we initially thought that it was underpinned by a shortage of developer skills that could be addressed with more training opportunities. So, we reinvested the bulk of our revenue back into the tech community to fund meetups and workshops, sponsor swag for speakers, and help developers share their learning with the community.

But, when looking at our data and speaking to members of the community, we realised that training in tech skills is not the only, and perhaps not even the primary challenge. Over the past three years, we've seen how good developers, who know their stuff, enter the job market and still face several unfair barriers while preparing for work or moving through the hiring process. This reality means that we don't fully understand the problem and need to change our approach to give even more respect to its importance and difficulty.

Now that OfferZen is in its third year, we've also started to get a better understanding of the complexities involved in achieving our mission and we've realised that we can't tackle it alone. We need support and guidance from the community, which we want to kick-start by asking community members to help us understand the challenges in the inclusion and diversity space. It would be awesome if OfferZen Foundation acts as the rally point for everyone who cares about this challenge. So far, this report has already helped us grow our network and learn a lot as a team.

But, we are by no means finished with the work in this space. The ultimate mission of OfferZen Foundation is to solve the tech inclusion problem, which would mean that people who want to thrive in the South African tech industry, are actually thriving. By joining forces with other companies and community members, we hope that we can work together towards achieving this mission. Starting with this report.

ACKNOWLEDGEMENTS

This report would not be possible without the immense support of the OfferZen team.

In particular we'd like to thank the editors: Summer Smith, Jomiro Eming, Robyn Luyt, Candice Grobler and Anne Gonschorek at OfferZen. Special thanks to Megan Boshoff, our designer, for her incredible work designing this report.