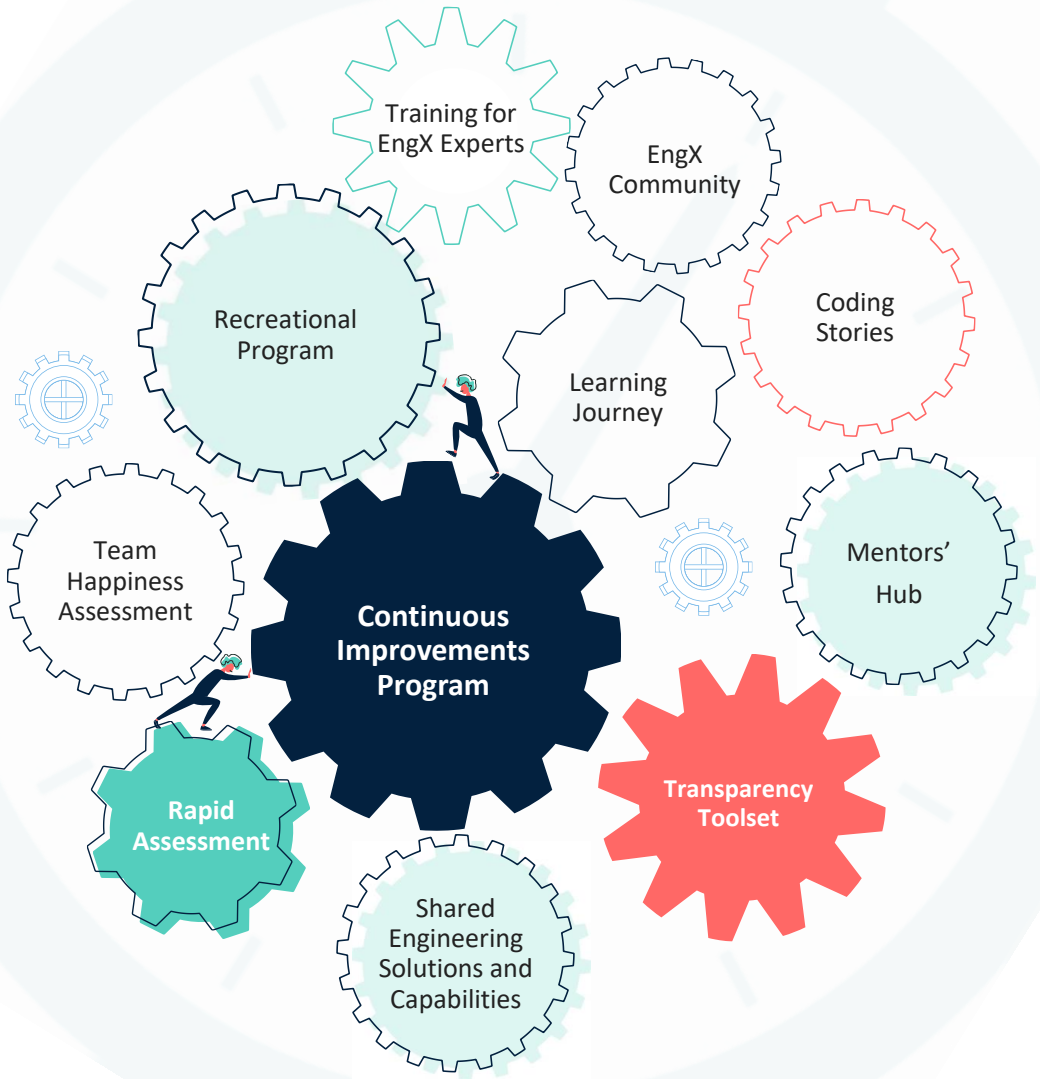
EngX Rapid Assessment

2022

EPAM Engineering Excellence Program

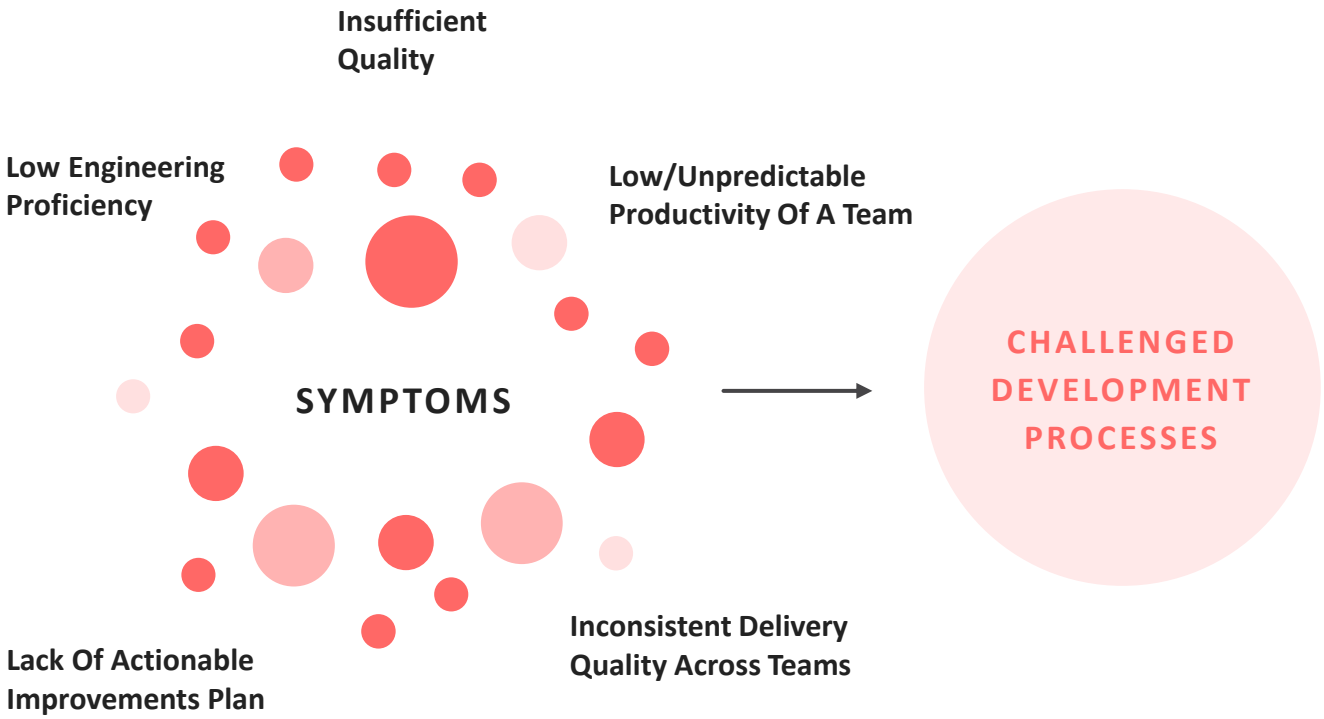


ENGINEERING EXCELLENCE PROGRAM HELPS IMPROVE DEVELOPMENT PROCESSES AND INCREASE ENGINEERING PROFICIENCY ACROSS ORGANIZATION BY PROVIDING:

- full picture of current state
- ideal maturity model
- roadmap for improvements
- establishing governance and sustainability through a robust toolbox of programs

EngX

Common Clients' Challenges



106 times slower

lead time from commit to deploy

and

208 times less frequent

deployments lead to missed opportunities quickly bring new business features to the market[1]

7 times higher

change failure rate

which is in combination with

2,604 times slower time to recover

from incidents lead to leaking of money and trust[1]

1 – Based on Accelerate: State of DevOps 2019

Rapid Assessment

Rapid engineering proficiency assessment of a product's team with the strong focus on development process. The goal is to identify gaps and bottlenecks which impact productivity, and creation the backlog of improvements.

INPUT

- Product team interviews
- Assessment survey for key aspects of SDLC
- Team happiness survey

TIMELINE

- **8** days assessment per team led by **EngX Expert**
- **6 hours** of efforts from key team stakeholders
- **10 minutes** of efforts from every team member

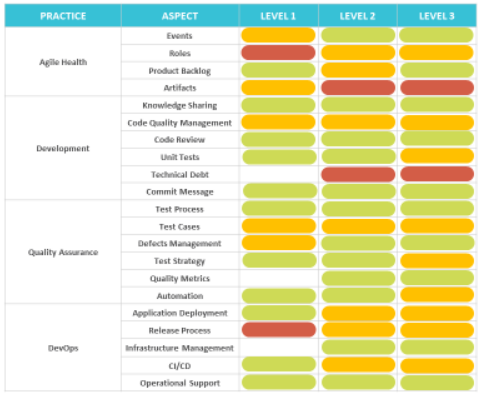
DELIVERABLES

- Findings and Improvements Report
- Engineering Maturity Level
- Delivery Pipeline Model (As-Is & To-Be)
- Best Practices Heatmap

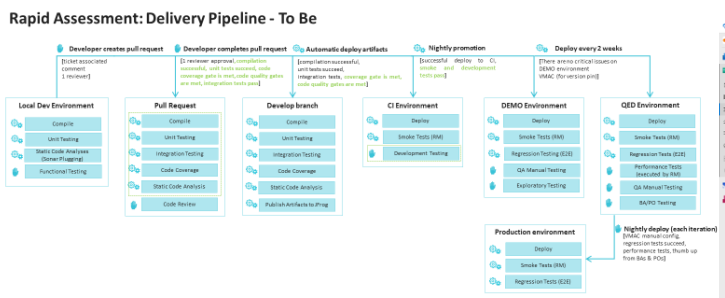
VALUE

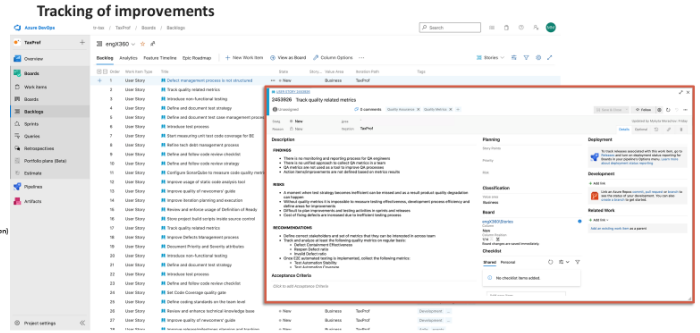
- Analysis of potential gaps in productivity and quality of a development process
- Transparency inside development teams
- Fresh view to reveal engineering bottlenecks

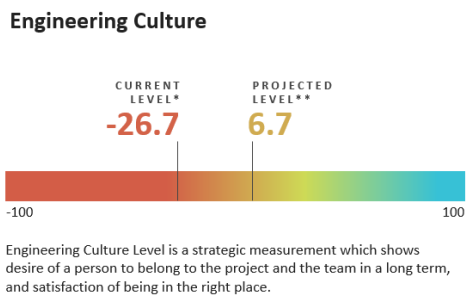
EXAMPLE OF DELIVERABLES



PRACTICE	ASPECT	LEVEL 1	LEVEL 2	LEVEL 3
Agile Health	Events			
	Roles			
	Product Backlog			
	Artifacts			
Development	Knowledge Sharing			
	Code Quality Management			
	Code Review			
	Unit Tests			
	Technical Debt			
	Commit Message			
Quality Assurance	Test Process			
	Test Cases			
	Defects Management			
	Test Strategy			
	Quality Metrics			
	Automation			
DevOps	Application Deployment			
	Release Process			
	Infrastructure Management			
	CI/CD			
	Operational Support			







Engineering Culture Level is a strategic measurement which shows desire of a person to belong to the project and the team in a long term, and satisfaction of being in the right place.

EngX

Engineering Excellence Framework

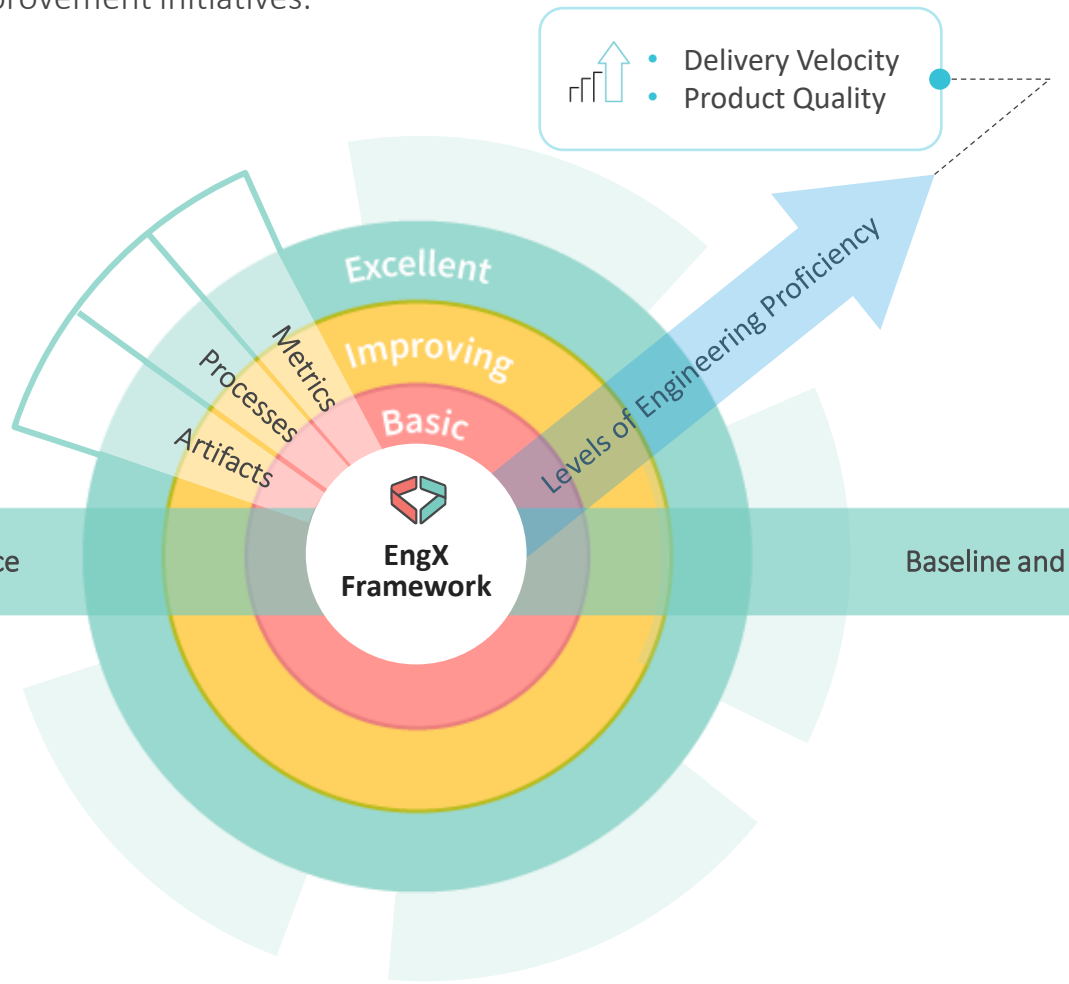
Condensed knowledge and experience about high engineering proficiency to become a foundation of various improvement initiatives.

EPAM

More than **25 years** of software development experience condensed by EPAM competency centers and practices

Industry

Latest best practices, researches and studies across the industry and related academic institutions



Assessment Methodology

to identify challenges which impact engineering proficiency, and create *backlog of required improvements*

Learning Journey

to enforce *high proficiency engineering skillset* across organization

Example of *Development* aspect of EngX Framework

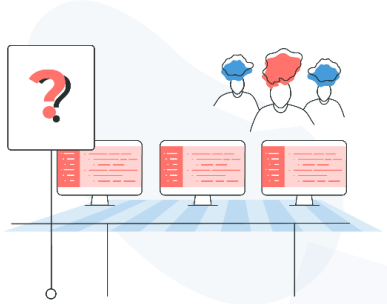
Development	Code Quality Management	Unit Tests	Basic Level	Improving Level	Excellent Level	Question/Metric/Artifact	Improving Value	Excellent Value
			Unit tests do not exist or written occasionally.	Unit tests are part of the development process and written using standard frameworks. Unit tests are run as a part of the build process, code coverage target is reasonably chosen, achieved and verified using mutation testing if possible.	Developers follow a test-first approach writing unit tests. Unit tests are first-class citizen and subject for the same code quality control as a production code. Code coverage target enforced by quality gate for every Pull Request.	Are unit tests written at the same time as functionality and included in Definition of Done?	Yes	Yes
						Do you use unit testing framework approved by company standards?	Yes	Yes
						Does unit test's code follow the same coding standards as a production code?		Yes
						Does every developer follow the test-first development approach, for example Test Driven Development?		Yes
						Is successful run of unit tests required for CI pipeline that builds and promote code?	Yes	Yes
						Is successful run of unit tests required for every Pull Request to be approved for merge?		Yes
						Is build breaker based on unit test coverage included in the CI pipeline that builds and promote code?		Yes
						Metric: Unit Test Coverage Metric	> 75%	> 90%
						Artifact: Definition of Done	Has unit tests	Has unit tests

Example of *Quality Assurance* aspect of EngX Framework

			Basic Level	Improving Level	Excellent Level	Question/Metric/Artifact	Improving Value	Excellent Value
Quality Assurance	Defect Management	Defect Management Tools	Defects are not tracked or are managed in a defect management tool and are uncontrollable items (defect comment, e-mail, chat message, stickers, verbal communication).	Defects are registered manually in special tool with enough data and supporting materials to reproduce/clarify. A single tool is being used for defect reporting with unified submission rules and workflow across the organization. Change request process is identified and followed.	Unified special tools are in use for all defects. All defects are being registered directly from the test automation reports or test runs. No manual effort is required to collect all required data and supporting materials. Live dashboards are in place for defects status tracking and reporting.	Are live dashboards used for reporting and status tracking?		Yes
						Do you create new defect directly from automated or manual execution runs?		Yes
						Do you separate defects from change requests?	Yes	Yes
						Are client reported defects logged and tracked in the same tool?	Yes	Yes
						Do you have defect workflow unified for all projects across organization?	Yes	Yes
						Is defect workflow defined and used for all defects?	Yes	Yes
						Do you add to defect supporting materials to ease investigation and resolution (e.g., environment details, screenshots, logs)?	Yes	Yes
						Do defects have severity and priority?	Yes	Yes
						Do you manage defects in specialized tools (e.g., Atlassian Jira, Bugzilla)?	Yes	Yes

Rapid Assessment and Continuous Improvement Cycle

Agile-like improvement cycles owned by engineering team and driven by EngX Expert until high level of engineering proficiency is achieved



Rapid engineering proficiency assessment of a team to identify gaps and bottlenecks which impact productivity, and create *backlog of required improvements*

8 days

Rapid Assessment

Metrics Monitoring Setup

Team Happiness Assessment

Rapid Re - Assessment

Improvements Implementation

EngX Expert Consulting

Bi-weekly standups

Continuous Improvement Cycle

12 WEEKS

Improvement Iteration Planning

Refinement of Improvements Backlog

EngX

Rapid Assessment Deep Dive

EngX

Aspects of EngX Rapid Assessment

Agile

- Roles
- Events
- Artifacts
- Collaboration

Engineering

- Knowledge Sharing
- Code Quality Management
- Technical Debt Management
- Code Traceability
- Secure Development
- Architecture Foundation

Testing

- Test Strategy
- Test Cases Management
- Defect Management
- Test Automation

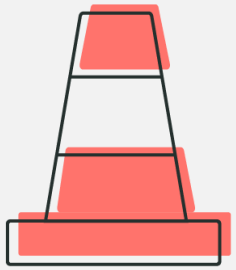
DevOps

- CI/CD
- Operations Excellence
- Cloud Readiness
- Metrics and Monitoring

TEAM HAPPINESS

WE DO

+ Trust the development team
+ Review processes and practices
+ Provide a backlog of improvements



WE DON'T

× Research engineering data and artifacts
× Review code and architecture
× Create a transformation roadmap*

*but we work together to help you build it

Who do we need to participate?

Key Stakeholders

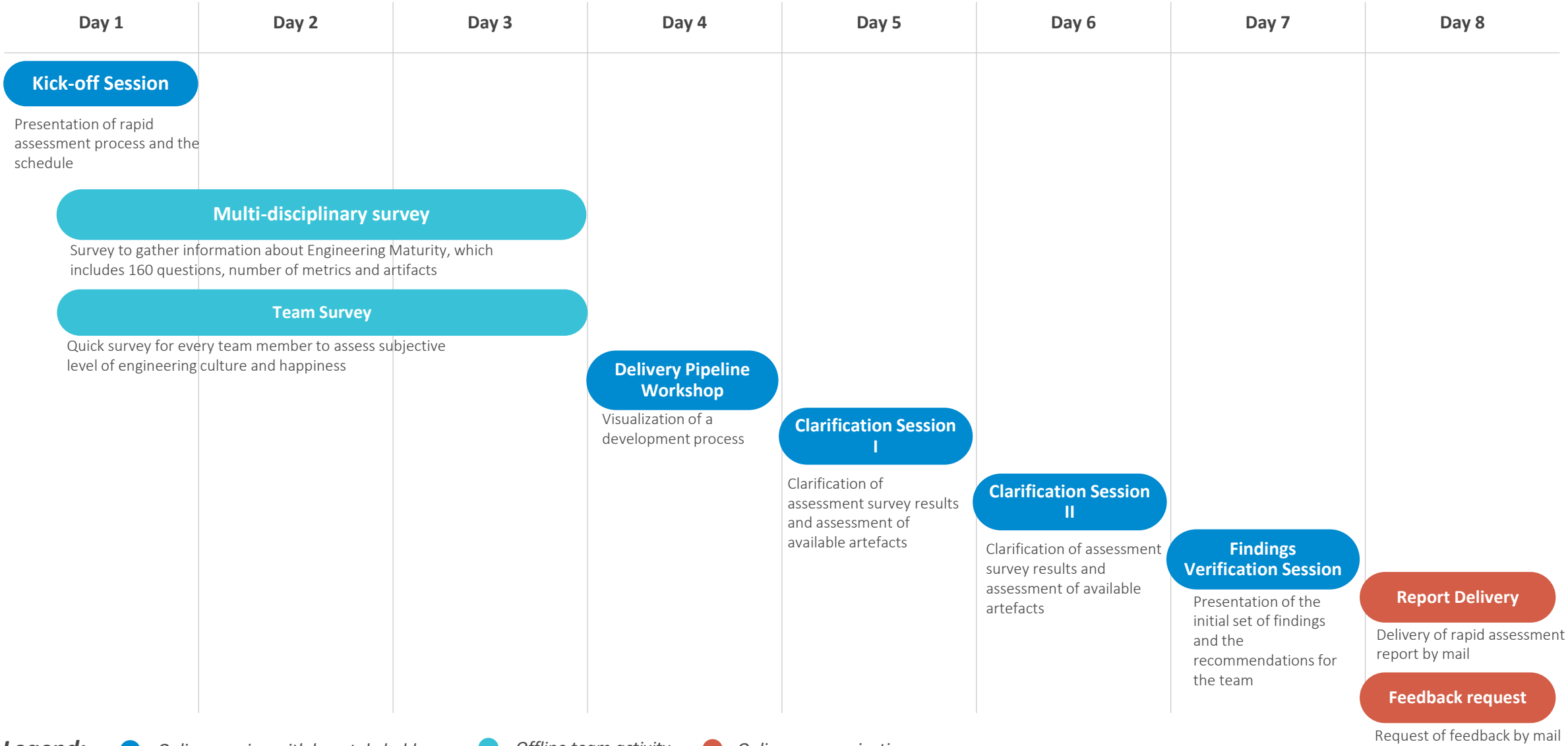
Effort: approx. 6 hours per person

- Project Manager
- Development Lead
- QA Lead
- DevOps Lead
- Product Owner / Scrum Master

Development Team

Effort: 10 minutes per person

What is our timeline?



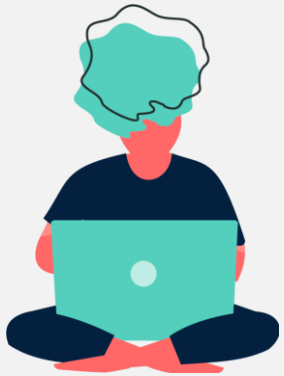
Legend: *Online session with key stakeholders* *Offline team activity* *Online communication*

EngX

Deliverables

What do we deliver?

EngX Expert



Executive Summary Report

Delivery Pipeline Model (As-is & To-be)

List Of Findings and Improvements

Best Practices Heatmap

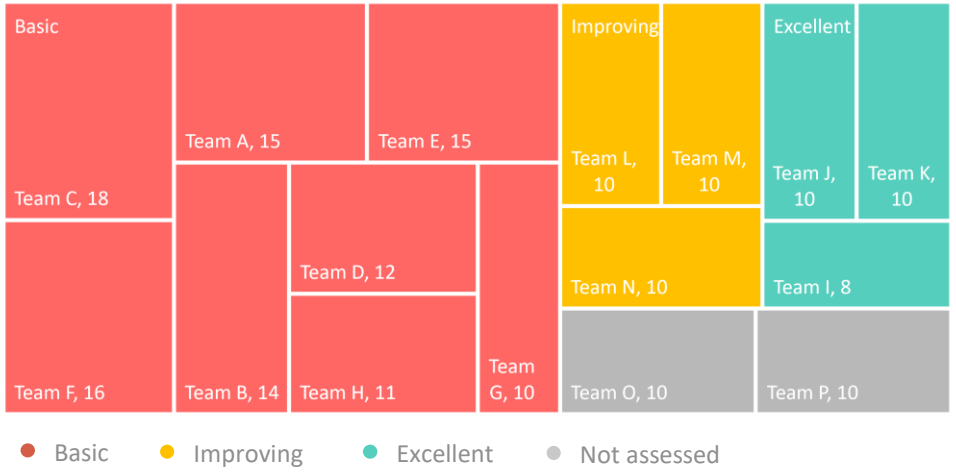
Engineering Maturity Level

Team Happiness Report

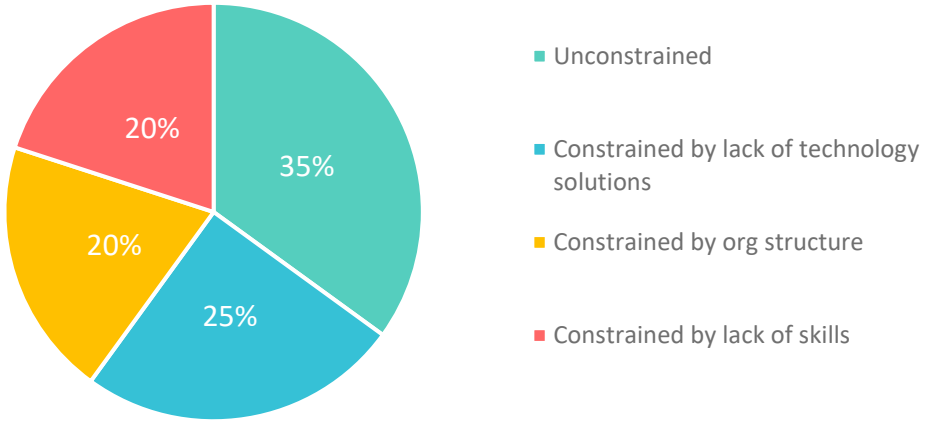
Executive Summary Report

Bi-weekly reporting of engineering proficiency level across organization and real-time monitoring of improvements process.

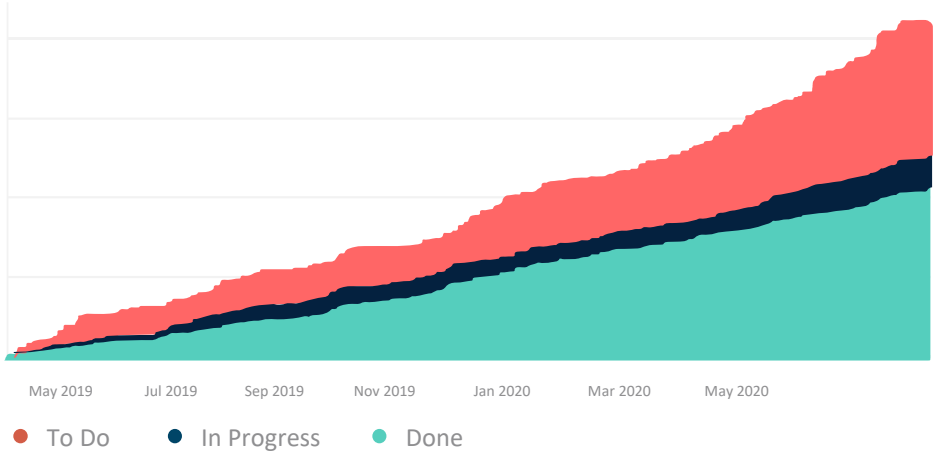
ENGINEERING EXCELLENCE MAP



IMPROVEMENT BACKLOG BY TYPE OF CONSTRAINTS



IMPROVEMENTS BURN-UP CHART



TOP CONSTRAINTS

- C# module is not enabled in SonarQube which prevents static code analysis set up for 7 teams.
- BDD skill set is inconsistent across engineering teams which prevents introduction of balanced test pyramid across 15 teams.

EngX

Summary and Heat Map

ACHIEVED: LEVEL 2

Improvements Focus

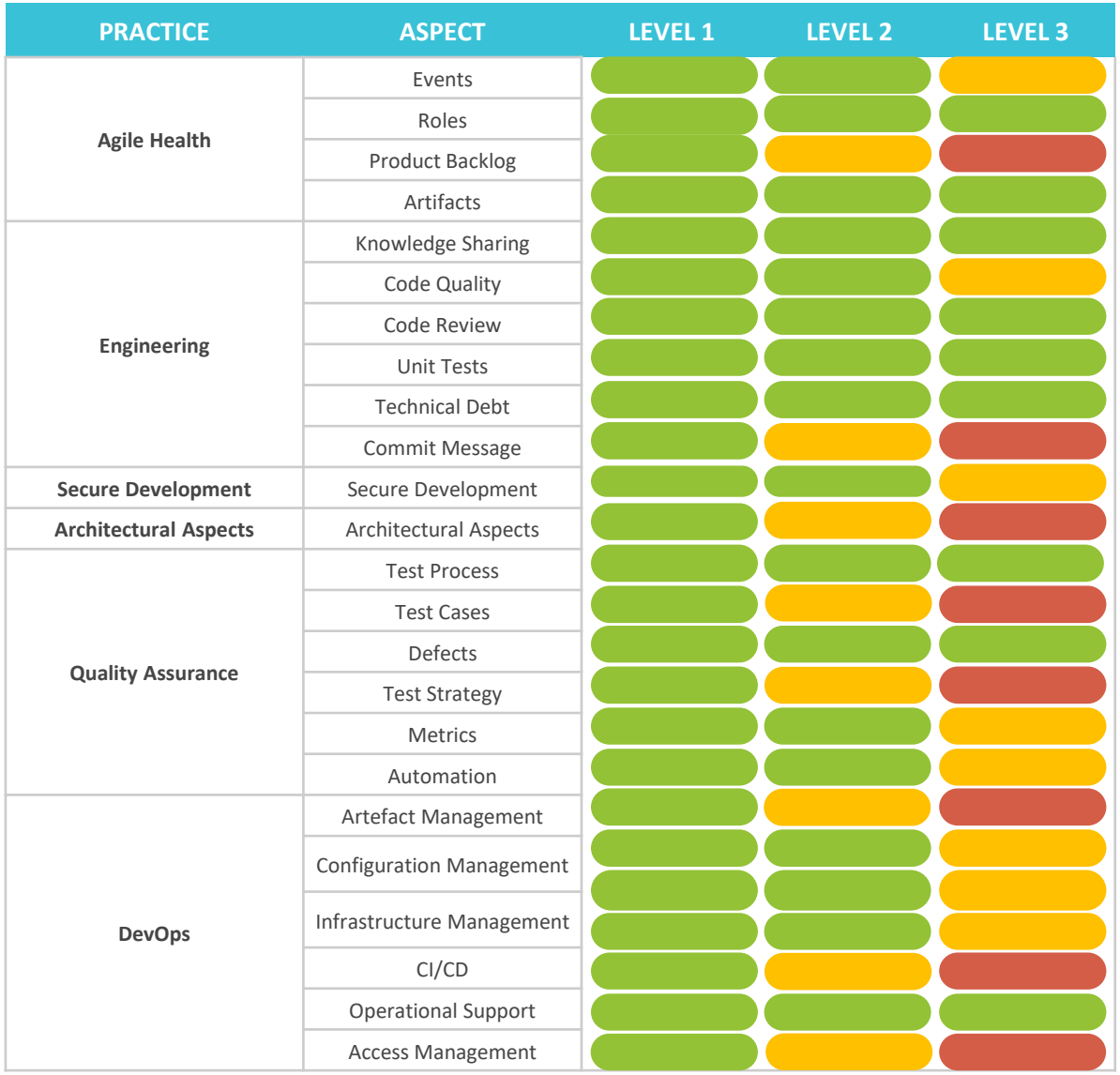
Create Testing Strategy for the whole product

Constraints

Several vendors work on the same product

Top Issues

- There is no common tests strategy aligned between all development streams;
- Test automation coverage is low;
- There is no verification job for Pull Request;
- Definition of Done and Definition of Ready are not used
- Acceptance Criteria are not mandatory for user stories



EXAMPLE OF DELIVERABLE

Lack of Quality Gates

KEY FACTS

- The earliest CI job is executed on master branch
- About 20-30% of master build failures are caused by unit test and compilation failures.

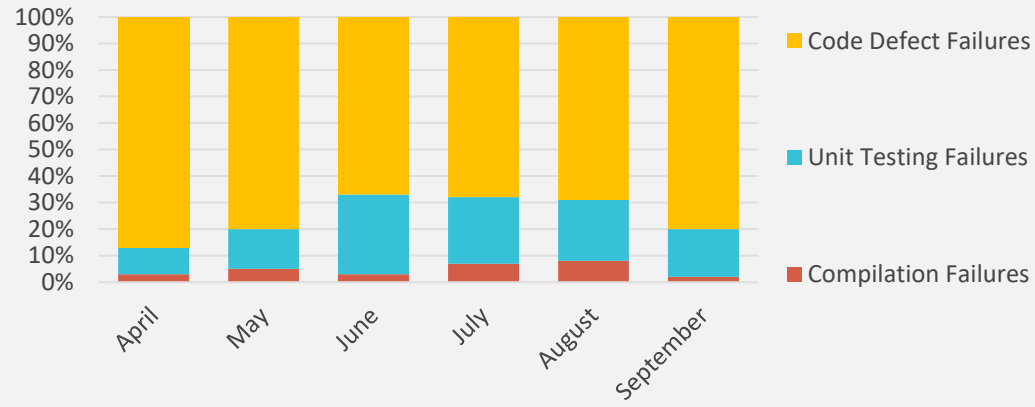
IMPACT

Lack of quality gates is a cause of decreased stability of the master branch with the following consequences:

- It will be impossible to create new branch while the defect in the master branch is not fixed, in opposite case defects will propagate to derived feature branches
- Merge of new features into broken master branch hides defects in merged functionality

METRICS

Master build failure types

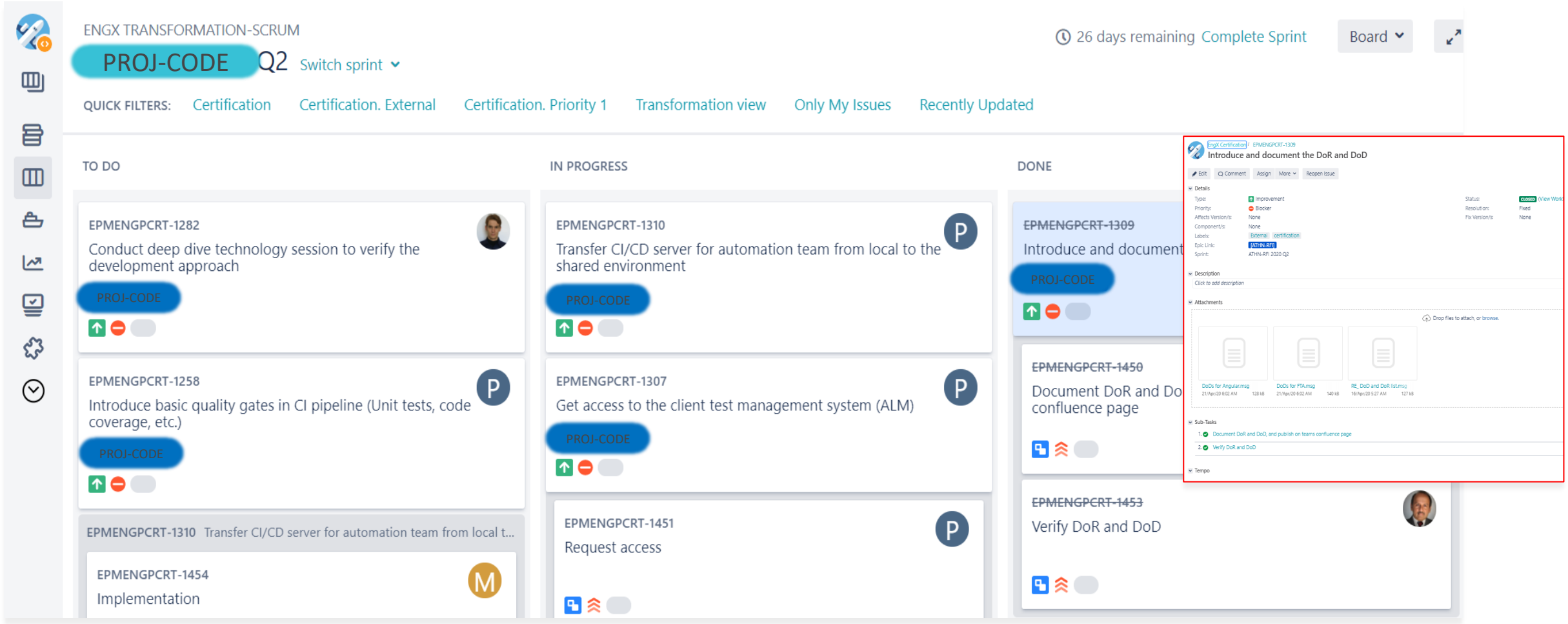


RECOMMENDATIONS

- Introduce CI job which is triggered by Pull Request creation/update
- Include additional verification to PR associated CI job:
  - Unit Tests
  - Code Coverage
  - Component Tests
  - Integration Tests
  - E2E Smoke test

EngX

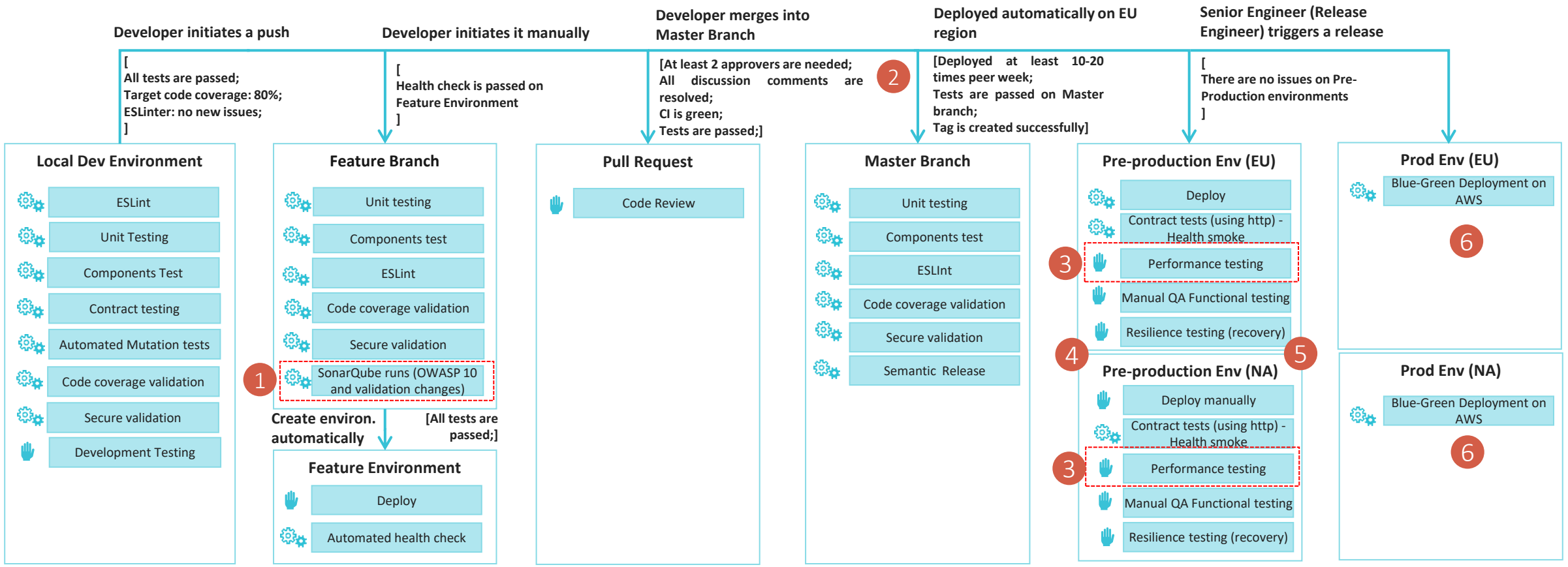
Tracking improvements



EngX

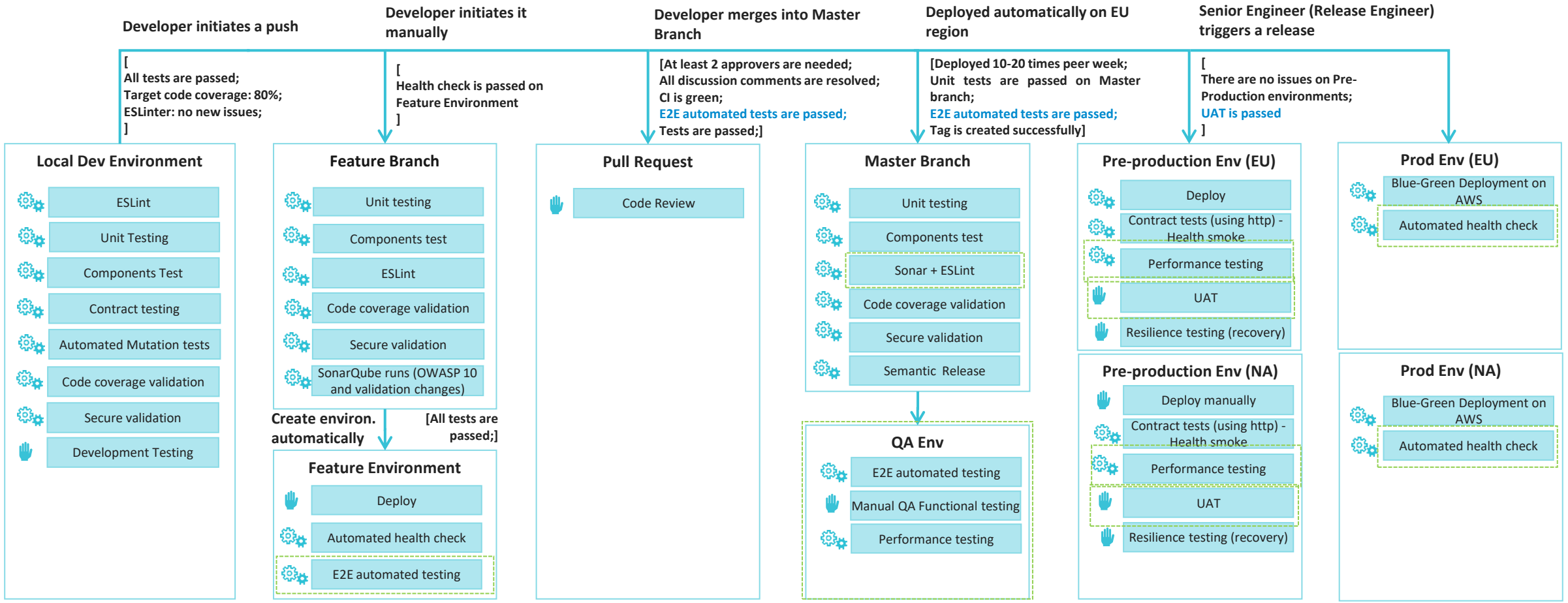
EXAMPLE OF DELIVERABLE

Delivery Pipeline As Is



1. Currently SonarQube is not used by developers daily due to technical limitations
2. There is no QA environment
3. Performance testing triggers manually
4. There are no E2E automated tests
5. There is no UAT on Pre-Production environment
6. There is no Health check on Production

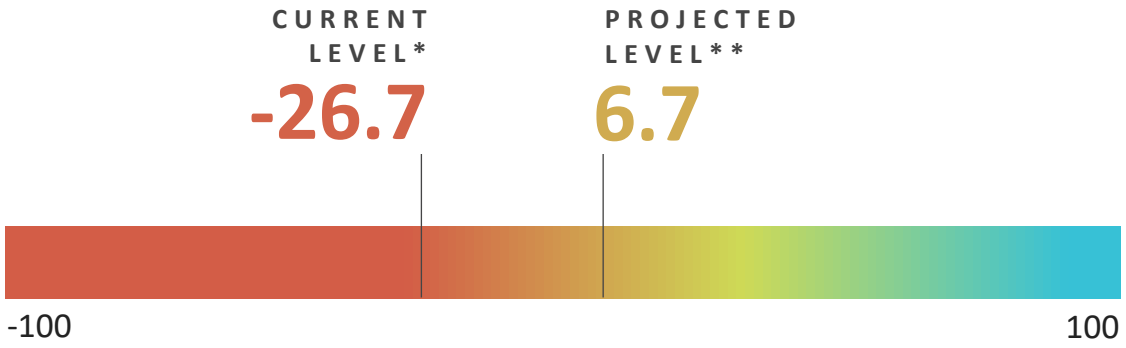
Delivery Pipeline to Be



Legend: *Proposed environment or testing activity* **<Quality Gate>** *Proposed quality gate*

TEAM HAPPINESS ASSESSMENT: EXAMPLE OF DELIVERABLE

Engineering Culture



Engineering Culture Level is a strategic measurement which shows desire of a person to belong to the project and the team in a long term, and satisfaction of being in the right place.

* Engineering Culture Level is measured using the Net Promoter Score which is an index ranging from -100 to 100 that measures the willingness of customers to recommend a product or service to others. Number below 0 shows negative perception, while number above 0 – positive.

** Projected level is a potential level of engineering culture if team expectations are met.

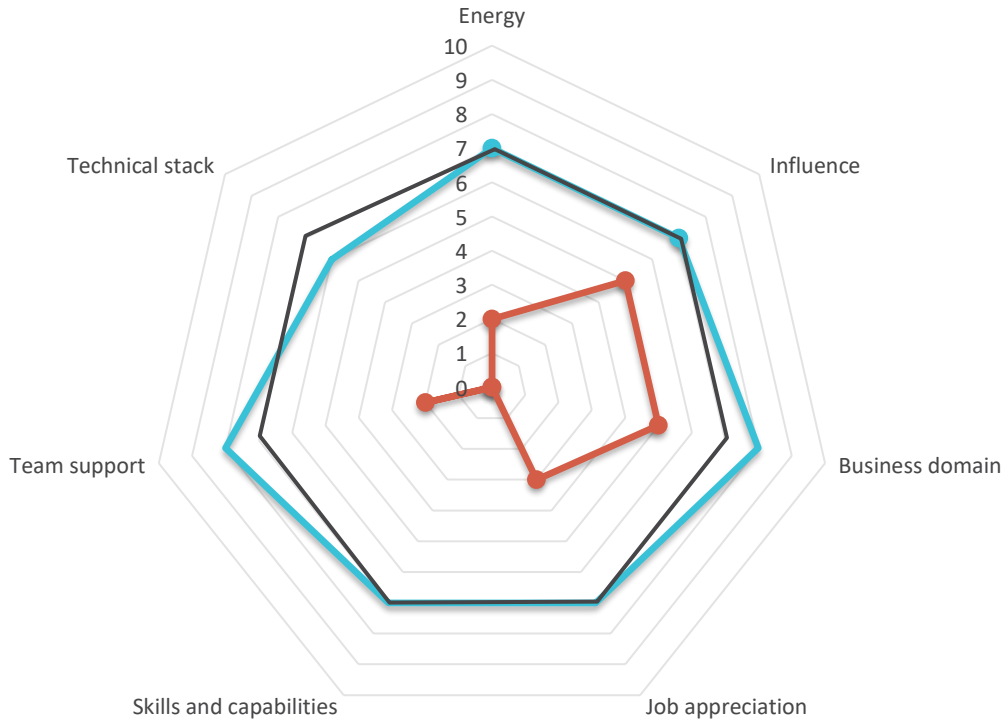
SUMMARY

- Perception of engineering culture is **negative**; some of the developers have significant concerns about the project.
- There's a possibility for NPS to increase if team's expectations are met.
- NPS could potentially decrease if the team's expectations aren't met, which could lead to attrition growth

TEAM EXPECTATIONS

- Upgrade technology stack. This is the major concern across the team.
- Review system design and modernize it. Seek for new opportunities to utilize more cloud functionalities.
- Build closer communication with business users to improve customer satisfaction and meeting client needs.
- Improve development processes: Planning, Code Review.
- Review prioritization process to have important feature implemented first.
- Setup architectural meetings with platform solution architects to share ideas and inform team what happens on the platform level.
- Make sure project roadmap is clearly defined for the release and communicated to the whole development team.

Recreational Demand



Legend: *Acceptable state* *Team median* *Team minimum*

FINDINGS

- **Energy:** Most of the team feels ok. However, some people need urgent attention due to tech stack and skills appliance on the project.
- **Team Support:** High level of team support and solidarity. However, some team members do not feel engaged enough to the team processes.
- **Skills and capabilities:** About half of the team doesn't not feel satisfied with how they apply their skills on the project. This has drastic short- and long-term impact on the team productivity.
- **Job appreciation:** Bigger part of the team does not feel they are properly rewarded for their job. Ignoring this may result on productivity drop in a short- and long-term.
- **Technical Stack:** The most urgent concern of the project team. This has critical impact on short- and long-term team productivity. Ignoring this can lead to people attrition.
- **Business Domain:** The team does not show interest in current business domain area. However, it does not have big impact on team productivity.
- **Influence:** Majority of the team members do not feel that they can influence development process and project architecture. It has significantly negative impact on short- and long-term team productivity.

Case Studies

MULTINATIONAL MEDIA CONGLOMERATE

Engineering Proficiency Assessment

EPAM helped a multinational media conglomerate baseline development processes with the engineering excellence fundamentals and identify areas for improvement.

In the result of 22 teams' assessment EPAM identified gaps and bottlenecks which impacted productivity and built a backlog of over 200 potential improvements, driving changes across the development organization.

Engineering Excellence Map

Team Happiness Map

"Our organization has objectives centered on engineering excellence, continuous improvement, and team building; we often struggle with identifying tangible actions to achieve those objectives. The EngX assessments provided a thorough **deep-dive effectiveness review** specific to each team to include helping them identify a backlog of improvement.

I greatly appreciated these assessments as it **empowered our teams** to look at their own processes, pipeline, and capabilities with an eye **toward improvement** in which they normally don't get that chance while they're in the midst of rapid delivery.

We also appreciated the **EPAM EngX Team's expertise, guidance, and coaching**. Although our teams are **effectively delivering value to our customers**, I look forward to partnering with EPAM EngX in the future to help us go from effective to world-class."

Vice President of Technology

CASE STUDY:

Weather Forecast Company

2 ····▸ 5	8 ····▸ 9	0 ····▸ 67
number of projects	customer satisfaction	engineering culture NPS

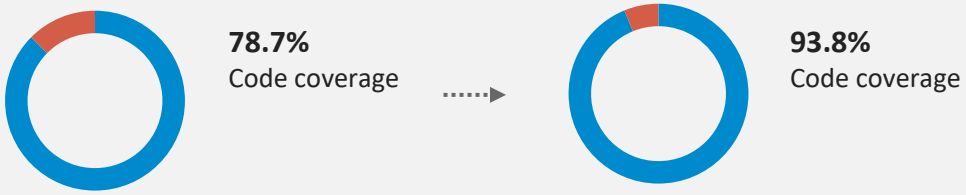
30% more team performance

5% reduction in technical debt

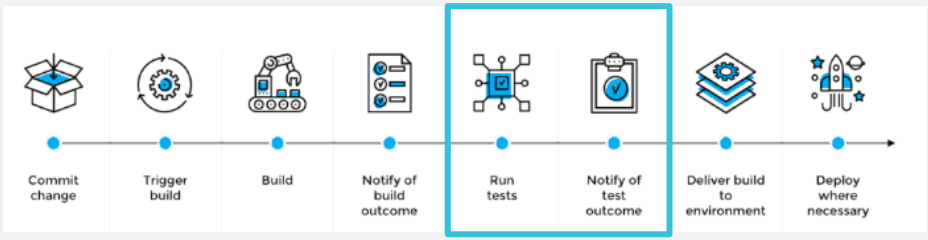
IMPLEMENTED IMPROVEMENTS

All improvements were implemented by the development team

- Unit test coverage is increased and included in Continuous Integration process



- Smoke and regression automation tests are included in CI/CD pipeline:



- Defined a process to track prudent technical debt on the project
- Established Security development process and included into SDLC
- Dedicated environment for automation testing process was requested
- Production / runtime environment requirements are specified and documented

EngX

Thank You!

For more information, contact

AskEngX@epam.com