

Ak Way Defaults Cheatsheet v0.3.0



Purpose

Return creativity and joy to the world of software engineering & delivery.

Vision

To be the world's most trusted partner for empowering businesses to accelerate the delivery of their greatest ideas

Mission

Businesses, at any scale, are awash with great ideas. However, the cost of learning which idea customers will love is too high. We evaporate this constraint and build quality in by accelerating cloud native adoption using next-generation consultancy, engineering and immersive learning

Values

- ❑ Honesty
- ❑ Pride
- ❑ Ownership
- ❑ Creativity
- ❑ Diversity

Principles

- ❑ People First
- ❑ Empathy
- ❑ Radical Candor
- ❑ Action Biased
- ❑ Experimentation
- ❑ Fast Feedback

How to use this sheet

This is not a checklist. These are your fallback defaults, if nothing else is set, not a golden master you must do.

Vision & Empathy

We understand the customer

The customer is as the centre

- ❑ Understand the customer
- ❑ Understand our stakeholders
- ❑ Shared view of how value flows to our customers
- ❑ Regular, early and direct feedback from the customer
- ❑ Understand the current domain, language & terminology

We have a clear vision

Highly aligned and focused

- ❑ Understand the broader business's vision
- ❑ We can all present a clear vision, mission and purpose
- ❑ We can measure what good looks like
- ❑ We have just enough of a shareable roadmap

Team Topology

Team can be successful

We can have sustainable success

- ❑ Delivery expectation doesn't exceed the team's cognitive capacity
- ❑ Roles, and role rotation are discussed and understood
- ❑ Working agreements negotiated with the whole team

Cloud Native

We start with production first

Production-ready from the outset

- ❑ Production deployed first
- ❑ Deployed Hello World on Production
- ❑ Deployed other environments

We continuously deploy

Pipelines!

- ❑ All deployments are pipelined
- ❑ Build fixed within 10 min of breaking
- ❑ All tests run and are reliable
- ❑ Coding standards are checked

We automate everything

Everything as Code

- ❑ Environments recreated from only code
- ❑ Environments regularly destroyed and created

We do day 2 operations on day 1

Day2Ops right from the start

- ❑ Logs are fully accessible to engineers
- ❑ Events are fully traceable to requests
- ❑ Service status is monitored
- ❑ Service Level Objectives set
- ❑ We can recover from disasters
- ❑ We can upgrade continuously

Code As Craft

Coding practice

Code well, code fast

- ❑ Commit to master daily
- ❑ Leave code better than you found it
- ❑ Pair program daily

We test everything

Automated tests

- ❑ All new functionality has automated tests
- ❑ All bug fixes have automated tests
- ❑ Non-functional requirements are verified with automated tests
- ❑ Test Driven Design is the default practice
- ❑ Tests give the confidence to deploy
- ❑ Failing tests fail the pipeline

Prototype Working Agreements:

Definition of Ready (DoR)

- ❑ The story discussed in an IPM
- ❑ All acceptance criteria defined
- ❑ The story is prioritised
- ❑ Team has estimated the story

Definition of Done (DoD)

- ❑ All tasks on the story are completed (and marked as such)
- ❑ All acceptance criteria completed
- ❑ All commits are tied to the story
- ❑ Story validation instructions included

XP & Agility

We set goals every quarter

Inception meetings

- ❑ Goals and Non-goals defined
- ❑ Actors identified
- ❑ Risks & mitigations identified

We work in weekly iterations

Iterations

- ❑ Started with IPM, restated goals
- ❑ Daily standups, terse and quick
- ❑ Each story completed in priority order
- ❑ Work in progress has limit of 1
- ❑ Pair program daily
- ❑ Ended with a demo & retrospective

We plan work as user stories

User stories

- ❑ Customer value is defined
- ❑ Stories sized to 0.5 days work
- ❑ Acceptance criteria are defined
- ❑ Prioritised with the Product Owner
- ❑ Accepted with the Product Owner
- ❑ Conform to Working Agreements
- ❑ Document just enough to proceed

We measure everything

Learn in a structure

- ❑ A Baseline metric is taken first
- ❑ Experiment is ran to change the metric
- ❑ Retrospect on if the experiment succeeded

Engagement

We enhance team enthusiasm

Growing engineer confidence

- ❑ Retro outcomes are actions
- ❑ Actions are carried out
- ❑ Rotate roles
- ❑ Everyone demos and talks
- ❑ See one, do one, teach one

We radiate our success

Visual working

- ❑ Display WIP Board/backlog
- ❑ Display deployment pipeline
- ❑ Display product metrics
- ❑ Show & tell product and WoW

We demonstrate client success

Show it's working

- ❑ The client can make simple investment decisions
- ❑ Regular, investment to value-focused reporting

We deliver client outputs

Safe & Secure delivery

- ❑ Ways of working are always secure and legal
- ❑ Golden source client artefacts live on client systems



Purpose

Return creativity and joy to the world of software engineering & delivery.

Vision

To be the world's most trusted partner for empowering businesses to accelerate the delivery of their greatest ideas

Mission

Businesses, at any scale, are awash with great ideas. However, the cost of learning which idea customers will love is too high. We evaporate this constraint and build quality in by accelerating cloud native adoption using next-generation consultancy, engineering and immersive learning

Values

- Honesty
- Pride
- Ownership
- Creativity
- Diversity

Principles

- People First
- Empathy
- Radical Candor
- Action Biased
- Experimentation
- Fast Feedback

How to use this sheet

This is not a checklist. These are your fallback defaults, if nothing else is set, not a golden master you must do.

Cloud Native

We start with production first

Production-ready from the outset

- Production deployed first
- Deployed Hello World on Production
- Deployed other environments

We continuously deploy

Pipelines!

- All deployments are pipelined
- Build fixed within 10 min of breaking
- All tests run and are reliable
- Coding standards are checked

We automate everything

Everything as Code

- All environments recreated from only code
- Environments regularly destroyed and created

We do day 2 on day 1

Consider day2ops right from the start

- Logs are fully accessible to engineers
- Events are fully traceable to requests
- Service status is monitored
- Service Level Objectives set

Code As Craft

Coding practice

Code well, code fast

- Commit to master
- Leave code better than you found it
- Pair program daily

We test everything

Automated tests

- All new functionality has automated tests
- All bug fixes have automated tests
- Non-functional requirements are verified with automated tests
- TDD is the default practice
- Tests give the confidence to deploy
- Failing tests fail the pipeline

Prototype Working Agreements

Definition of Ready (DoR)

- The story discussed in an IPM
- All acceptance criteria defined
- The story is prioritised
- Team has estimated the story

Definition of Done (DoD)

- All tasks on the story are completed (and marked as such)
- All acceptance criteria completed
- All commits are tied to the story
- Story validation instructions included

XP & Agility

We set goals every quarter

Inception meetings

- Define Goals and Non-goals
- Identify Actors
- Identify Risks & Mitigations

We work in weekly iterations

Iterations

- Start with IPM, restated goals
- Daily standups, terse and quick
- Complete each story in priority order
- WIP limit of 1
- Pair program daily
- End with a retrospective

We plan work as user stories

User stories

- Value is defined
- Stories sized to 0.5 day's work
- Acceptance criteria are defined
- Prioritised with the Product Owner
- Accepted with the Product Owner
- Conform to Working Agreements
- Document just enough to proceed

We measure everything

Learn in a structure

- A Baseline metric is taken first
- Experiment is ran to change the metric
- Retrospect on if the experiment succeeded

Engagement

We enhance client team enthusiasm

Growing engineer confidence

- Retro outcomes are actions
- Actions are carried out
- Rotate roles
- Everyone demos and talks
- See one, do one, teach one

We radiate our status

Visual working

- Display WIP Board/backlog
- Display deployment pipeline
- Display product metrics
- Show & tell product and WoW

We demonstrate client success

Show it's working

- Regular, value-focused reporting
- Know your client (empathy map)
- WoW always secure and legal
- Client artefacts live on client systems



Purpose

Return creativity and joy to the world of software engineering & delivery.

Vision

To be the world's most trusted partner for empowering businesses to accelerate the delivery of their greatest ideas

Mission

Businesses, at any scale, are awash with great ideas. However, the cost of learning which idea customers will love is too high. We evaporate this constraint and build quality in by accelerating cloud native adoption using next-generation consultancy, engineering and immersive learning

Values

- ☐ Honesty
- ☐ Pride
- ☐ Ownership
- ☐ Creativity
- ☐ Diversity

Principles

- ☐ People First
- ☐ Empathy
- ☐ Radical Candor
- ☐ Action Biased
- ☐ Experimentation
- ☐ Fast Feedback

How to use this sheet

This is not a checklist. These are your fallback defaults, if nothing else is set, not a golden master you must do.

Product team

We have a vision

We understand our domain

We understand the flow of work to customers

Our customers understand us

Own the product

- ☐ Empower the team
- ☐ Bridge business and implementation
- ☐ Co-locate with the team

We sell the product

Sales and marketing

- ☐ Actively talk to everybody
- ☐ Listen to our customers' needs
- ☐ Understand our customers' motivations
- ☐ Understand our suppliers

We share the vision

Vision is not just one person's view

- ☐ Driven by customer needs
- ☐ Shared by PO, team, customers
- ☐ Expressed as a product box
- ☐ Communicated

Product as a Service Tools / techniques

We make products fun to use

User-friendly products

- ☐ Make everything self-service
- ☐ Write standard APIs
- ☐ Write useful documentation, preferably so good it doesn't need documentation
- ☐ Create tested working examples
- ☐ Consumable with standard tools
- ☐ Radiate production status
- ☐ How can we tell when is our customer happy?

We sell the product

Sales and marketing

- ☐ Actively talk to everybody
- ☐ Listen to the customers' needs
- ☐ Understand the customers' motivations
- ☐ Understand the suppliers
- ☐ Radiate with show & tell
- ☐ Radiate with demonstrations
- ☐ Radiate the roadmap

Consider using some of these

- ☐ Perform domain modeling
- ☐ Event storming
- ☐ Context mappings
- ☐ Value stream mappings
- ☐ Empathy mapping

We get things done

No cruff

- ☐ Minimise and focus meetings and ceremonies
- ☐ No more than 2 weeks backlog
- ☐ Make time for tech debt

Product Owner

We build great products

Product ownership is a capability

- ☐ Build agree and publish the vision
- ☐ Understand our customers
- ☐ Customer experimentation - how do we show ideas to them quickly / customer feedback work
- ☐ Understand the flow of value to customers
- ☐ Understand our stakeholders
- ☐ Build a rapport with our suppliers
- ☐ Understand the business' vision
- ☐ How do we articulate what "good product" looks like, via trackable metrics
- ☐ Prioritise the backlog
- ☐
- ☐ Work toward psychological safety
- ☐ Empower team to make decisions
- ☐ Blameless post-mortems
- ☐
- ☐ Ensure team size meets cognitive load
- ☐ Demand empowerment
- ☐ Make meetings fun
- ☐

