madewithlove

Audit report Sample

We're the market leader for technical due diligence of SaaS companies in Belgium and the Netherlands, servicing clients all over the world.

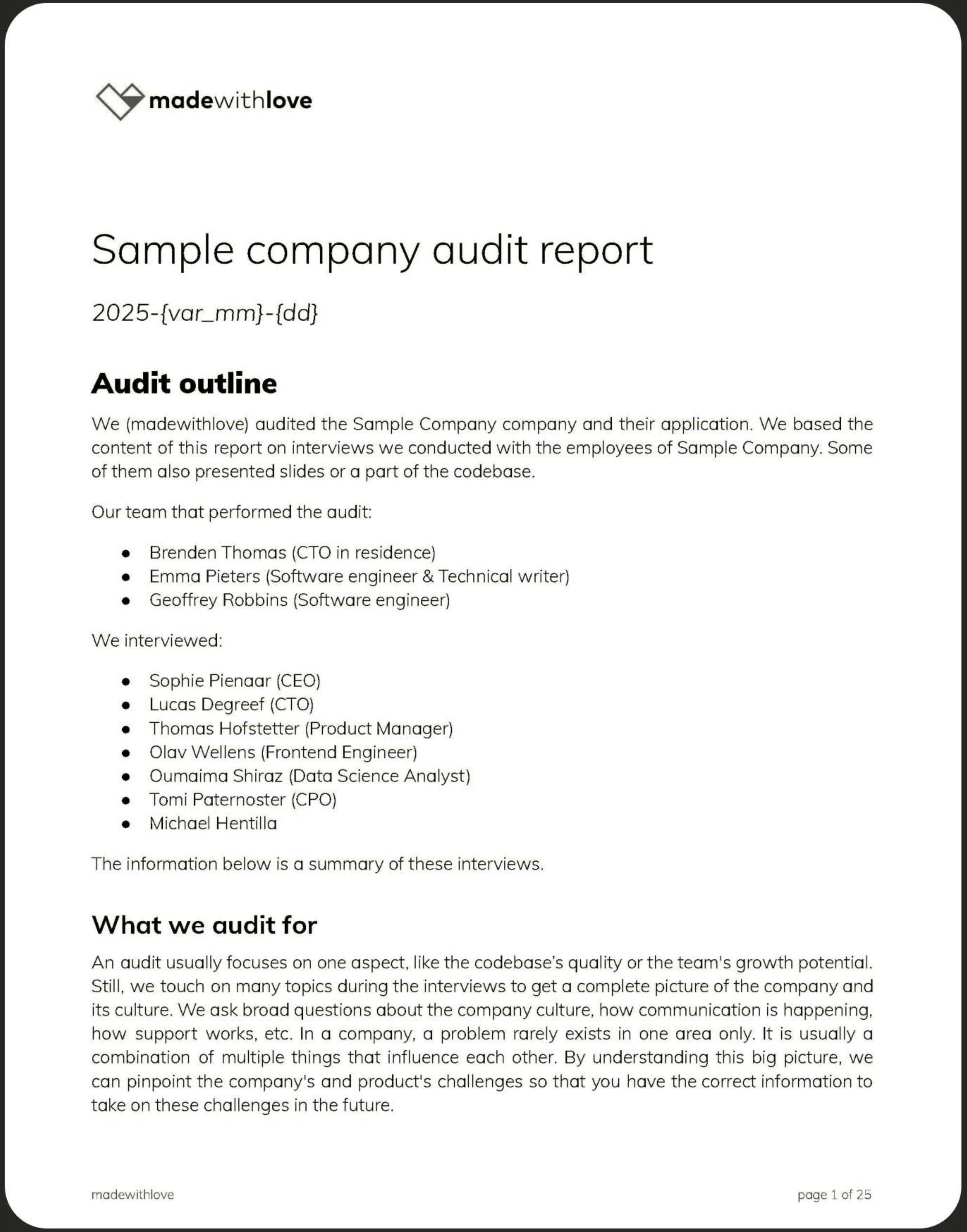
Investors appreciate our smooth process, sharp opinions, and top-notch reports.

Last updated: **April 1st, 2025**

IN THIS DOCUMENT, YOU'LL FIND:

- **An overview of our audit process —** what we audit for and why
- **The key questions we answer** as part of our audit
- **The lead time** of our audits
- **A bit more about us —** and how to get in touch if you'd like to gain insights into your engineering team or startup

Please note: some parts of this report have been blurred due to confidentiality reasons



madewithlove

Sample company audit report

2025-{var_mm}-{dd}

Audit outline

We (madewithlove) audited the Sample Company company and their application. We based the content of this report on interviews we conducted with the employees of Sample Company. Some of them also presented slides or a part of the codebase.

Our team that performed the audit:

- Brenden Thomas (CTO in residence)
- Emma Pieters (Software engineer & Technical writer)
- Geoffrey Robbins (Software engineer)

We interviewed:

- Sophie Pienaar (CEO)
- Lucas Degreef (CTO)
- Thomas Hofstetter (Product Manager)
- Olav Wellens (Frontend Engineer)
- Oumaima Shiraz (Data Science Analyst)
- Tomi Paternoster (CPO)
- Michael Hentilla

The information below is a summary of these interviews.

What we audit for

An audit usually focuses on one aspect, like the codebase's quality or the team's growth potential. Still, we touch on many topics during the interviews to get a complete picture of the company and its culture. We ask broad questions about the company culture, how communication is happening, how support works, etc. In a company, a problem rarely exists in one area only. It is usually a combination of multiple things that influence each other. By understanding this big picture, we can pinpoint the company's and product's challenges so that you have the correct information to take on these challenges in the future.

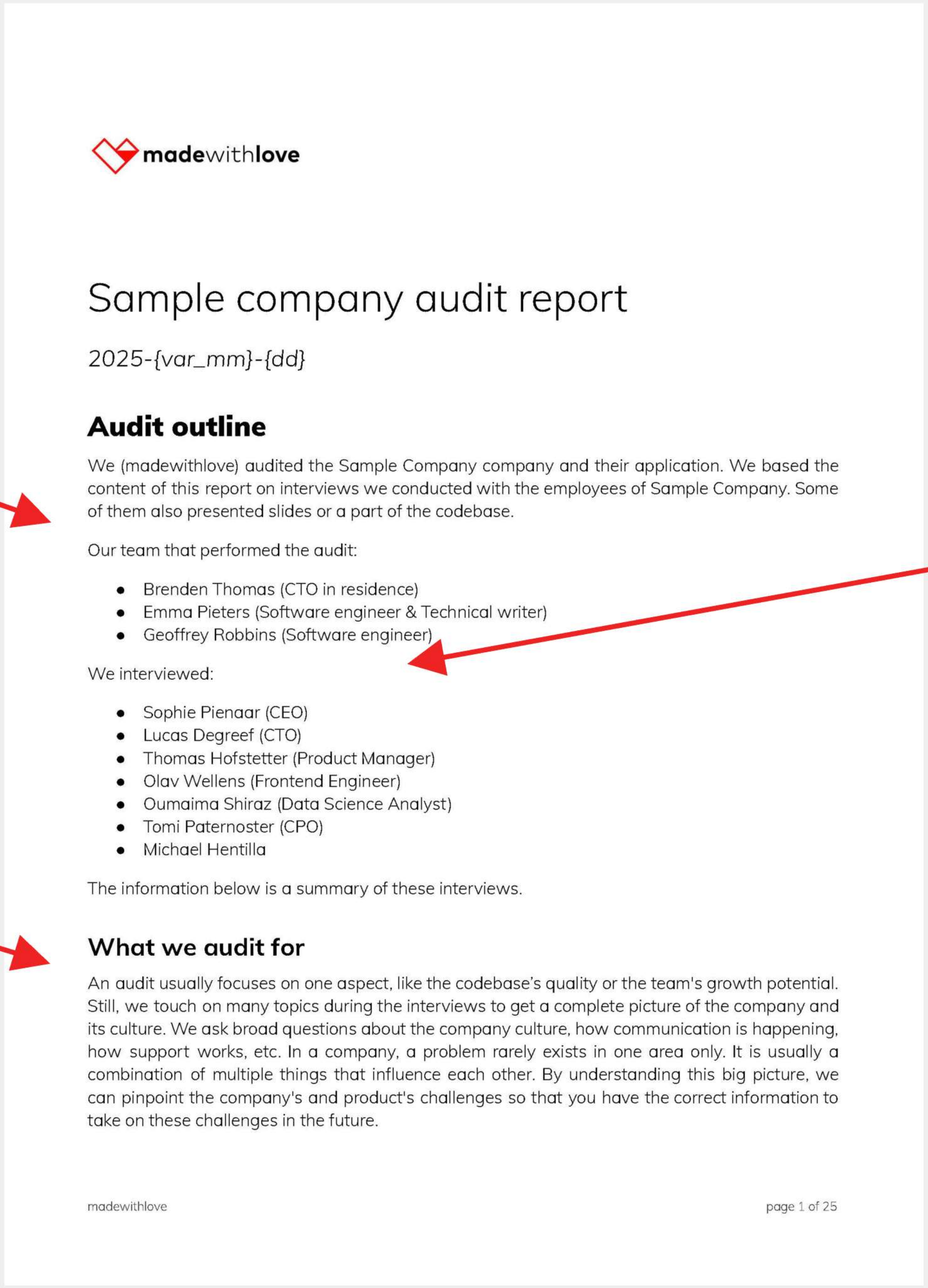
madewithlove page 1 of 25

Every audit is performed by **a team of world-class experts** who look at 5 pillars (critical areas)

Our report begins with a short reminder of them...

List of our experts who performed the audit

Short explanation of **our approach to auditing**



madewithlove

Sample company audit report

2025-{var_mm}-{dd}

Audit outline

We (madewithlove) audited the Sample Company company and their application. We based the content of this report on interviews we conducted with the employees of Sample Company. Some of them also presented slides or a part of the codebase.

Our team that performed the audit:

- Brenden Thomas (CTO in residence)
- Emma Pieters (Software engineer & Technical writer)
- Geoffrey Robbins (Software engineer)

We interviewed:

- Sophie Pienaar (CEO)
- Lucas Degreef (CTO)
- Thomas Hofstetter (Product Manager)
- Olav Wellens (Frontend Engineer)
- Oumaima Shiraz (Data Science Analyst)
- Tomi Paternoster (CPO)
- Michael Hentilla

The information below is a summary of these interviews.

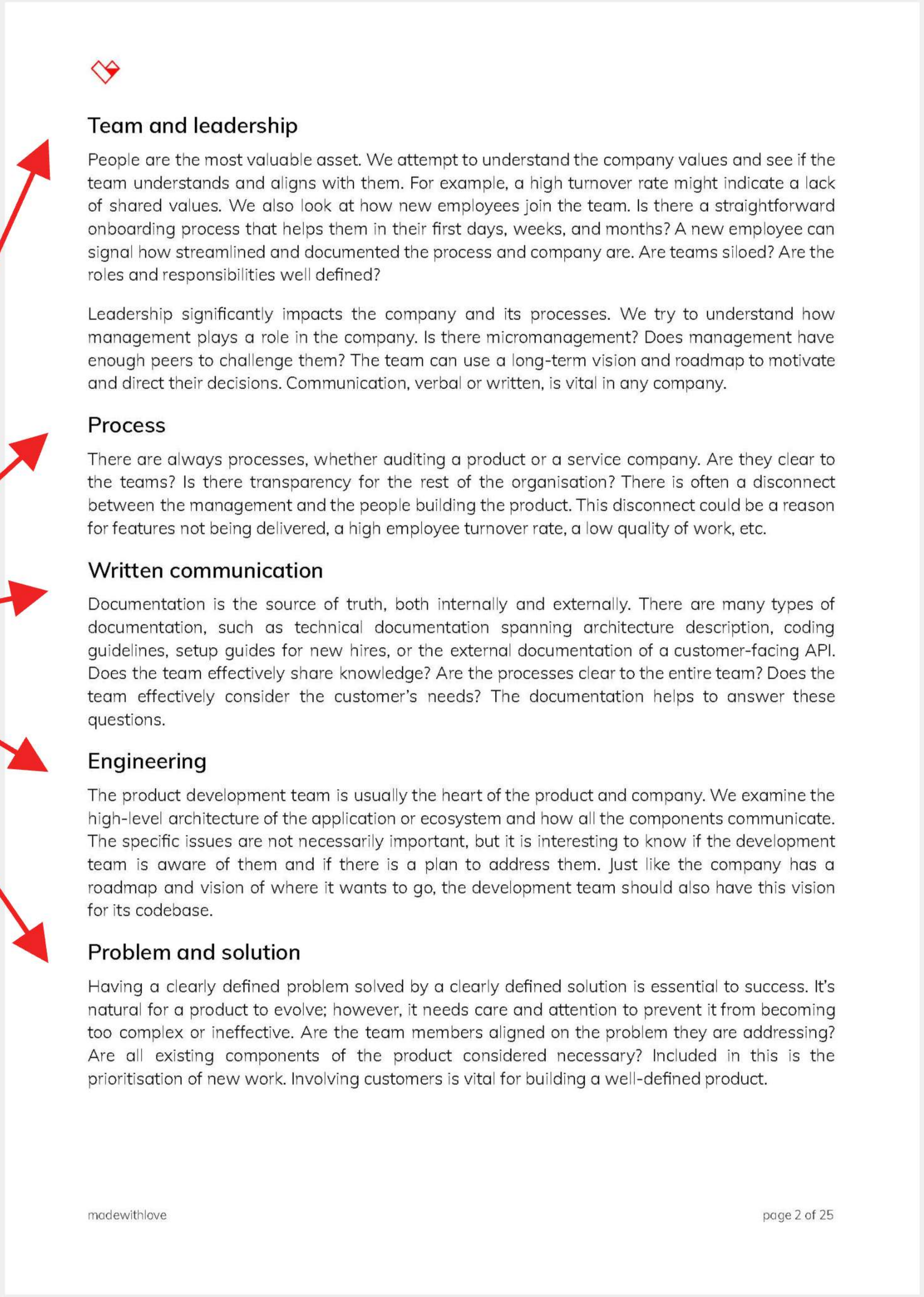
What we audit for

An audit usually focuses on one aspect, like the codebase's quality or the team's growth potential. Still, we touch on many topics during the interviews to get a complete picture of the company and its culture. We ask broad questions about the company culture, how communication is happening, how support works, etc. In a company, a problem rarely exists in one area only. It is usually a combination of multiple things that influence each other. By understanding this big picture, we can pinpoint the company's and product's challenges so that you have the correct information to take on these challenges in the future.

madewithlove page 1 of 25

List of people interviewed during the audit: employees are more likely to open up to us than to our clients **because their feedback remains anonymous.**

The introductory part of the report ends with an explanation of the **5 pillars that we analyse during audits**.



Team and leadership

People are the most valuable asset. We attempt to understand the company values and see if the team understands and aligns with them. For example, a high turnover rate might indicate a lack of shared values. We also look at how new employees join the team. Is there a straightforward onboarding process that helps them in their first days, weeks, and months? A new employee can signal how streamlined and documented the process and company are. Are teams siloed? Are the roles and responsibilities well defined?

Leadership significantly impacts the company and its processes. We try to understand how management plays a role in the company. Is there micromanagement? Does management have enough peers to challenge them? The team can use a long-term vision and roadmap to motivate and direct their decisions. Communication, verbal or written, is vital in any company.

Process

There are always processes, whether auditing a product or a service company. Are they clear to the teams? Is there transparency for the rest of the organisation? There is often a disconnect between the management and the people building the product. This disconnect could be a reason for features not being delivered, a high employee turnover rate, a low quality of work, etc.

Written communication

Documentation is the source of truth, both internally and externally. There are many types of documentation, such as technical documentation spanning architecture description, coding guidelines, setup guides for new hires, or the external documentation of a customer-facing API. Does the team effectively share knowledge? Are the processes clear to the entire team? Does the team effectively consider the customer's needs? The documentation helps to answer these questions.

Engineering

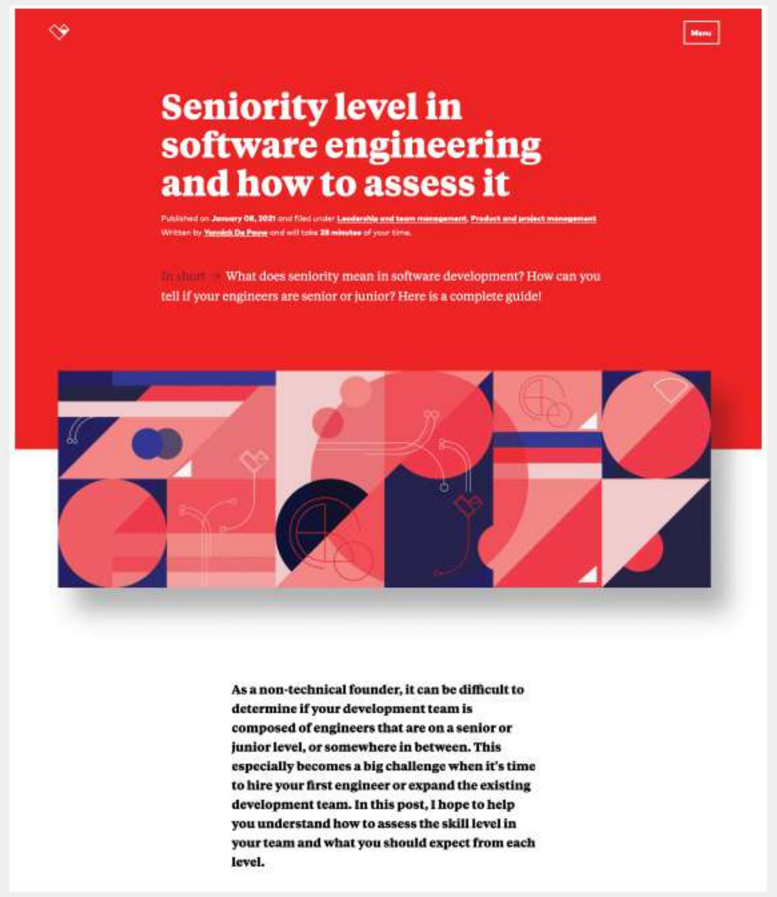
The product development team is usually the heart of the product and company. We examine the high-level architecture of the application or ecosystem and how all the components communicate. The specific issues are not necessarily important, but it is interesting to know if the development team is aware of them and if there is a plan to address them. Just like the company has a roadmap and vision of where it wants to go, the development team should also have this vision for its codebase.

Problem and solution

Having a clearly defined problem solved by a clearly defined solution is essential to success. It's natural for a product to evolve; however, it needs care and attention to prevent it from becoming too complex or ineffective. Are the team members aligned on the problem they are addressing? Are all existing components of the product considered necessary? Included in this is the prioritisation of new work. Involving customers is vital for building a well-defined product.

madewithlove page 2 of 25

Bonus: We've written an article about **assessing seniority of devs and engineers** (click image to open).



Menu

Seniority level in software engineering and how to assess it

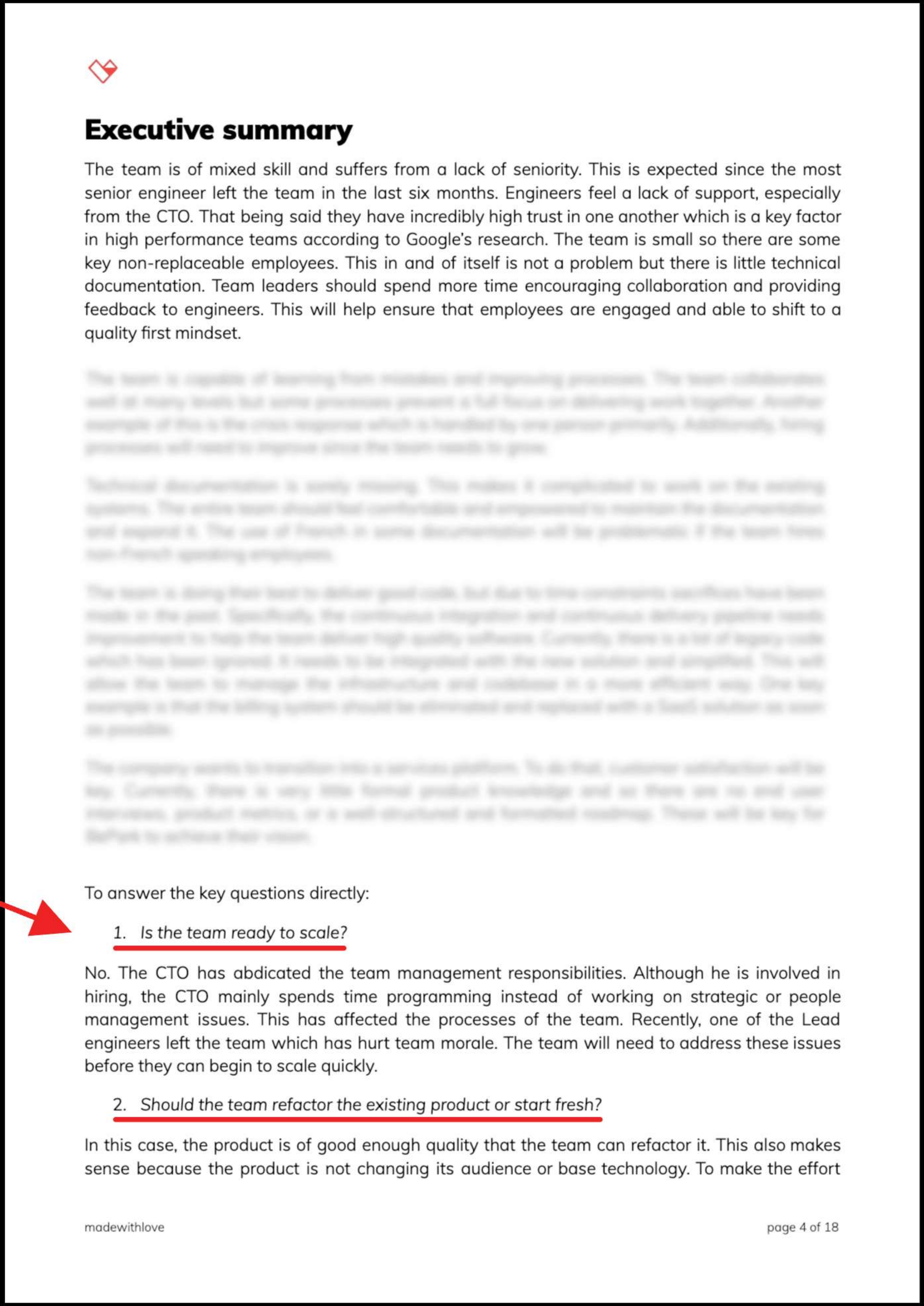
Published on January 08, 2021 and filed under Leadership and team management, Product and project management
Written by Yannick De Pauw and will take 28 minutes of your time.

In short → What does seniority mean in software development? How can you tell if your engineers are senior or junior? Here is a complete guide!

As a non-technical founder, it can be difficult to determine if your development team is composed of engineers that are on a senior or junior level, or somewhere in between. This especially becomes a big challenge when it's time to hire your first engineer or expand the existing development team. In this post, I hope to help you understand how to assess the skill level in your team and what you should expect from each level.

Before we present the detailed findings, we provide an **Executive Summary** to address the key questions our client had about their team or product

This summary provides Yes/No answers to the key questions our clients care about most. These questions are not fixed; **we define them individually with each client before starting the audit.**



Executive summary

The team is of mixed skill and suffers from a lack of seniority. This is expected since the most senior engineer left the team in the last six months. Engineers feel a lack of support, especially from the CTO. That being said they have incredibly high trust in one another which is a key factor in high performance teams according to Google's research. The team is small so there are some key non-replaceable employees. This in and of itself is not a problem but there is little technical documentation. Team leaders should spend more time encouraging collaboration and providing feedback to engineers. This will help ensure that employees are engaged and able to shift to a quality first mindset.

[illegible]

[illegible]

[illegible]

[illegible]

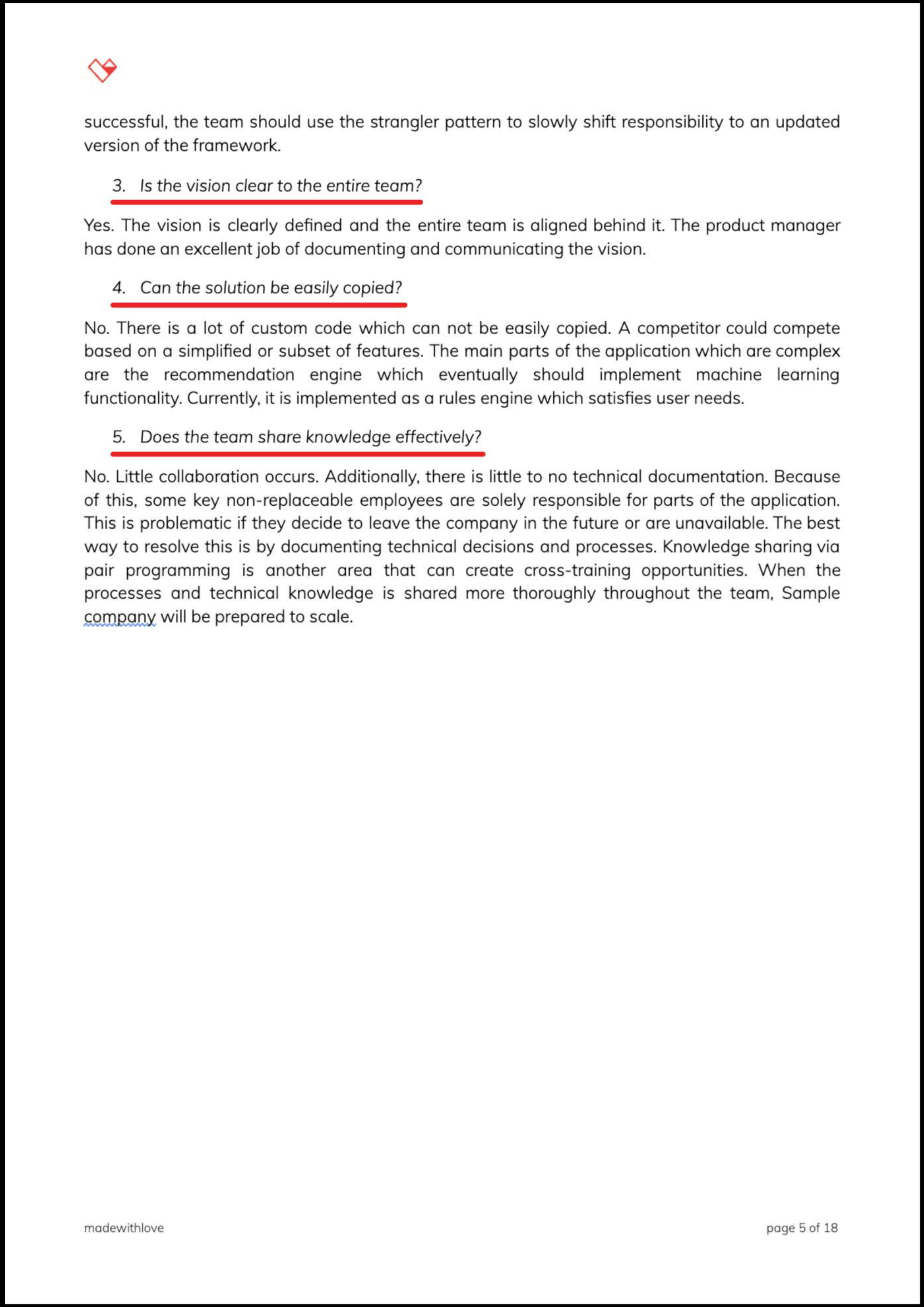
To answer the key questions directly:

1. *Is the team ready to scale?*

No. The CTO has abdicated the team management responsibilities. Although he is involved in hiring, the CTO mainly spends time programming instead of working on strategic or people management issues. This has affected the processes of the team. Recently, one of the Lead engineers left the team which has hurt team morale. The team will need to address these issues before they can begin to scale quickly.

2. *Should the team refactor the existing product or start fresh?*

In this case, the product is of good enough quality that the team can refactor it. This also makes sense because the product is not changing its audience or base technology. To make the effort



successful, the team should use the strangler pattern to slowly shift responsibility to an updated version of the framework.

3. *Is the vision clear to the entire team?*

Yes. The vision is clearly defined and the entire team is aligned behind it. The product manager has done an excellent job of documenting and communicating the vision.

4. *Can the solution be easily copied?*

No. There is a lot of custom code which can not be easily copied. A competitor could compete based on a simplified or subset of features. The main parts of the application which are complex are the recommendation engine which eventually should implement machine learning functionality. Currently, it is implemented as a rules engine which satisfies user needs.

5. *Does the team share knowledge effectively?*

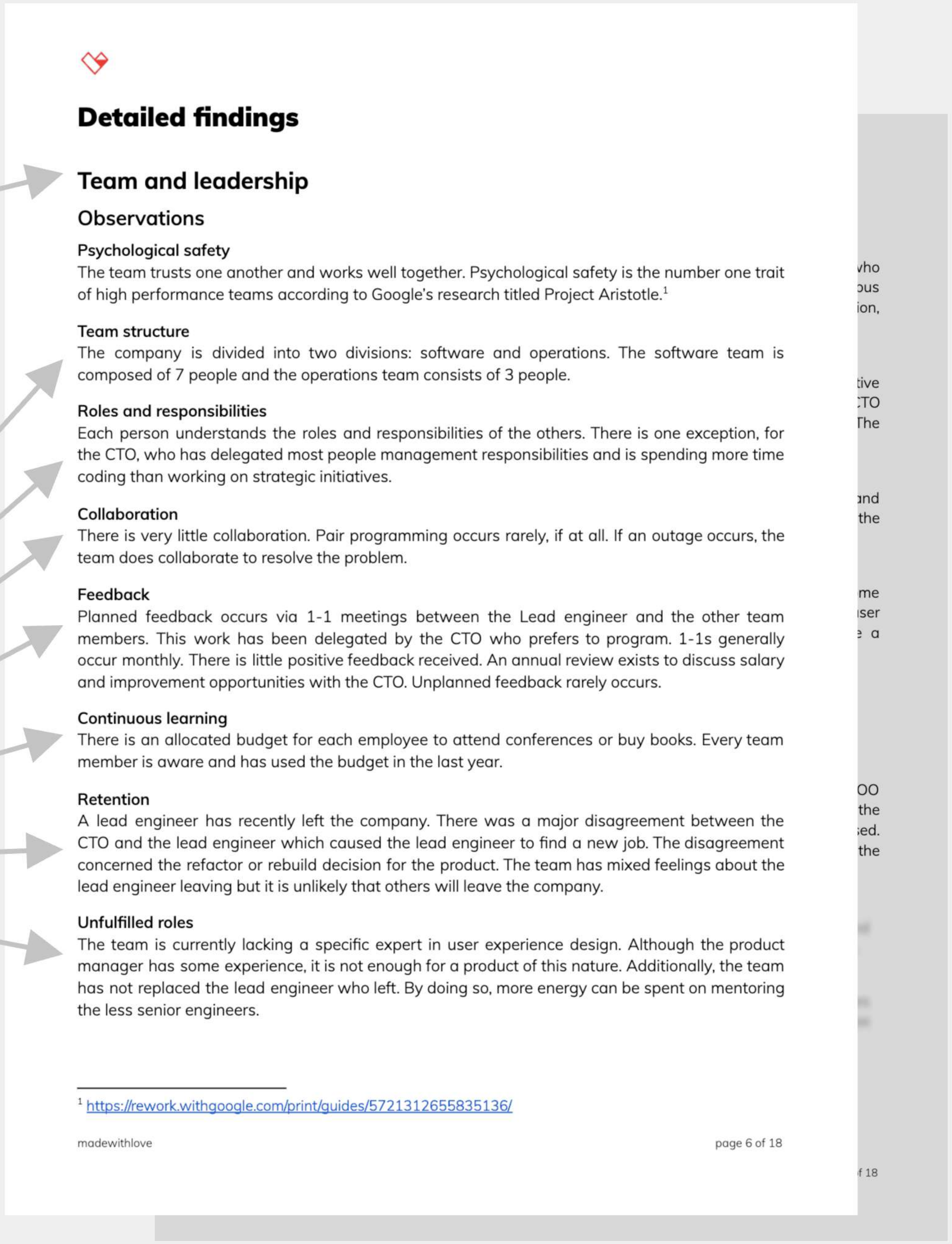
No. Little collaboration occurs. Additionally, there is little to no technical documentation. Because of this, some key non-replaceable employees are solely responsible for parts of the application. This is problematic if they decide to leave the company in the future or are unavailable. The best way to resolve this is by documenting technical decisions and processes. Knowledge sharing via pair programming is another area that can create cross-training opportunities. When the processes and technical knowledge is shared more thoroughly throughout the team, Sample company will be prepared to scale.

Finally, the main part of the report — **Observations** *(objective)* **and Concerns** *(subjective)* for each of the 5 pillars that we audit

All observations fact-checked by the audited company to ensure validity

We discuss every pillar separately, starting with
Pillar #1: Team and leadership.

We break down each area into smaller observations. **They provide a clear picture of the processes and relationships in the company, which, very often, different team members describe differently.** And that's how we can find out if there are communication issues in the team.



Detailed findings

Team and leadership

Observations

Psychological safety
The team trusts one another and works well together. Psychological safety is the number one trait of high performance teams according to Google's research titled Project Aristotle.[1]

Team structure
The company is divided into two divisions: software and operations. The software team is composed of 7 people and the operations team consists of 3 people.

Roles and responsibilities
Each person understands the roles and responsibilities of the others. There is one exception, for the CTO, who has delegated most people management responsibilities and is spending more time coding than working on strategic initiatives.

Collaboration
There is very little collaboration. Pair programming occurs rarely, if at all. If an outage occurs, the team does collaborate to resolve the problem.

Feedback
Planned feedback occurs via 1-1 meetings between the Lead engineer and the other team members. This work has been delegated by the CTO who prefers to program. 1-1s generally occur monthly. There is little positive feedback received. An annual review exists to discuss salary and improvement opportunities with the CTO. Unplanned feedback rarely occurs.

Continuous learning
There is an allocated budget for each employee to attend conferences or buy books. Every team member is aware and has used the budget in the last year.

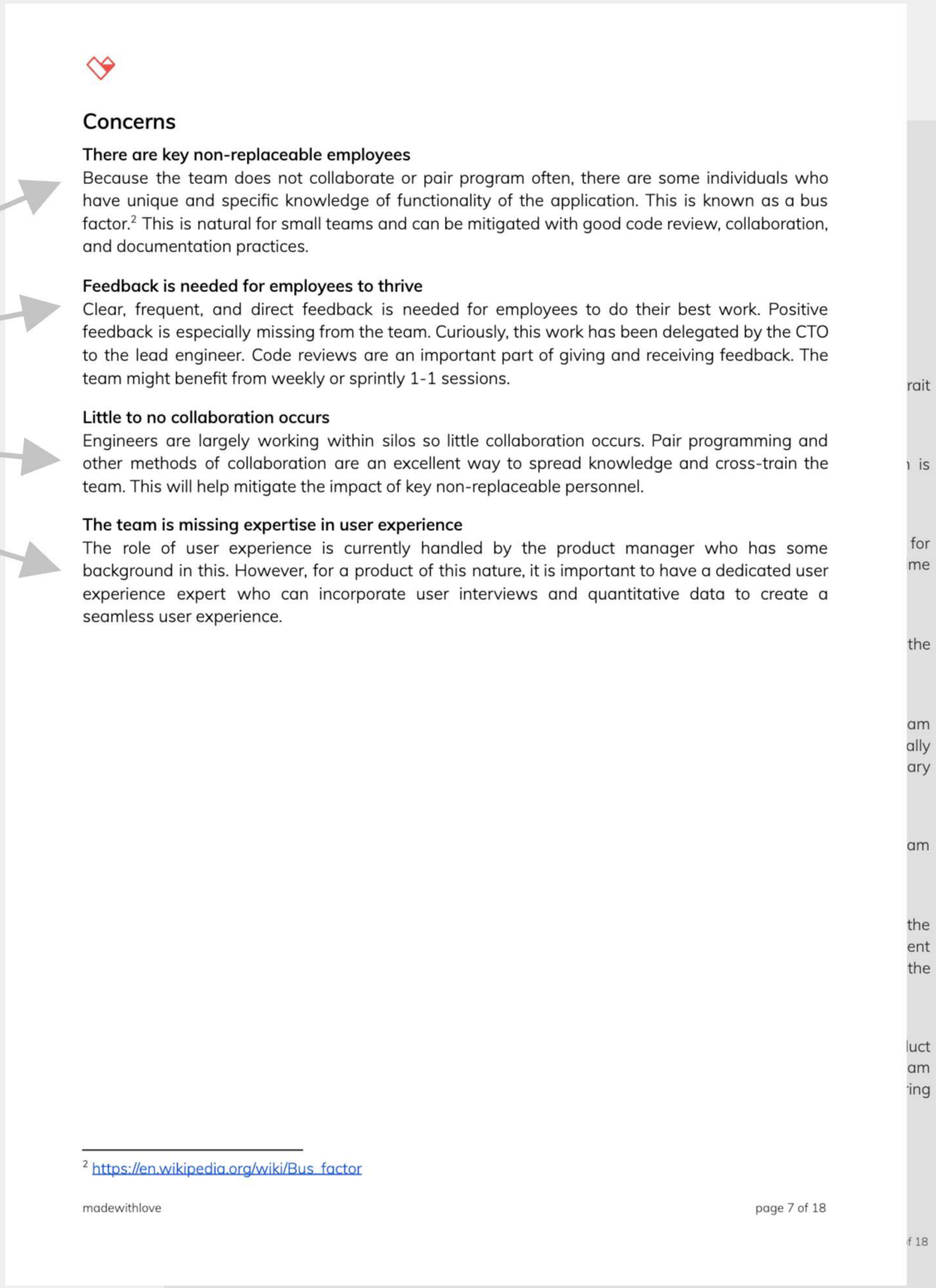
Retention
A lead engineer has recently left the company. There was a major disagreement between the CTO and the lead engineer which caused the lead engineer to find a new job. The disagreement concerned the refactor or rebuild decision for the product. The team has mixed feelings about the lead engineer leaving but it is unlikely that others will leave the company.

Unfulfilled roles
The team is currently lacking a specific expert in user experience design. Although the product manager has some experience, it is not enough for a product of this nature. Additionally, the team has not replaced the lead engineer who left. By doing so, more energy can be spent on mentoring the less senior engineers.

[1] https://rework.withgoogle.com/print/guides/5721312655835136/

madewithlove page 6 of 18

Based on the Observations, we describe our Concerns — things that sometimes might not even seem that important, **but which are likely to escalate in the (near) future**.



Concerns

There are key non-replaceable employees
Because the team does not collaborate or pair program often, there are some individuals who have unique and specific knowledge of functionality of the application. This is known as a bus factor.[2] This is natural for small teams and can be mitigated with good code review, collaboration, and documentation practices.

Feedback is needed for employees to thrive
Clear, frequent, and direct feedback is needed for employees to do their best work. Positive feedback is especially missing from the team. Curiously, this work has been delegated by the CTO to the lead engineer. Code reviews are an important part of giving and receiving feedback. The team might benefit from weekly or sprintly 1-1 sessions.

Little to no collaboration occurs
Engineers are largely working within silos so little collaboration occurs. Pair programming and other methods of collaboration are an excellent way to spread knowledge and cross-train the team. This will help mitigate the impact of key non-replaceable personnel.

The team is missing expertise in user experience
The role of user experience is currently handled by the product manager who has some background in this. However, for a product of this nature, it is important to have a dedicated user experience expert who can incorporate user interviews and quantitative data to create a seamless user experience.

[2] https://en.wikipedia.org/wiki/Bus_factor

madewithlove page 7 of 18

Menu

Why tech startups fail - based on auditing 50+ SaaS startups

Published on January 18, 2021 and filed under This is madewithlove
Written by Peter Packet and will take 17 minutes of your time.

In short I'm sure you've heard the statistic that 9 out of 10 startups fail. How can you win against such odds? Is there a secret startup handbook to read or an investment due diligence checklist to follow?

I'm sure you've heard the statistic that 9 out of 10 startups fail. How can you win against such odds? Is there a secret startup handbook to read or an investment due diligence checklist to follow? Madewithlove has audited over 50 companies, usually as part of an investment round. Sometimes CEOs sense something is wrong and want an honest opinion.

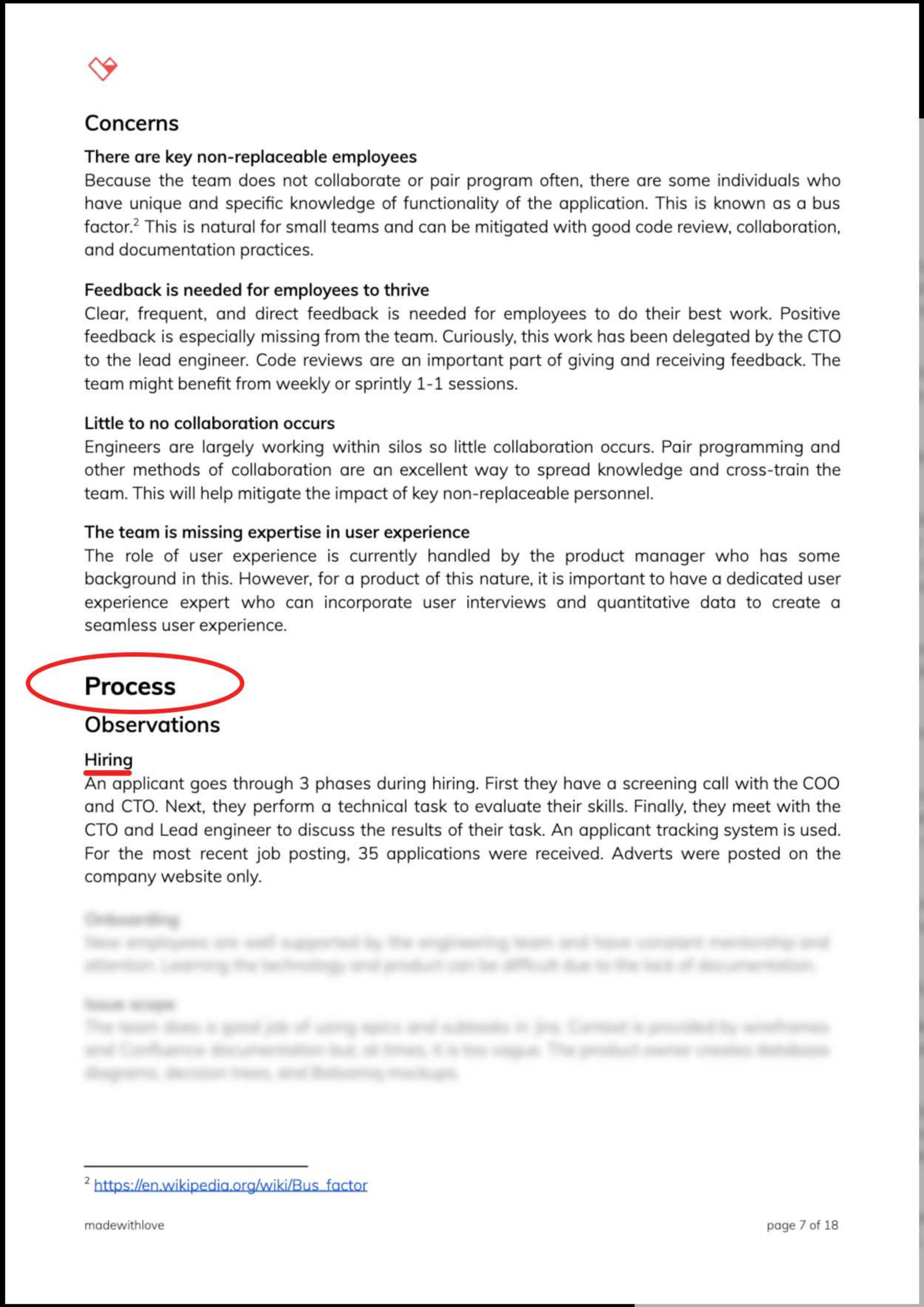
Bonus: We've written an article about **why startups fail —** based on auditing 50+ companies (click image to open).

And then we go on to describe **the other 4 pillars.**

We can't reveal everything, but we'll show you just enough so that you can see how much work and expertise goes into the report

Pillar #2: Process

An example of the Observations we made about this client's hiring process, as well as our Concerns about their deployment process.



Concerns

There are key non-replaceable employees
Because the team does not collaborate or pair program often, there are some individuals who have unique and specific knowledge of functionality of the application. This is known as a bus factor.[2] This is natural for small teams and can be mitigated with good code review, collaboration, and documentation practices.

Feedback is needed for employees to thrive
Clear, frequent, and direct feedback is needed for employees to do their best work. Positive feedback is especially missing from the team. Curiously, this work has been delegated by the CTO to the lead engineer. Code reviews are an important part of giving and receiving feedback. The team might benefit from weekly or sprintly 1-1 sessions.

Little to no collaboration occurs
Engineers are largely working within silos so little collaboration occurs. Pair programming and other methods of collaboration are an excellent way to spread knowledge and cross-train the team. This will help mitigate the impact of key non-replaceable personnel.

The team is missing expertise in user experience
The role of user experience is currently handled by the product manager who has some background in this. However, for a product of this nature, it is important to have a dedicated user experience expert who can incorporate user interviews and quantitative data to create a seamless user experience.

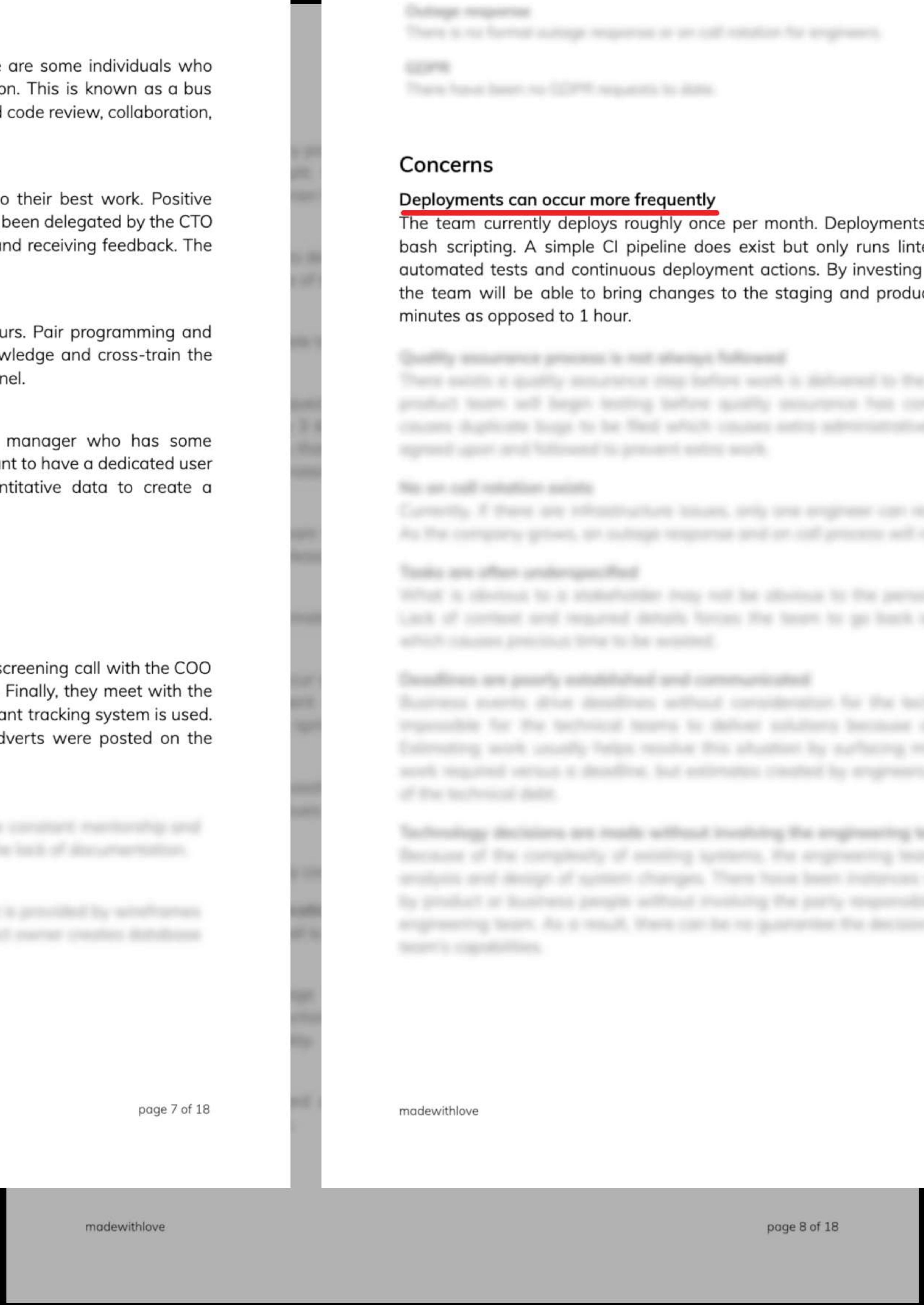
Process

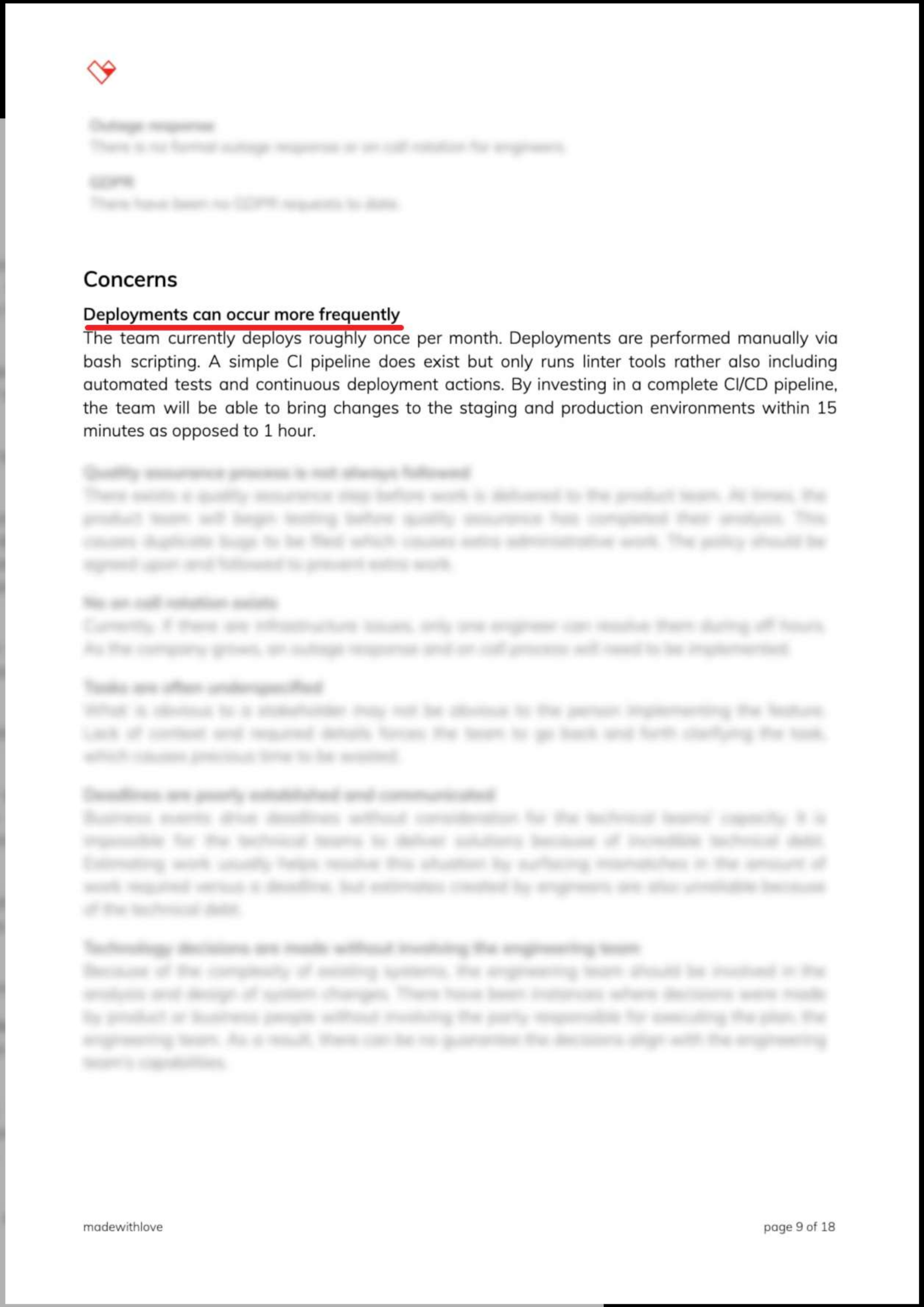
Observations

Hiring
An applicant goes through 3 phases during hiring. First they have a screening call with the COO and CTO. Next, they perform a technical task to evaluate their skills. Finally, they meet with the CTO and Lead engineer to discuss the results of their task. An applicant tracking system is used. For the most recent job posting, 35 applications were received. Adverts were posted on the company website only.

[illegible]

[2] https://en.wikipedia.org/wiki/Bus_factor





[illegible]

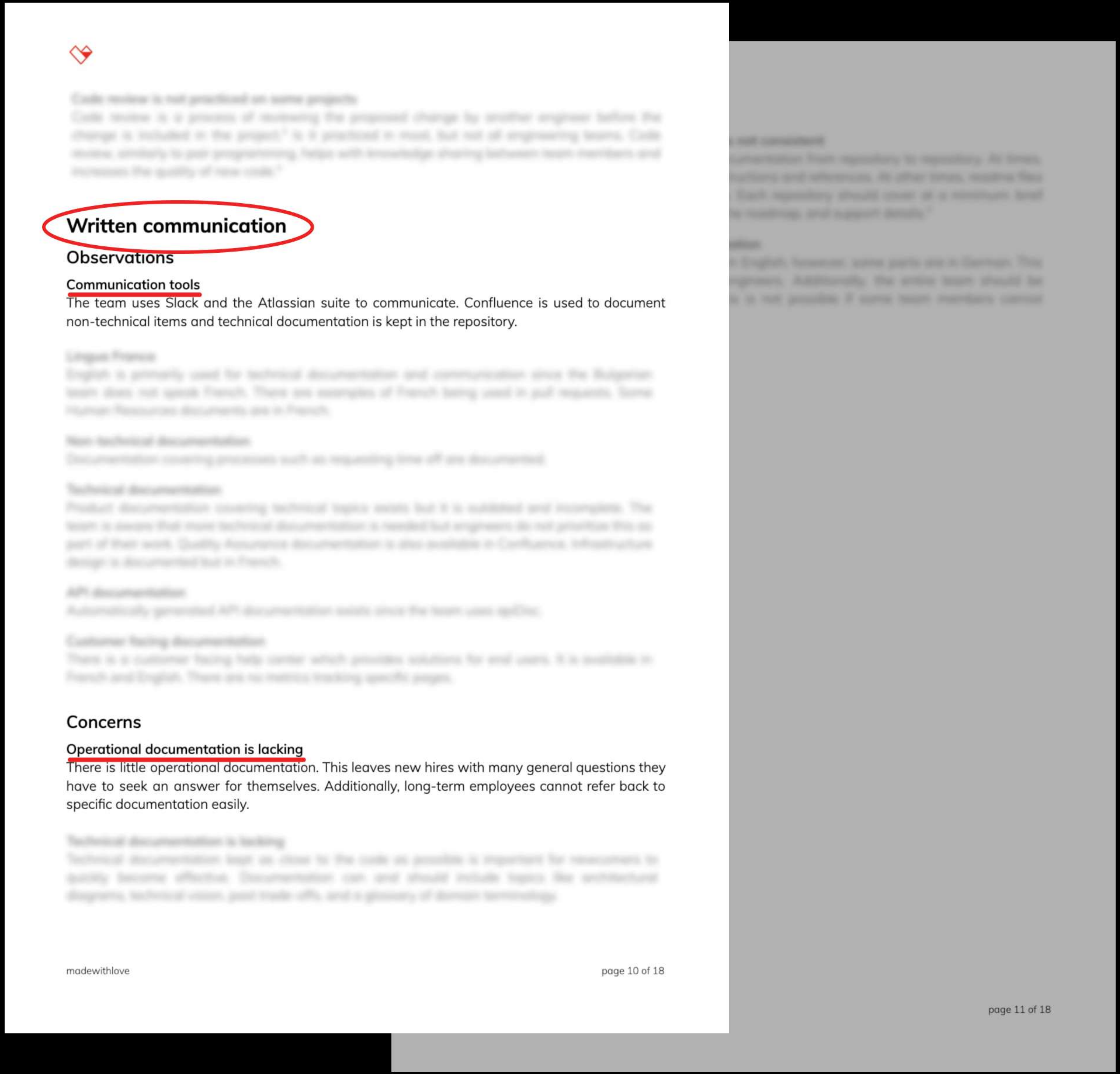
Concerns

Deployments can occur more frequently
The team currently deploys roughly once per month. Deployments are performed manually via bash scripting. A simple CI pipeline does exist but only runs linter tools rather also including automated tests and continuous deployment actions. By investing in a complete CI/CD pipeline, the team will be able to bring changes to the staging and production environments within 15 minutes as opposed to 1 hour.

[illegible]

Pillar #3: Written communication

We made Observations about the way this team communicates (in writing) and expressed Concerns about the lack of documentation which new hires could learn from (or long-term employees could refer to).



Code review is not practiced on some projects

[illegible]

Written communication

Observations

Communication tools

The team uses Slack and the Atlassian suite to communicate. Confluence is used to document non-technical items and technical documentation is kept in the repository.

Lingua Franca

[illegible]

Non-technical documentation

[illegible]

Technical documentation

[illegible]

API documentation

[illegible]

Customer facing documentation

[illegible]

Concerns

Operational documentation is lacking

There is little operational documentation. This leaves new hires with many general questions they have to seek an answer for themselves. Additionally, long-term employees cannot refer back to specific documentation easily.

Technical documentation is lacking

[illegible]

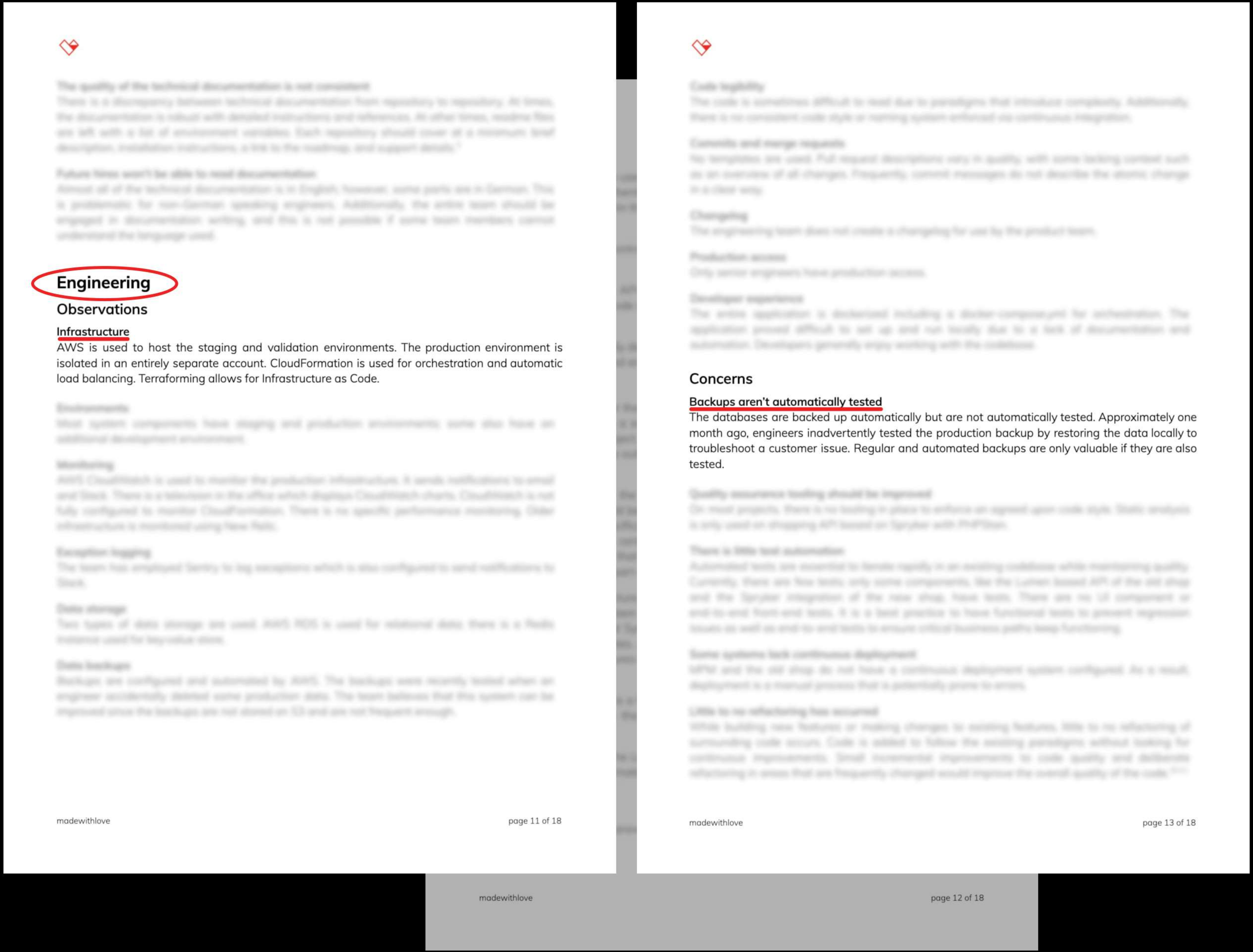
madewithlove page 10 of 18

[illegible]

page 11 of 18

Pillar #4: Engineering

We've saved (or managed) dozens of tech companies and we know which infrastructure, codebase, and lack of best practices will eventually lead to a failure of a business or product.



Engineering

Observations

Infrastructure
AWS is used to host the staging and validation environments. The production environment is isolated in an entirely separate account. CloudFormation is used for orchestration and automatic load balancing. Terraforming allows for Infrastructure as Code.

madewithlove page 11 of 18

madewithlove page 12 of 18

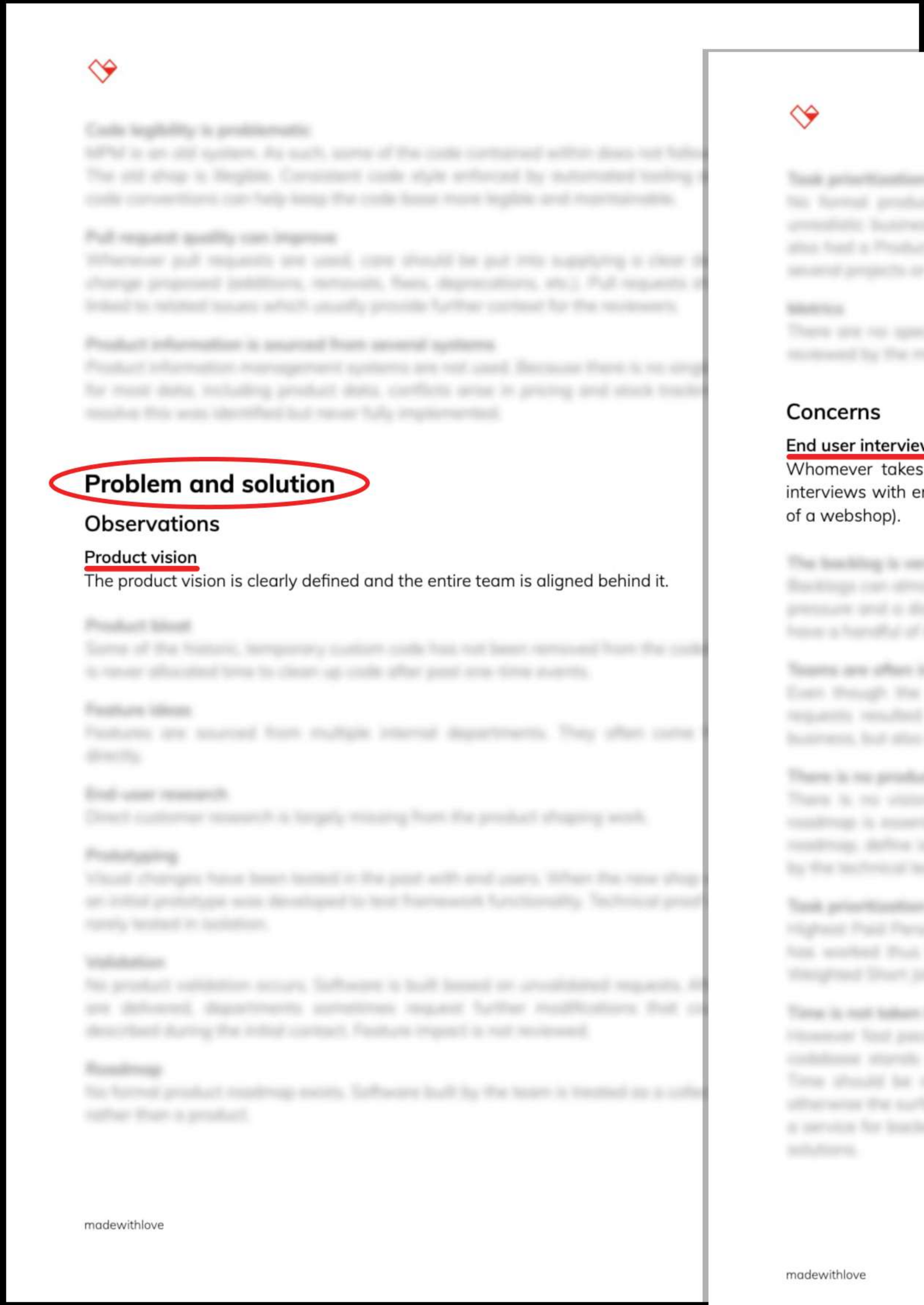
Concerns

Backups aren't automatically tested
The databases are backed up automatically but are not automatically tested. Approximately one month ago, engineers inadvertently tested the production backup by restoring the data locally to troubleshoot a customer issue. Regular and automated backups are only valuable if they are also tested.

madewithlove page 13 of 18

Pillar #5: Problem & Solution

Finally, we check if the product vision is well defined, if the end-user's needs are understood, and, most importantly: **if the product is ready to scale or will it eventually fail?**



[illegible]

Problem and solution

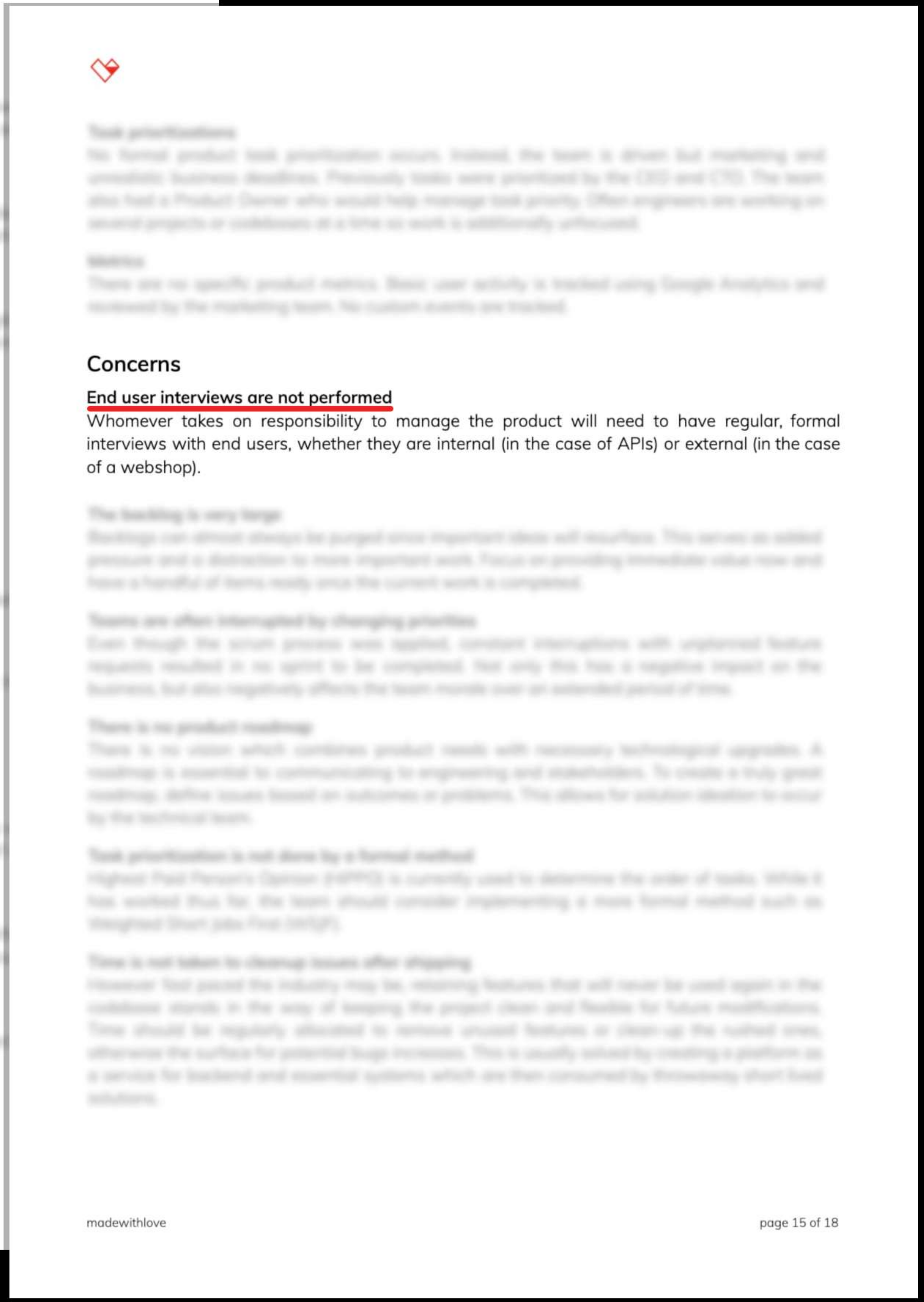
Observations

Product vision

The product vision is clearly defined and the entire team is aligned behind it.

[illegible]

madewithlove



[illegible]

Concerns

End user interviews are not performed

Whomever takes on responsibility to manage the product will need to have regular, formal interviews with end users, whether they are internal (in the case of APIs) or external (in the case of a webshop).

[illegible]

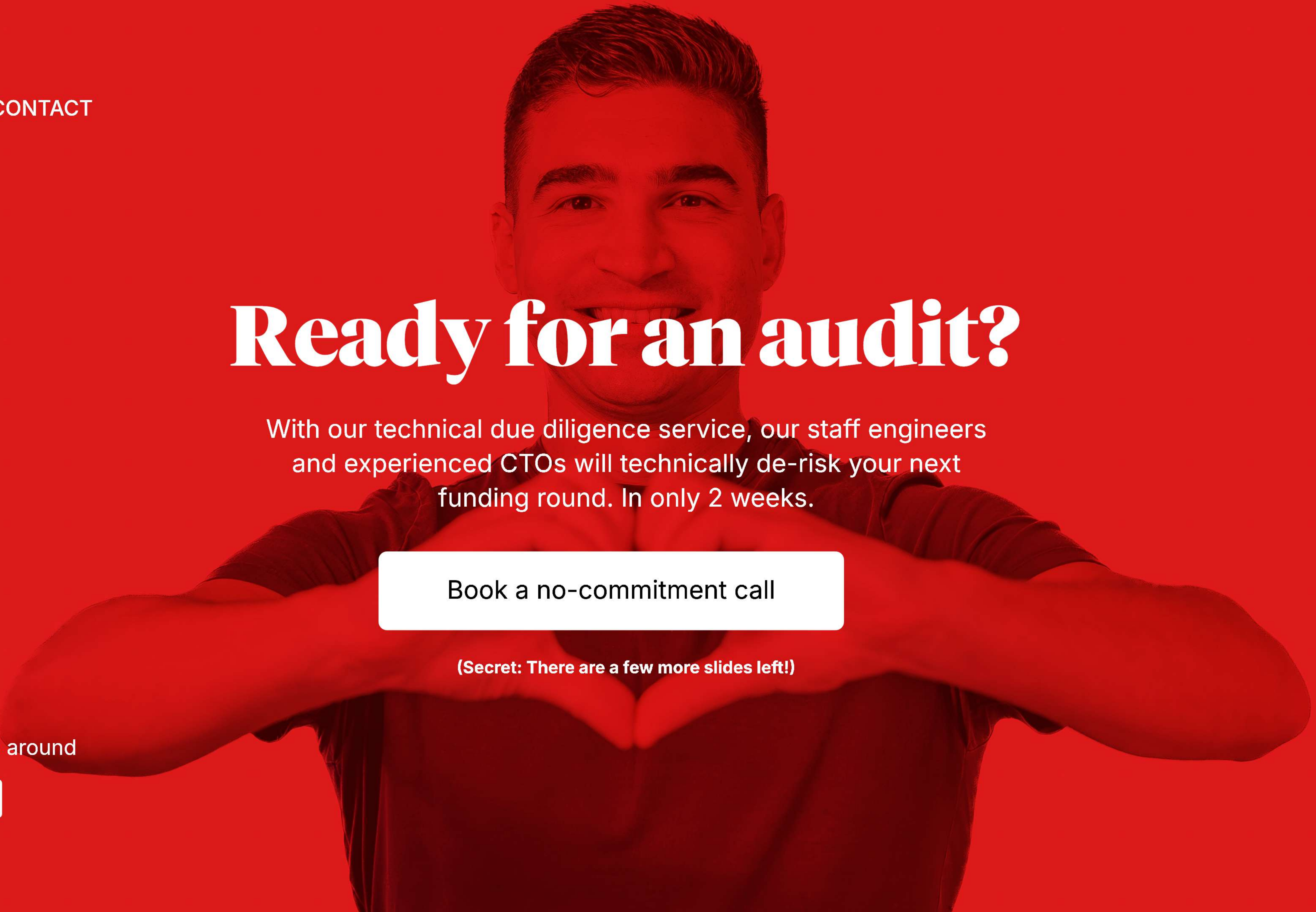
madewithlove page 15 of 18

Lead time of our audits

We don't expect a thorough preparation from your side.
How much time do we need to perform our audits? Our ambition is to finish it within two weeks, but it also depends on your availability. All our meetings happen via video call. This is what we do:

- Validate key questions and focus points with investor *(*)*
- Have a baseline interview (2 hours)
- Conduct up to 7 additional interviews (7x 1 hour)
- In depth code review
- Debrief startup for fact checking (1 hour)
- Debrief investor (1 hour) *(*)*

(*) We don't do these if it's an internal audit and there is no investor involved



CONTACT

Ready for an audit?

With our technical due diligence service, our staff engineers and experienced CTOs will technically de-risk your next funding round. In only 2 weeks.

Book a no-commitment call

(Secret: There are a few more slides left!)

Follow us around

We step in when others step out. We enhance software, transform legacy code, derisk investments and strengthen tech teams. Fearlessly honest, highly experienced, and obsessed with quality, we bridge the gap between technical and non-technical teams. When the stakes are high, we make the difference.

madewithlove, a world-class team of engineering experts

Our members are hand-picked masters of their crafts. We operate world-wide. Diverse cultures, interests, and backgrounds are a part of our company values, while physical borders are not.

You can find us in coworking spaces or home offices in Belgium, the UK, France, The Netherlands, Nigeria, Germany, South Africa, and Poland.

(click image to meet our team members)

WE WORK WITH INVESTORS ON A DAILY BASIS

CAPITAL MILLS

kpn ventures

9.5 VENTURES

CAREVOLUTION Investing in Health

inventures impact venture capital funds

FORTINO

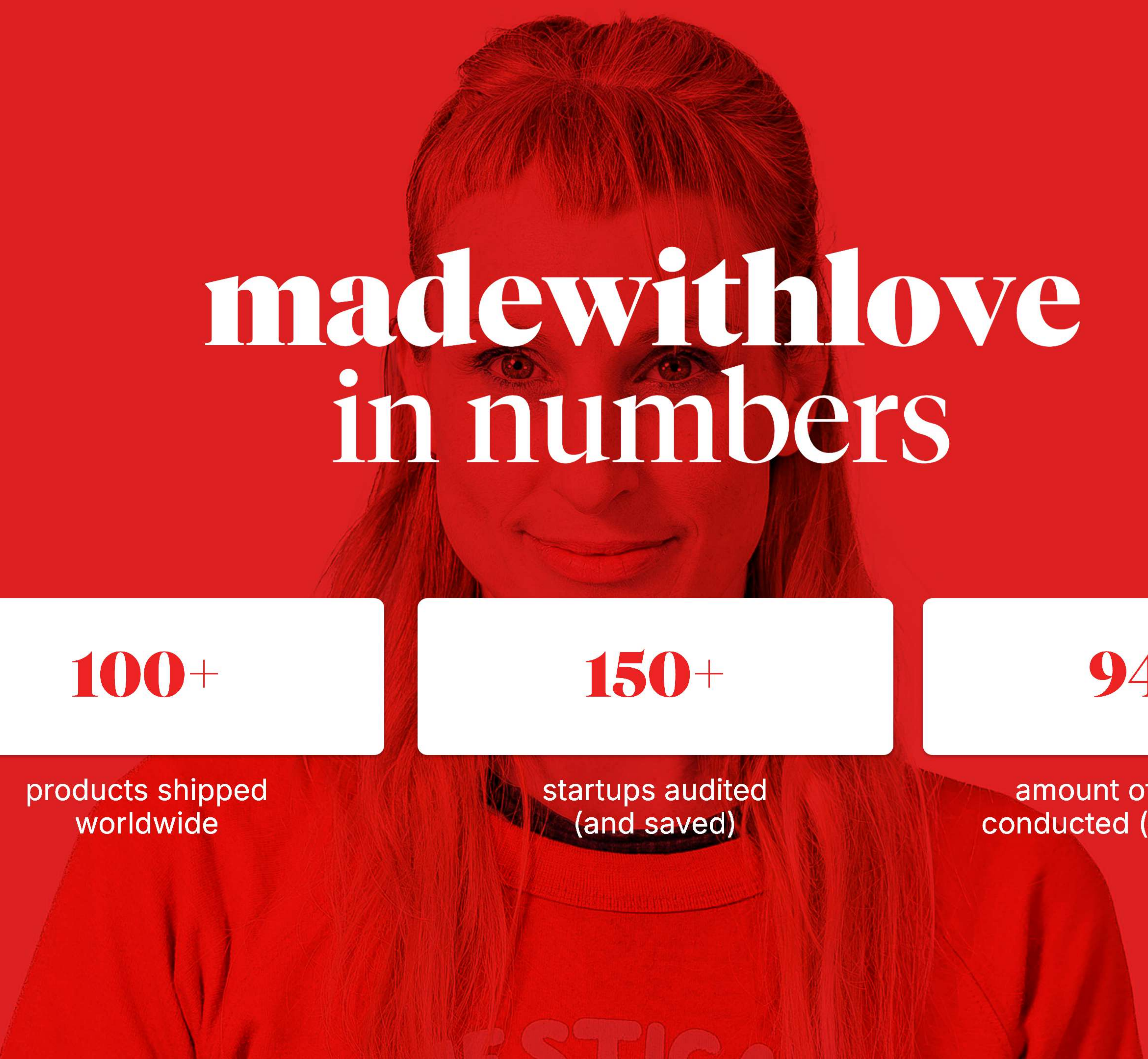
THE FAKTORY

PEAKCAPITAL

Smartfin

Freshmen entrepreneurs for entrepreneurs

VOLTA VENTURES



madewithlove in numbers

100+
products shipped worldwide

150+
startups audited (and saved)

941+
amount of interviews conducted (and counting)

Get in touch

Our technical experts are ready when you are

Andreas (our CEO) will jump on **a free, no-commitment call** with you, to find out why you need an audit, and explain what the best next steps would be — **in your specific case**.

Book a no-commitment call

Follow us around

audits@madewithlove.com

Andreas Creten, CEO