

Goal

We need to be more efficient

Goal

We want to better understand
what our customers want

*"If I had asked people what they wanted,
they would have said faster horses." -
Henry Ford*

Goal

We want to adapt to the market
more quickly

Goal

We want to get something to production much faster

Goal

We want to be more agile
because our management wants
us to

Goal

We want to have more fun at
work

Goal

We want to be more predictable
in delivering products

Goal

We want to make less errors in
our work

*"Success consists of going from failure to
failure without loss of enthusiasm" -
Winston Churchill*

Goal

We want to increase the quality
of what we deliver

Goal

We want to do what our
competitors do

Goal

We want to understand what
quality means to us

Goal

We want to release more reliable
to production

Goal

We want to be less managed by
our boss

Goal

We want to be more flexible in changing the functional scope

Goal

We want to be less tool
dependent

Goal

We want to focus on core
business

Goal

We want to know better what to
do

Goal



Goal



Context

Small company

Context

Large company

Context

IT project

Context

Non-IT project

Context

Requirements are mostly known

Context

Requirements are unknown

Context

Technically challenging

Context

Working with known technology

Context

Project

Context

Product

"If it is free, you are the product"

Context

Fixed dedicated team(s)

Context

Flexible parttime team(s)

Context

Component team

Context

Feature team

Context

External client

Context

Internal client

Context

Long-lasting project

Context

Short-running project

Context

External contractors

Context

Internal development team

Context

Maintanance project

Context

New product development

Context

One team

Context

Multiple teams

Context

Hierarchical organisation

Context

Flat organisation

Context

The organisation is aware of what it wants to achieve

Context

The organisation has no idea
where it goes

Context

**Roles in the organisation are
clear**

Context

**Roles in the organisation are
unclear**

Context

Co-located team

Context

Team of remote working people

Context

Working for multiple clients

Context

Working for a single client

Context

Separate development and
operations teams

Context

Devops teams

Context



Context



Practice

Arrange an external audit

Practice

Calculate the cost of delay

Practice

**Change your organisation to a
flat one based on circles**

Practice

Clean up the team room

“That’s the only thing you can do with a mess. Start cleaning it up, a little at a time.” - Lisa Wingate

Practice

Conduct an A/B test on your
assumptions

Practice

Create a Causal Loop Diagram on
the problem

Practice

Create a customer journey
mapping

Practice

**Create a Lean Canvas of your
product**

Practice

Create teams by self selection

Practice

Dedicate one person for ops
during the sprint

Practice

Develop a Minimal Viable Product

Practice

DevOps

Practice

Do a regular planning of work

"Plans are nothing; planning is everything."

- Dwight D. Eisenhower

Practice

Do a regular retrospect

"If you don't feel stupid about things you did last week, you're probably not learning fast enough." - Jurgen Appelo

Practice

Do a regular review (demo)

"Do or do not, there is no try." - Yoda

Practice

Do a user story mapping

Practice

Do nothing

*"The sweetness of doing nothing." -
Elizabeth Gilbert*

Practice

Do test-driven development

"Always code as if the guy who ends up maintaining your code will be a violent psychopath who knows where you live." - John Woods

Practice

Enhance team responsibility

Practice

Find a successful example and
make sure everyone knows about
it

Practice

**Find relevant literature and leave
it around the office**

Practice

Focus communication on richer forms

"Either write something worth reading or do something worth writing." - Benjamin Franklin

Practice

Hire a coach

Practice

Hire external expertise

Practice

Implement the 1:3 3:1 rule across
the team

Practice

Introduce product owner

Practice

Introduce scrum master

“A good scrum master can handle two teams, a great can handle one.” - Unknown

Practice

Locate waste in the development cycle and address it

Practice

Make a product backlog

Practice

make an inventory of your
applications

Practice

Make self-organising teams

Practice

**Measure the test coverage and
make it visible**

Practice

Measure what your team delivers

Practice

Measure your users behavior

*"In God we trust; all others must bring
data"*

Practice

Monitor and manage
development flow

Practice

Monitor your systems for bugs
and strange things

Practice

Organize yourself as feature
teams

Practice

Pivot your business

Practice

Practice what you preach

"The reality is that the only way change comes is when you lead by example" - Anne Wojcicki

Practice

Refactor your code

Practice

**Remove dependencies on others
in the organisation**

Practice

Sack a manager

Practice

Sack the team

Practice

Scope the product down

Practice

Ship the product now

"A lot of times, people don't know what they want until you show it to them." -

Steve Jobs

Practice

Split user stories

Practice

Start every day with a standup

Practice

Start using a release train

Practice

Start using continuous integration

Practice

Start with automated testing

Practice

**Start with daily releasing your
product**

Practice

Start with pair programming

Practice

Start with portfolio management

Practice

Start with scrum-of-scrums

Practice

Stop the project

Practice

Stop using a tool

Practice

Talk to the customer

“In a startup, no business plan survives first contact with customers.” - Steve Blank

Practice

Use a new tool

Practice

**Use a Proxy for the product
owner**

Practice

Visualize your workprocess

Practice

Work in iterations of one month

Practice

Work in iterations of two weeks

Practice

Work in weekly iterations

*"If it's hard to do, do it more often" -
Unknown*

Practice

Work with hypotheses on your
product

Practice



Practice

