


UdaPeople

Proposal of CI/CD process

Mission: Deliver value
to the company
through improved
frequency of software
deliveries

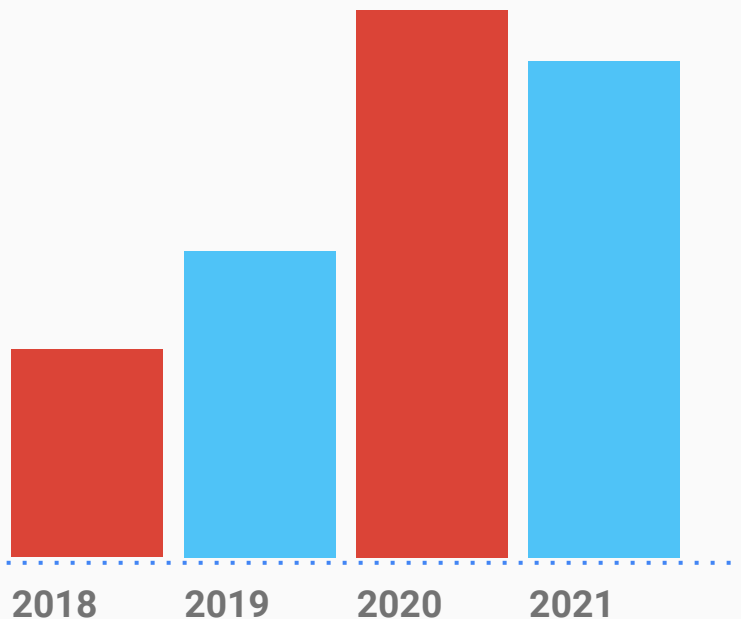
A laptop screen is visible in the background, displaying a dashboard. The dashboard features a line graph at the top with a blue line showing an upward trend, and a pie chart below it with a green slice. The text 'New Visitor' and 'Returning Visitor' is visible next to the pie chart. The laptop keyboard is partially visible at the bottom.

The Problem

The mean time to deploy new versions manually is rising

This leads to some effects:

- 😓 More human costs to deploy new builds
- 😓 Less time to build new features
- 😓 Longer time to market of new features





The Solution

Continuous Integration:

- ★ Pushing code frequently and run unit and integration tests, avoiding conflicts

Continuous Delivery:

- ★ Deploying the code built frequently to Development and Staging environments, and once are tested and reliable, 📶 promote to Production

Team

Responsibility chart



Quality Assurance

Writes automated tests
Performance tests
Security testes



DevOps Architect

Takes care of CI/CD
infrastructure, like
versioning platform and
pipelines to deploy the
code to Servers



Developer

Pushes the code daily on
the right branches
With CI/CD is able to run
the tests and deploy
automatically the code to
the environments

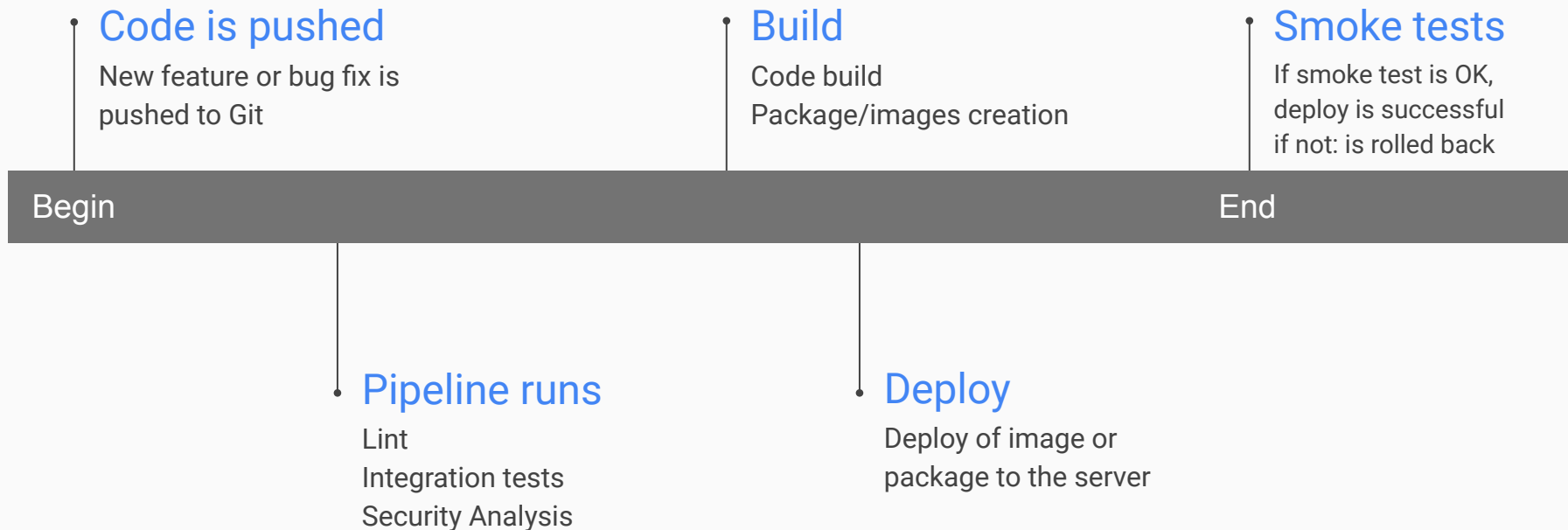


Project Lead

Provides the direction for
Team, prioritizing
adequately new features,
bug fixes and technical
debt reduction

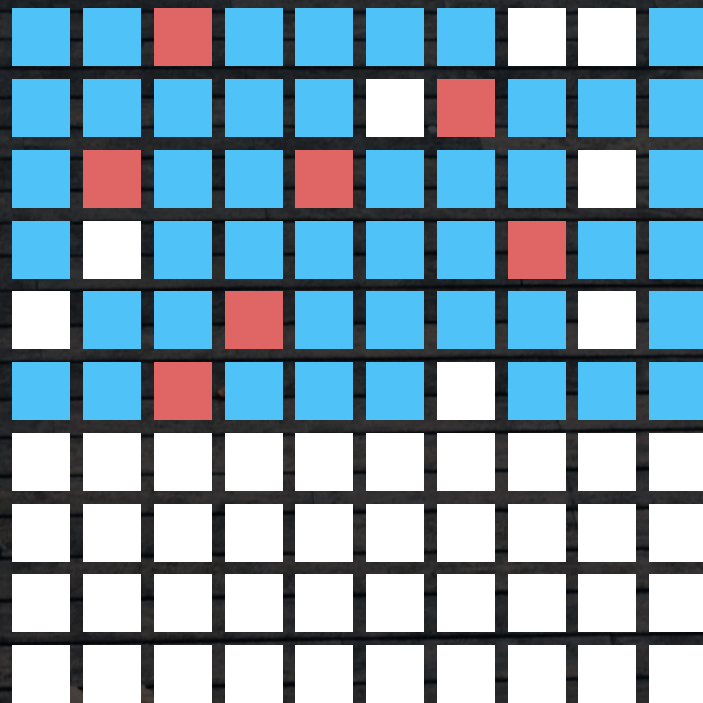
Process - Pipeline

Vision about how is the process of Continuous Integration / Continuous Delivery



Have you pushed the code today?

- Small pieces of software are delivered on a daily basis, reducing risks of big changes
- Tests, Builds and Deploy are automatic and frequent
- Broken Builds are noticed and corrected before going to Production
- Manual Build “hell” is avoided



Questions?