zalando

Production Readiness Review in Zalando

DevOps Finland meetup 20.08.2024

Uri Savelchev

Agenda

1. What is Production Readiness Review?
2. What's inside?
3. Why do we need it?
4. Useful links
5. Q & A

What is Production Readiness Review?

The Google SRE book defines it as

... a process that identifies the reliability needs of a service based on its specific details. Through a PRR, SREs seek to apply what they've learned and experienced to ensure the reliability of a service operating in production. A PRR is considered a prerequisite for an SRE team to accept responsibility for managing the production aspects of a service.

Goal

Verify that a [new] service is production-ready in terms of

- Reliability
- Operations
- Data handling

Steps

1. The PRR document is created from a template and filled in by one of the team members and reviewed within the team.
2. A Principal Engineer *from another organization* makes a review and outlines their comments and questions.
3. A joint session with the PE and the team goes through the questions and finalizes an action list.
4. The completed PRR is kept and stay valid for two years or until architecture-level changes in the app.

What's inside?

[APPLICATION_NAME]

Production Readiness Review

Updated: 2024-07-16
Status: Draft
Engineering Team: __ENGINEERING_TEAM__
Contributors: __CONTRIBUTOR_A__, __CONTRIBUTOR_B__

This “production readiness review” (or “PRR”) document is a set of questions that leverage the experience gained by the Zalando tech community that guides you through the process of making an application “Production Ready”. It is designed to surface some of the more surprising challenges of running software in the unpredictable “production” environment.

You use this template by:

1. Cloning this template to a new document
2. Completing all of the questions to the best of your knowledge.
3. Having the answers reviewed by senior team members to leverage their experience to spot additional risks.
4. Collaborating with those team members to draft recommendations or action items for your service to improve its “production readiness”

Sections

- Context, Background and Production Operations
- Traffic Handling and Observability
- Engineering
- Data Management and ML Models
- Release process

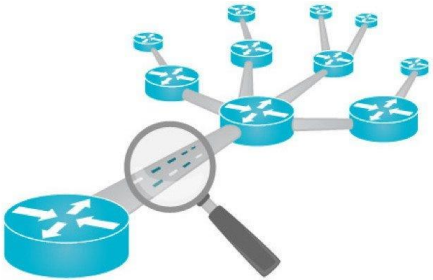
Context & Background

- What are the application's business functions?
- Who are the customers? What is expected SLA?
- Architecture diagrams, technical design document and other *technical* documents

Operations

- Downtime and failure impacts
- Is on-call required?
- Are necessary alerts and pages tested?
- Do all the on-call engineers have required access?

Traffic Handling

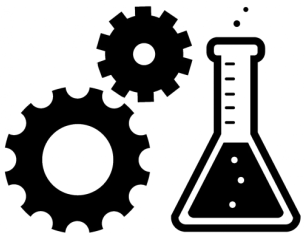


- Upstream traffic identification
- Rate limits (per upstream and global)
- Blocking bad traffic

Observability

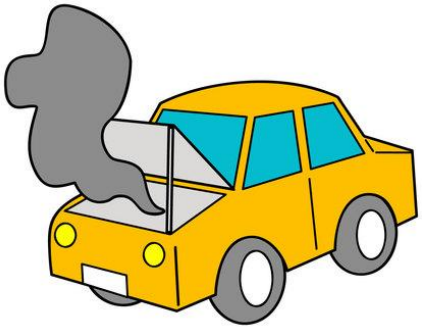
- Dashboards (golden signals, inbound/outbound streams)
- Data storages
- Dependencies / downstream monitoring
- Are the defined alerts and logged errors actionable?

Engineering



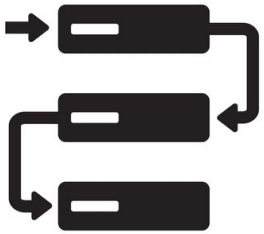
- Load testing and resource planning. Scaling.
- Deployment processes and timing
- Are all engineers in the team trained with the used technologies?

Failure Modes



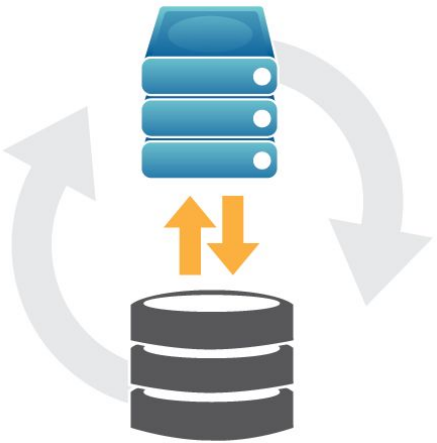
- What are the anticipated ways in which this application might fail?
- Are there single points of failure?
- Can the application be deployed successfully and then fail to start up?
- Does the application have timeouts for calls to its dependencies?
- Connection pools (to DBs and dependencies)
- Resilience patterns (retries, fallbacks, circuit breakers)

Dependencies



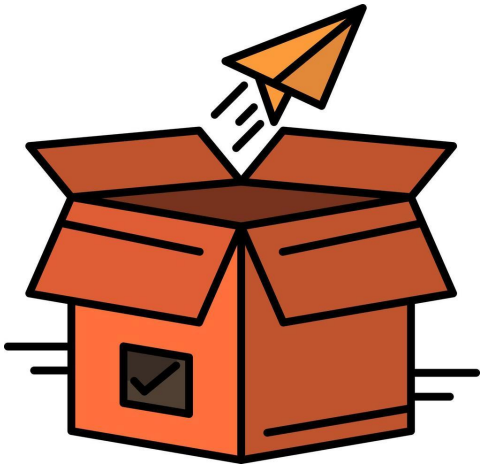
- List of dependencies and their SLOs
- Is this service’s SLO is more strict than the product of the service SLOs it depends on?
- Could a failure in a downstream cause the application to fail or respond with failures?
- Would scaling this application knock out a service it calls?

Data & ML



- Have data recovery scenarios been tested? How long do they take to execute?
- Can all data stores be upgraded without downtime?
- Can a single service node or process crash result in lost data?
- How often or when are the ML models updated?
- What approach is used to verify that a newly trained ML model is operating correctly?

Release



- Stakeholder management
- Upstream compatibility
- Rollout / rollback plan and criteria
- Data migration risks

Why do we need it?

To ensure operation excellence Zalando uses APEC checklist.

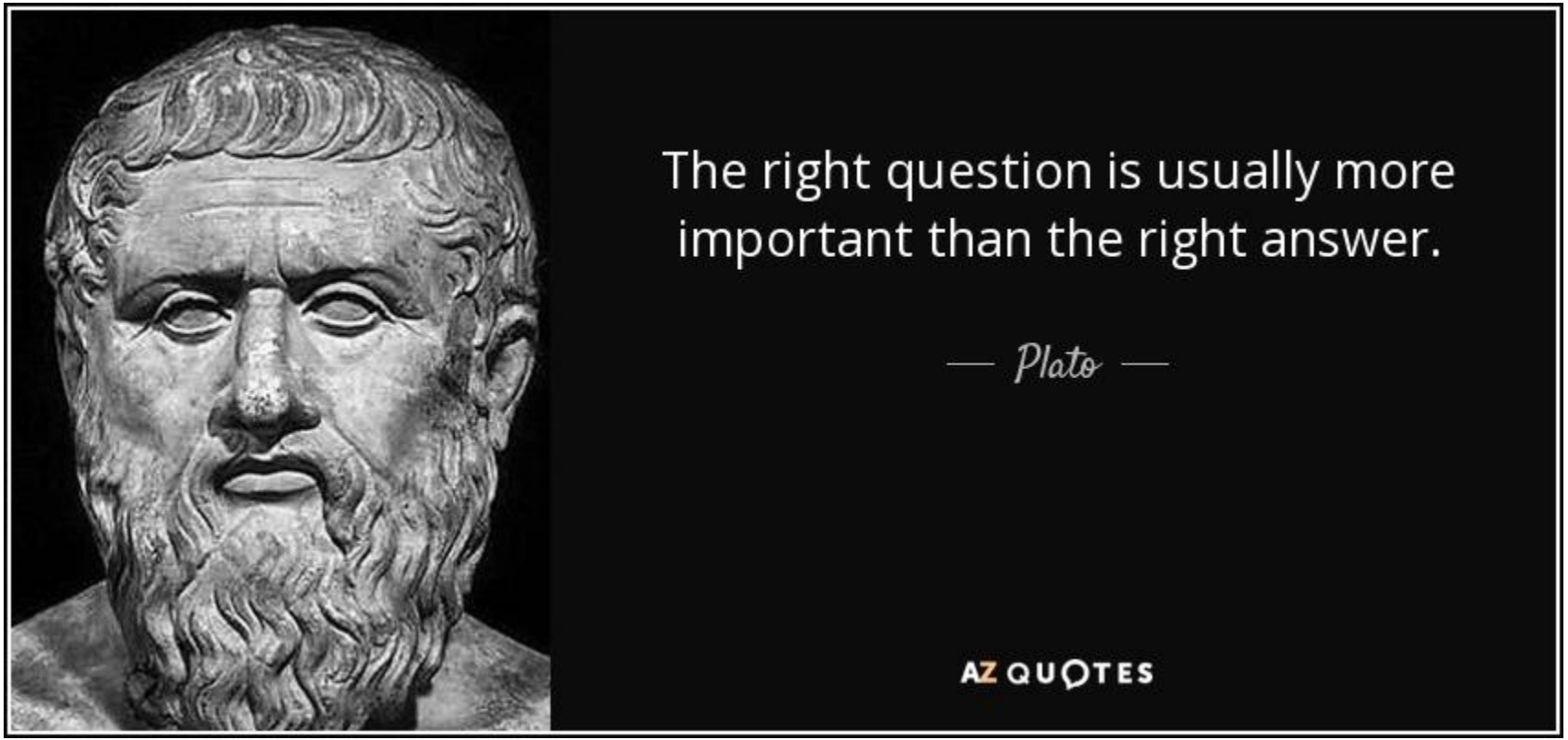
It identifies the most common problems, but for important applications we need a more deep-dive approach.

zalando

APEC: How to ensure your application is operated properly?

DevOps Finland meetup 10.10.2023

Uri Savelchev


The right question is usually more important than the right answer.
— Plato —
AZ QUOTES

Learn more

- James Cusick paper on architecture and production review
- AWS presentation on Production Readiness Review
- Zalando engineering blog


Q & A