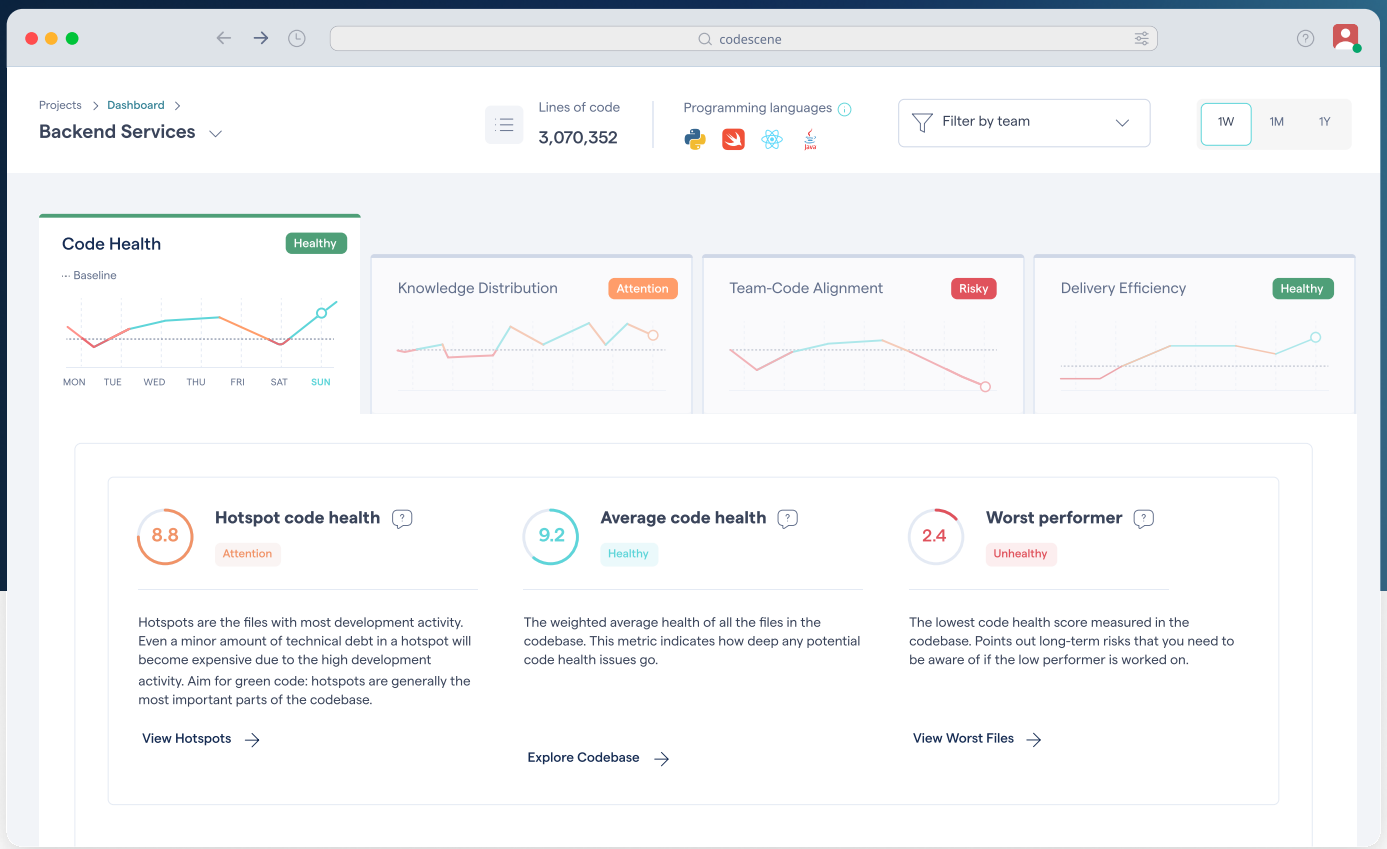
CodeScene

Four key factors behind high-performing software

CodeScene brings you real-time Software Engineering Intelligence with four factors. So easy to understand, it feels obvious.



Code quality issues cost time, money, and missed deadlines.
A healthy codebase enables 124% faster development time and contains 15 times fewer defects.

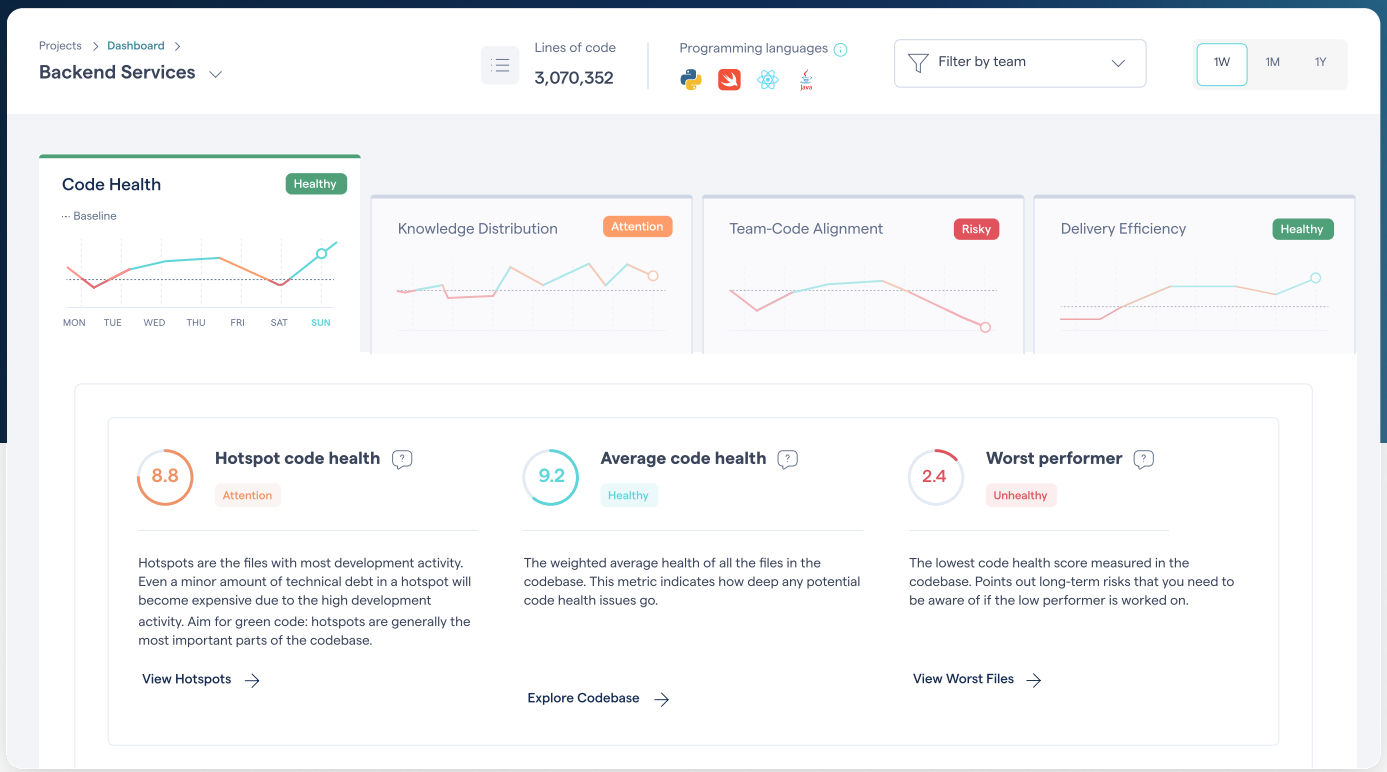
CodeScene's latest innovation is the simplest way to visualise, understand and improve your software.

Learn more at **codescene.com** or get in touch with us at **sales@codescene.com**

FIND THE NEEDLE IN THE HAYSTACK

Code Health

How healthy is your code? Is Technical Debt slowing you down?
Prioritize and reduce technical debt with precision.



What is Code Health?

Code Health is an aggregated metric based on 25+ factors scanned from the source code. Unhealthy code predicts increased maintenance costs and more defects.

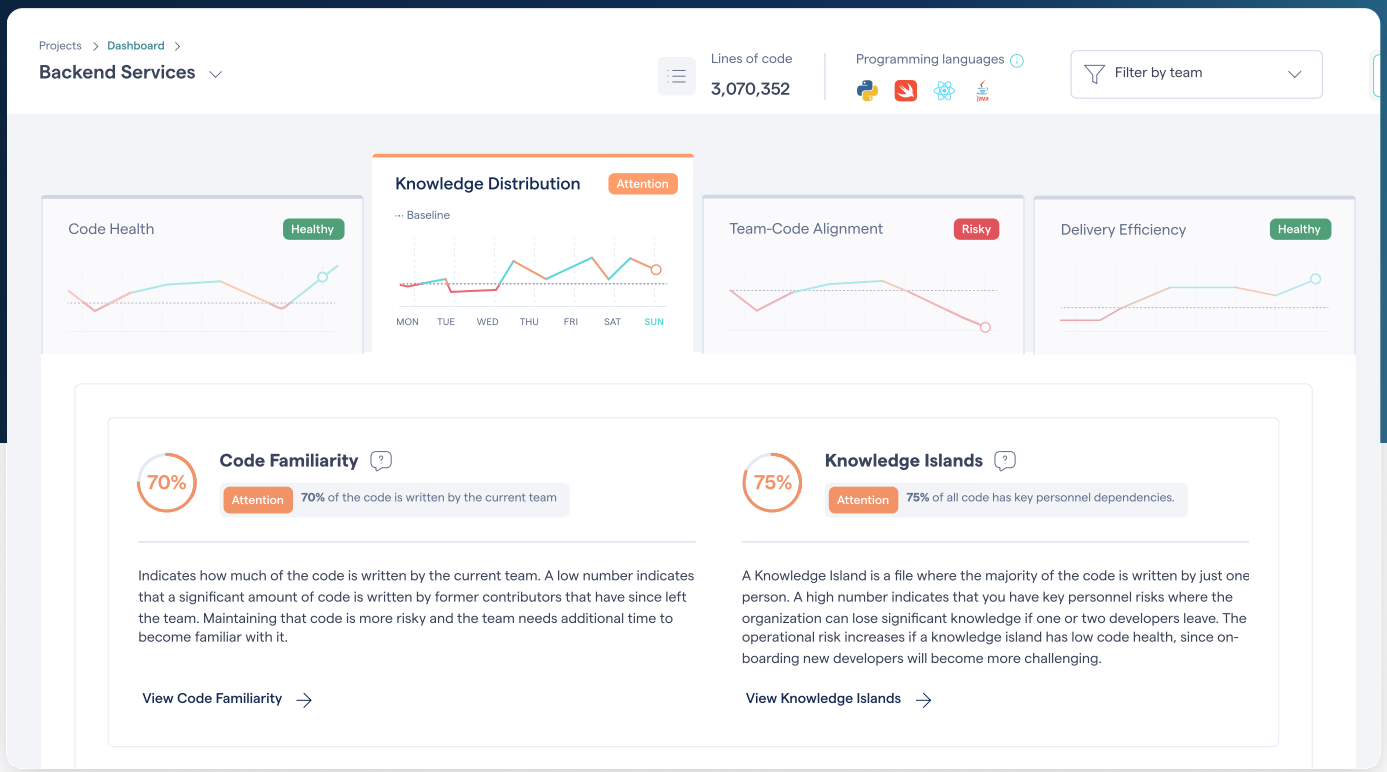
CodeScene calculates CodeHealth™ score and guides your Technical Debt management efforts.

- **Prioritize with surgical precision.** Focus remediation efforts where they provide the highest return on investment. Replace gut feelings with data.
- **Visualize code quality risks.** Understand how your organization writes code. Act now to maintain a high-quality, high-throughput environment.
- **Stop accumulating technical debt.** Automated, intelligent code reviews and quality gates supports development teams continuously.

STRONG CODE FAMILIARITY

Knowledge Distribution

Developer turnover leads to knowledge loss.
Ensure your teams have the knowledge to maintain and evolve your software.



What is Knowledge Distribution?

Successful software development largely depends on effective knowledge sharing within the organization. An organization loses collective knowledge when a code contributor leaves.

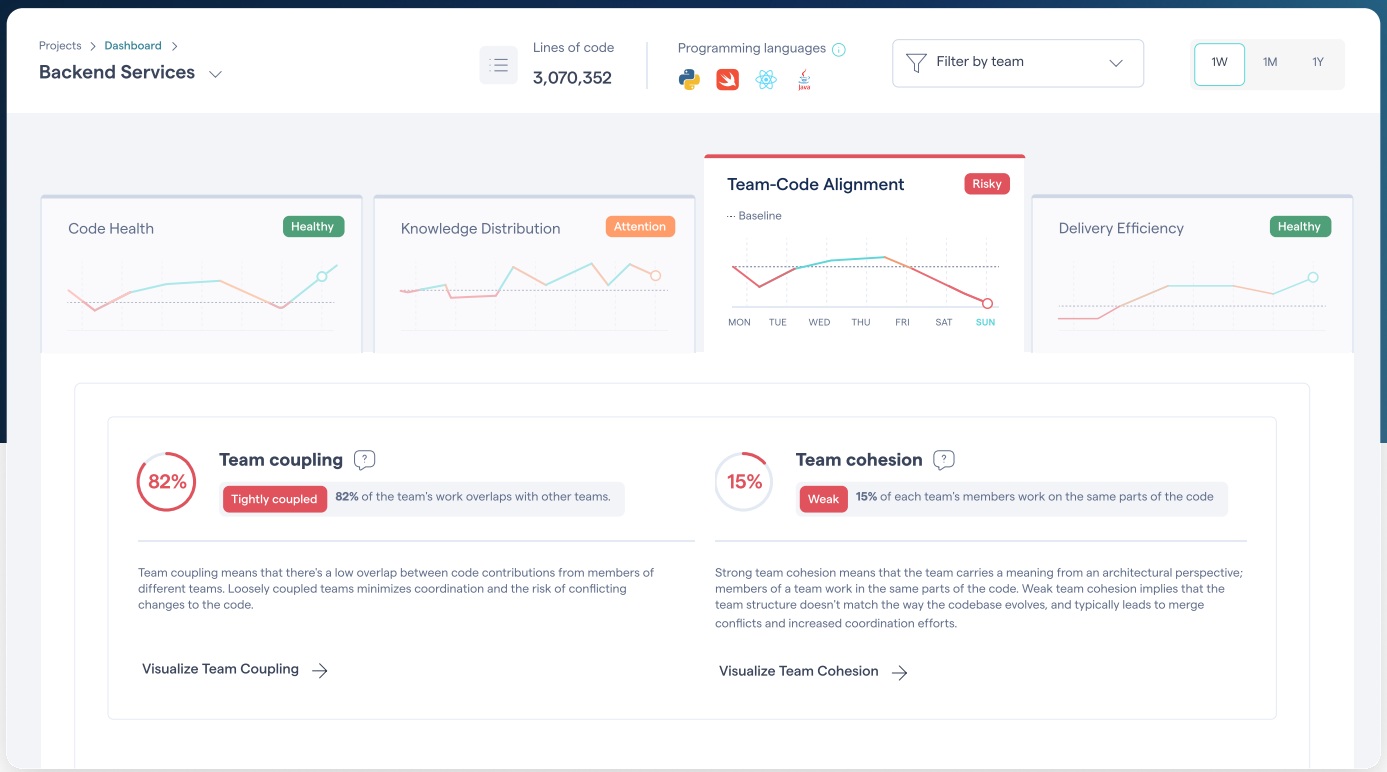
Mitigate this risk with CodeScene. Aim for high Code Familiarity and few Knowledge Islands.

- **Detect code written by people who have since left the company.** Low code familiarity requires additional time for learning.
- **Simulate off-boarding scenarios.** Collective knowledge is lost when a contributor leaves. Visualize the impact so that you can prepare.
- **Guide new developers** in unfamiliar territory with knowledge distribution maps.

EFFECTIVE COMMUNICATION

Team-Code Alignment

Optimize for better coordination and
set teams up for success.



What is Team-Code Alignment?

Getting the organizational side of software right is just as important as any properties of the code. The modularity of software design needs to align with responsibilities of the development teams.

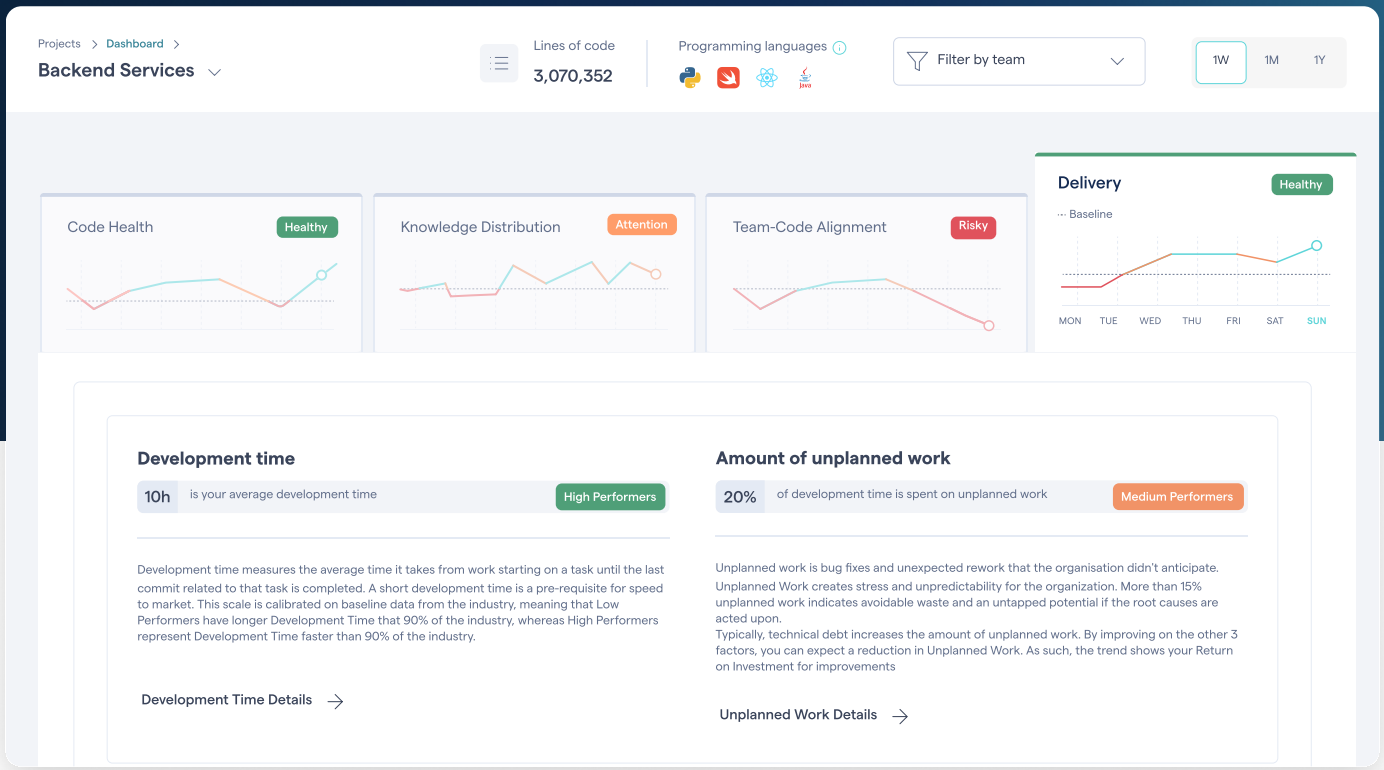
Do you have the optimal team structure that supports efficient team coordination and short lead times?

- **Visualize how your teams work and interact** with each other.
- **Reduce friction** by making sure your team structure aligns with the software architecture.
- **Identify dependencies and build more cohesive teams** that require less coordination and lead to fewer merge conflicts.
- **Organize teams for steady feature and value delivery.**

BUSINESS IMPACT

Delivery

See ahead and make data-driven decisions on the best way forward.
Visualize and improve your development process.



What is Delivery?

Delivery is an output metric that captures the efficiency of the overall process. Delivery is influenced by process, but also by how well you do on the other three factors. Improvements to Code Health, Knowledge Distribution and Team-Code Alignment lead to more efficient Delivery.

Efficient Delivery improves time-to-market and customer satisfaction.

- **Implementing new features in unhealthy code** leads to Unplanned Work and **takes 2x as long** compared to healthy code.
- **See how much time is spent on Unplanned Work** and explore the cost and root cause.
- **Get personalized Delivery feedback for each team.** What is their average Development Time, amount of Unplanned Work and Deployment Frequency?
- **Reduce Development Time** and **minimize Unplanned Work** with CodeScene.