Reliable by Design

Adding value in the design review process

A recipe by Laura Nolan (Slack Technologies)
Preparation time: 40 minutes
Served at SREcon APAC 2019

Twitter: @lauralifts

There are a lot of regretted decisions about software systems.

Image by eraphernalia_vintage@Flickr, CC BY-SA 2.0

Why write design documents or RFCs?

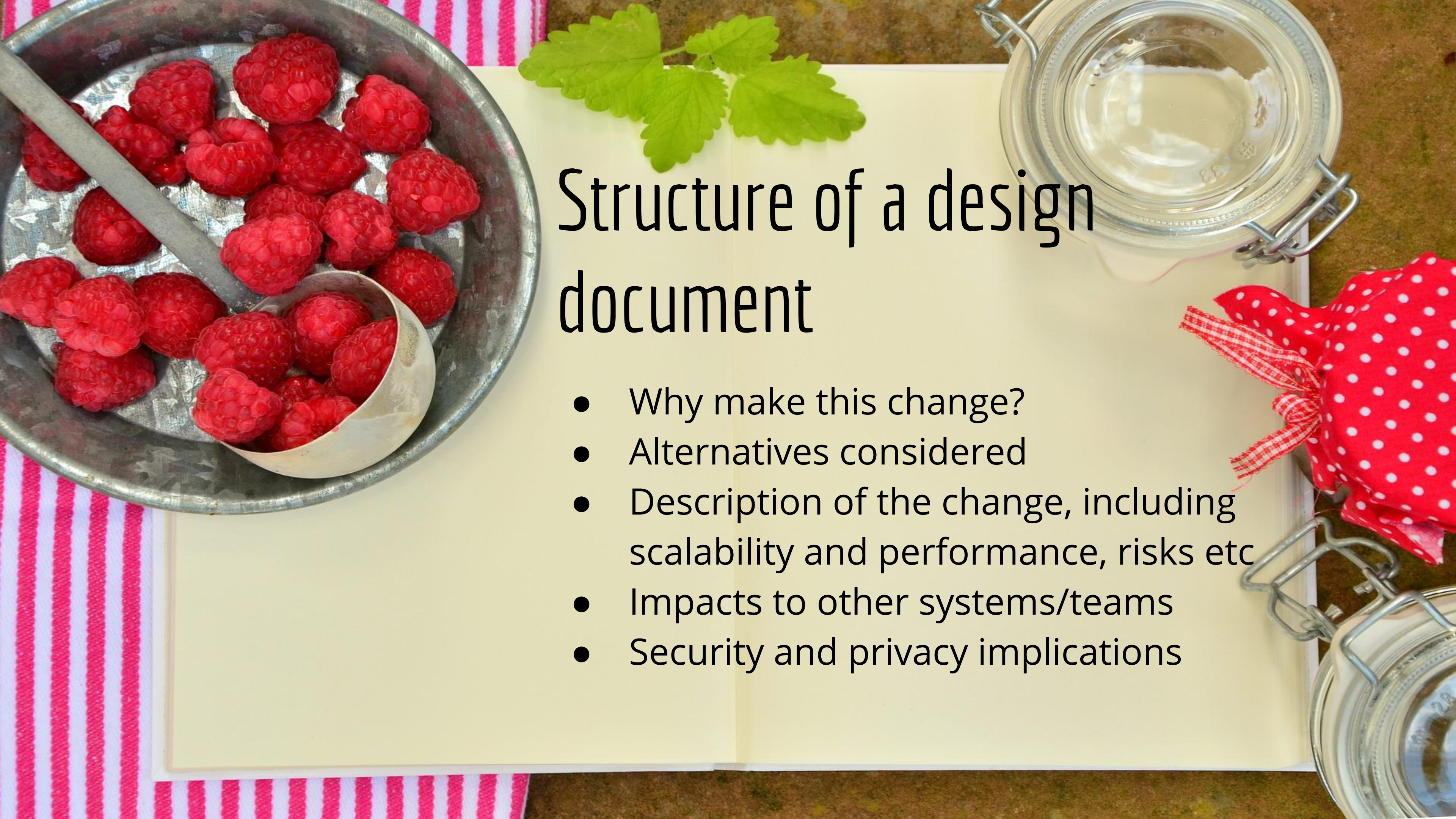
To promote shared understanding.

To spot problems early.

To give partner teams a voice.

Written designs are more powerful.

What kinds of changes need a written design?



Structure of a design document

- Why make this change?
- Alternatives considered
- Description of the change, including scalability and performance, risks etc
- Impacts to other systems/teams
- Security and privacy implications

Design review is **central** to SRE.

Design reviews can be time consuming and stressful.

It’s not about showing how clever you are.

It’s not about matters of taste.

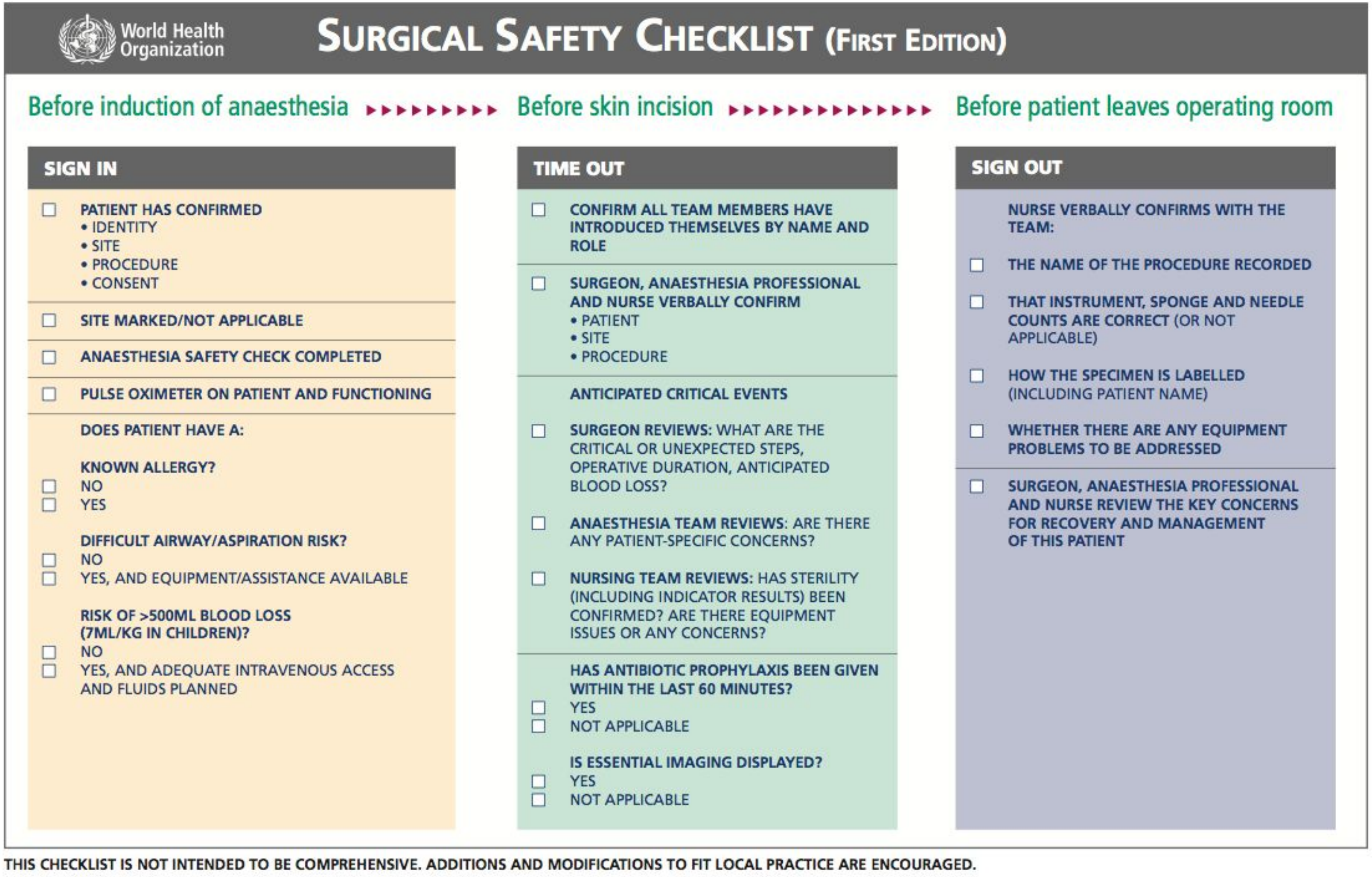
It’s about big, meaty concerns.

THE CHECKLIST MANIFESTO

HOW TO GET THINGS RIGHT

Two kinds of mistake

- Ignorance - there’s something we don’t know
- Ineptitude - we don’t make proper use of what we know



World Health Organization

SURGICAL SAFETY CHECKLIST (FIRST EDITION)

Before induction of anaesthesia ▸▸▸▸▸▸▸▸▸ Before skin incision ▸▸▸▸▸▸▸▸▸▸▸▸▸▸ Before patient leaves operating room

SIGN IN

☐ **PATIENT HAS CONFIRMED**
- IDENTITY
- SITE
- PROCEDURE
- CONSENT

☐ **SITE MARKED/NOT APPLICABLE**

☐ **ANAESTHESIA SAFETY CHECK COMPLETED**

☐ **PULSE OXIMETER ON PATIENT AND FUNCTIONING**

DOES PATIENT HAVE A:

KNOWN ALLERGY?
☐ NO
☐ YES

DIFFICULT AIRWAY/ASPIRATION RISK?
☐ NO
☐ YES, AND EQUIPMENT/ASSISTANCE AVAILABLE

RISK OF >500ML BLOOD LOSS (7ML/KG IN CHILDREN)?
☐ NO
☐ YES, AND ADEQUATE INTRAVENOUS ACCESS AND FLUIDS PLANNED

TIME OUT

☐ **CONFIRM ALL TEAM MEMBERS HAVE INTRODUCED THEMSELVES BY NAME AND ROLE**

☐ **SURGEON, ANAESTHESIA PROFESSIONAL AND NURSE VERBALLY CONFIRM**
- PATIENT
- SITE
- PROCEDURE

ANTICIPATED CRITICAL EVENTS

☐ **SURGEON REVIEWS:** WHAT ARE THE CRITICAL OR UNEXPECTED STEPS, OPERATIVE DURATION, ANTICIPATED BLOOD LOSS?

☐ **ANAESTHESIA TEAM REVIEWS:** ARE THERE ANY PATIENT-SPECIFIC CONCERNS?

☐ **NURSING TEAM REVIEWS:** HAS STERILITY (INCLUDING INDICATOR RESULTS) BEEN CONFIRMED? ARE THERE EQUIPMENT ISSUES OR ANY CONCERNS?

HAS ANTIBIOTIC PROPHYLAXIS BEEN GIVEN WITHIN THE LAST 60 MINUTES?
☐ YES
☐ NOT APPLICABLE

IS ESSENTIAL IMAGING DISPLAYED?
☐ YES
☐ NOT APPLICABLE

SIGN OUT

NURSE VERBALLY CONFIRMS WITH THE TEAM:

☐ **THE NAME OF THE PROCEDURE RECORDED**

☐ **THAT INSTRUMENT, SPONGE AND NEEDLE COUNTS ARE CORRECT** (OR NOT APPLICABLE)

☐ **HOW THE SPECIMEN IS LABELLED** (INCLUDING PATIENT NAME)

☐ **WHETHER THERE ARE ANY EQUIPMENT PROBLEMS TO BE ADDRESSED**

☐ **SURGEON, ANAESTHESIA PROFESSIONAL AND NURSE REVIEW THE KEY CONCERNS FOR RECOVERY AND MANAGEMENT OF THIS PATIENT**

THIS CHECKLIST IS NOT INTENDED TO BE COMPREHENSIVE. ADDITIONS AND MODIFICATIONS TO FIT LOCAL PRACTICE ARE ENCOURAGED.

I’m pretty sure this isn’t just ITIL for SREs.

Excellent process can set us free to be our best.

An SRE design checklist (v0.1)

- ❏ Who, what, and why?
- ❏ Alternatives considered
- ❏ Stickiness
- ❏ Data
- ❏ Complexity
- ❏ Scale and performance
- ❏ Operability
- ❏ Robustness

Who, what and why

- ❑ Do you understand the design?
- ❑ Do you understand the goal of the change?
- ❑ Why have you been asked to review?
- ❑ Have all affected teams been asked to review?
- ❑ Is specific privacy or security review needed?
- ❑ Who may be harmed by this system?

Alternatives considered

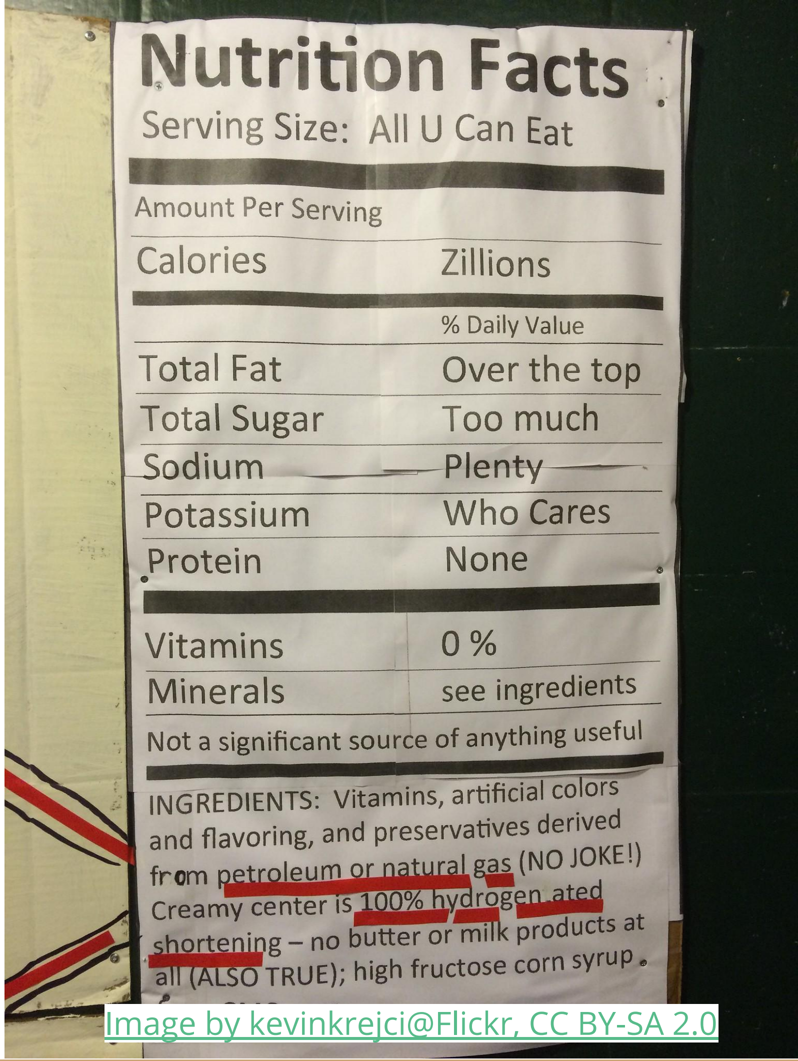
- ❏ Is there an open-source tool, or a similar proprietary system at this organisation that might work?
- ❏ If so, did the author of the design talk to owners of similar systems about this use-case?

Stickiness

- ❏ What is going to be hard to change later?
- ❏ What assumptions are baked into the architecture or the data model that might change in the future?

Data

- ❑ What is the flow of data through the system?
- ❑ What are the data consistency requirements and how does the design support them?
- ❑ What data can be recomputed from other sources and which cannot?



Nutrition Facts

Serving Size: All U Can Eat

Amount Per Serving

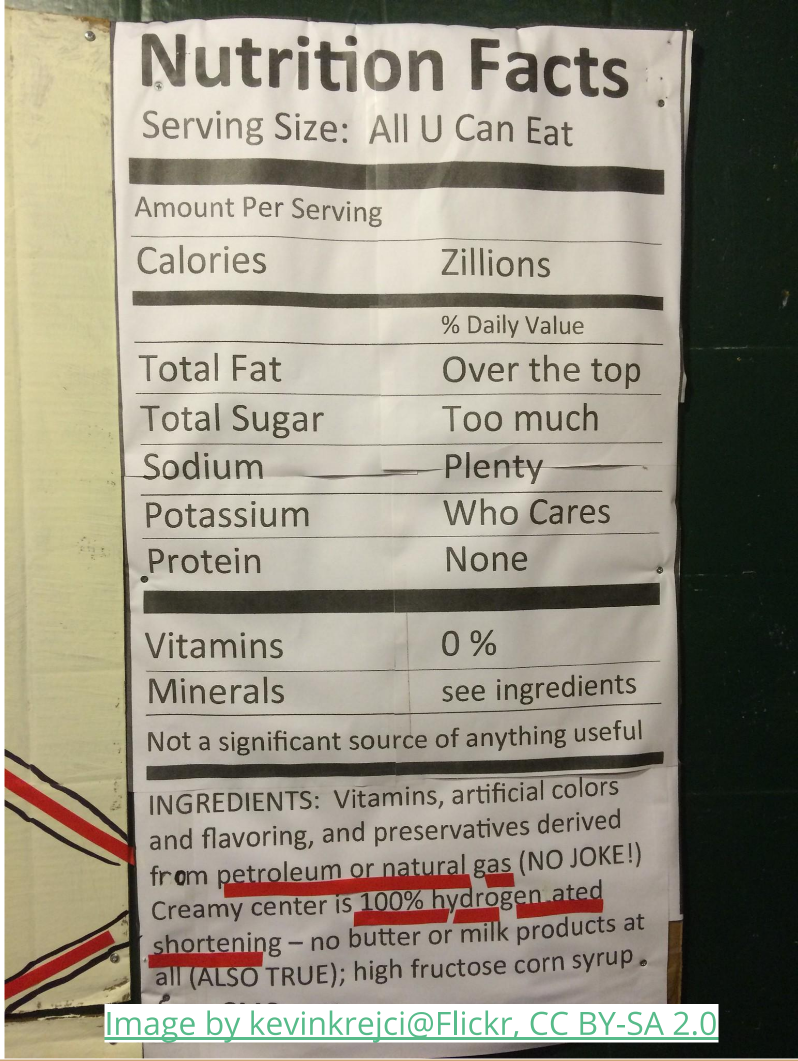
Calories	Zillions
	% Daily Value
Total Fat	Over the top
Total Sugar	Too much
Sodium	Plenty
Potassium	Who Cares
Protein	None
Vitamins	0 %
Minerals	see ingredients

Not a significant source of anything useful

INGREDIENTS: Vitamins, artificial colors and flavoring, and preservatives derived from petroleum or natural gas (NO JOKE!) Creamy center is 100% hydrogen ated shortening – no butter or milk products at all (ALSO TRUE); high fructose corn syrup

Data

- ❏ Is there a data loss Service Level Objective (SLO)?
- ❏ How long does data need to be retained, and why?
- ❏ Does it need to be encrypted at rest, in transit?
- ❏ Are there multiple replicas of the data?



Nutrition Facts

Serving Size: All U Can Eat

Amount Per Serving

Calories	Zillions
	% Daily Value
Total Fat	Over the top
Total Sugar	Too much
Sodium	Plenty
Potassium	Who Cares
Protein	None
Vitamins	0 %
Minerals	see ingredients

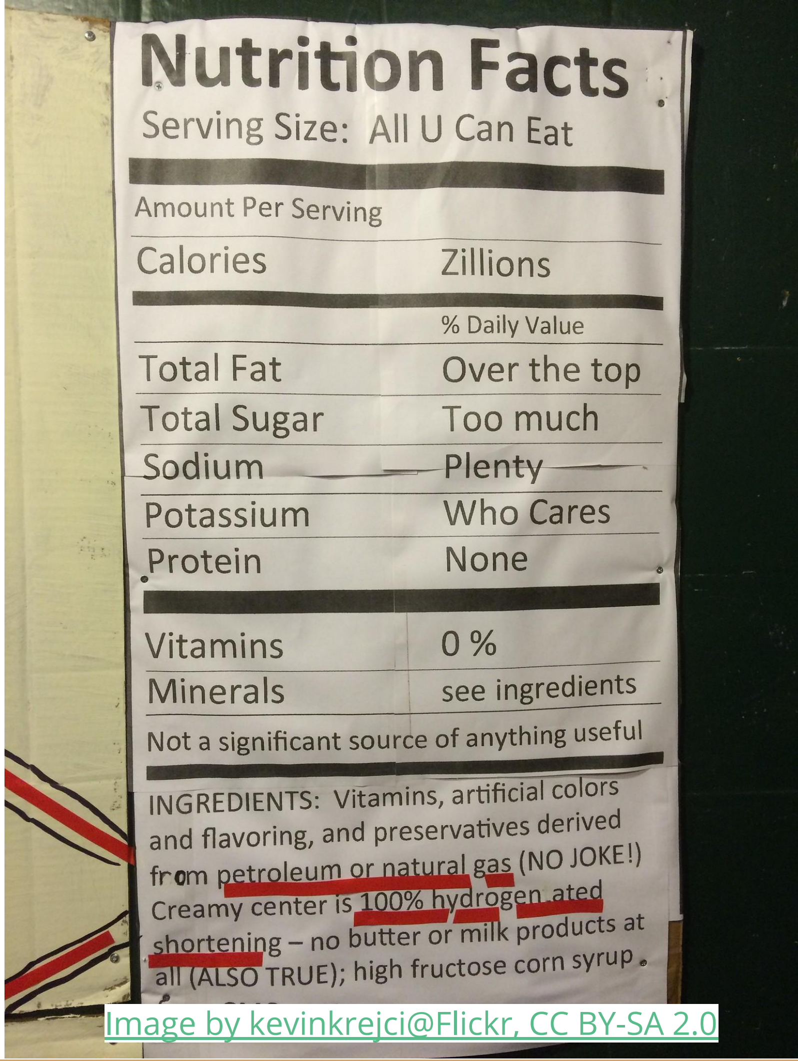
Not a significant source of anything useful

INGREDIENTS: Vitamins, artificial colors and flavoring, and preservatives derived from petroleum or natural gas (NO JOKE!) Creamy center is 100% hydrogen ated shortening – no butter or milk products at all (ALSO TRUE); high fructose corn syrup.

Image by kevinkrejci@Flickr, CC BY-SA 2.0

Data

- ❑ How do we detect and deal with loss or corruption of data?
- ❑ How is data sharded, and how do we deal with growth and resharding?
- ❑ How should data be backed up and restored?



Nutrition Facts

Serving Size: All U Can Eat

Amount Per Serving

Calories	Zillions
	% Daily Value
Total Fat	Over the top
Total Sugar	Too much
Sodium	Plenty
Potassium	Who Cares
Protein	None
Vitamins	0 %
Minerals	see ingredients

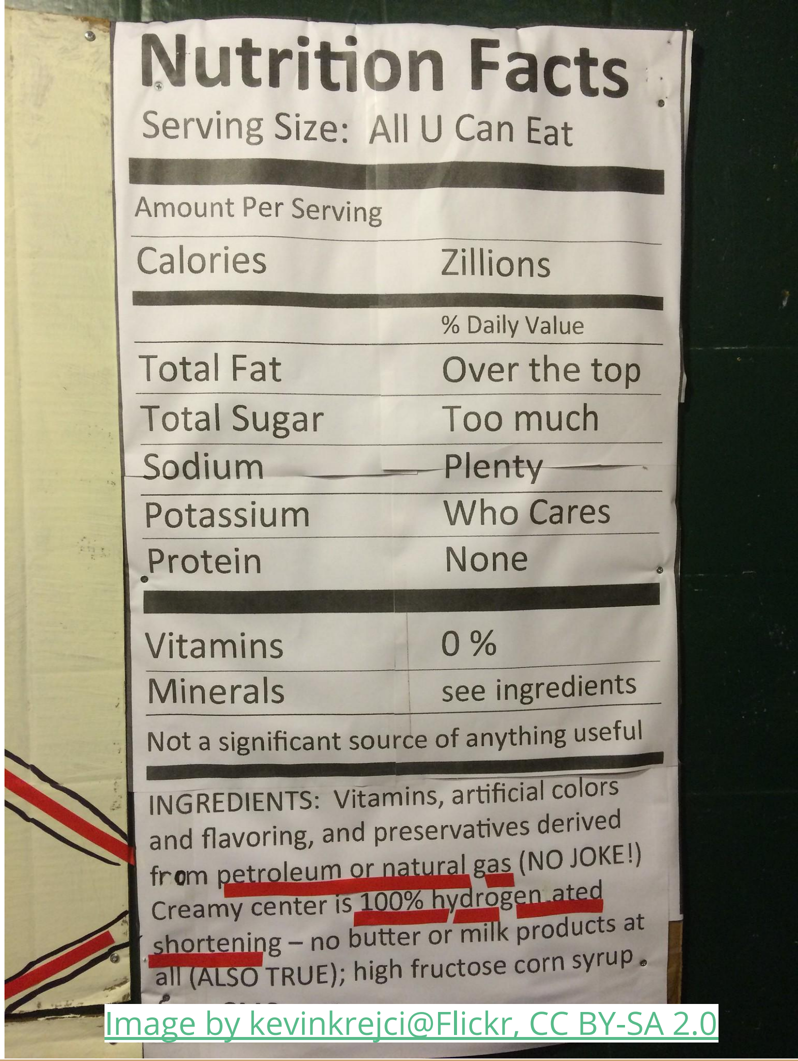
Not a significant source of anything useful

INGREDIENTS: Vitamins, artificial colors and flavoring, and preservatives derived from petroleum or natural gas (NO JOKE!) Creamy center is 100% hydrogen ated shortening – no butter or milk products at all (ALSO TRUE); high fructose corn syrup

Image by kevinkrejci@Flickr, CC BY-SA 2.0

Data

- ❏ What are the access control and authentication strategies?
- ❏ Have relevant regulations such as GDPR and any data residency requirements been addressed?



Nutrition Facts

Serving Size: All U Can Eat

Amount Per Serving

Calories	Zillions
	% Daily Value
Total Fat	Over the top
Total Sugar	Too much
Sodium	Plenty
Potassium	Who Cares
Protein	None
Vitamins	0 %
Minerals	see ingredients

Not a significant source of anything useful

INGREDIENTS: Vitamins, artificial colors and flavoring, and preservatives derived from petroleum or natural gas (NO JOKE!) Creamy center is 100% hydrogen ated shortening – no butter or milk products at all (ALSO TRUE); high fructose corn syrup.

Image by kevinkrejci@Flickr, CC BY-SA 2.0

Complexity

- ❑ Does each component of the system have a clearly defined role and a crisp interface?
- ❑ Can the number of moving parts be reduced?
- ❑ Is the design similar to existing systems at this organisation?
- ❑ Does the proposal introduce new dependencies?

Image by thecjm@Flickr, CC BY-SA 2.0

Scale and performance

- ❏ What are the bottlenecks in this system that will limit its scale and throughput?
- ❏ What's the critical path of each type of request, and how do requests fan-out into multiple sub-requests?

Image by aresauburnphotos@Flickr, CC BY-SA 2.0

Scale and performance

- ❏ What is the expected peak load and how does the system support it?
- ❏ What is the required latency SLO and how does the system support it?
- ❏ How will we capacity plan and loadtest?

Scale and performance

- ❏ What systems are we depending on, what are their performance limits and their documented SLOs?
- ❏ What will it cost to run financially?

Operability

- ❑ How does the design support monitoring and observability?
- ❑ Do all third-party system components provide appropriate observability features?

Image by davidspinks@Flickr, CC BY-SA 2.0

Operability

- ❏ What tools will be available to operators to understand and control the system’s behavior during production incidents?
- ❏ How will these tools make clear to the operator what specific actions they will take, to avoid surprises?

Image by davidspinks@Flickr, CC BY-SA 2.0

Operability

- ❑ What routine work is going to be needed for this system?
- ❑ Which team is expected to be responsible for it?
- ❑ How much of it can and should be automated, and will that automation reduce the operating team's understanding of the system?

Operability

- How do we detect abusive users or requests and what action can we take in response?
- how responsive will vendors be to your feature requests or problems?
- Are all configurations stored in source control?

Image by davidspinks@Flickr, CC BY-SA 2.0

Robustness

- ❏ How is the system designed to deal with failure in the various physical failure domains (device, rack, cluster/AZ, datacenter)?
- ❏ How will it deal with a network partition, or increased latency anywhere in the system?

Robustness

- ❏ Are there manual operations that will be required to recover from common kinds of failure?
- ❏ How could an operator accidentally (or deliberately) break the system?
- ❏ Is there isolation between users of the system?

Robustness

- ❑ What are the smallest divisible units of work and data, will we likely see hotspotting or large shards?
- ❑ What are the hard dependencies of this system, and can we degrade gracefully?

Robustness

- How can we restart this system from scratch and how long will that take?
- Do we depend on anything that might depend on this system?
    - Don’t forget DNS and monitoring.

Robustness

- ❑ How will this system deal with a large spike of load?
- ❑ Does the system use caching, and if so, will it be able to serve at increased latency without the cache?

Robustness

- ❏ Is the control plane fully separate from the data plane?
- ❏ Can I canary this design effectively?
- ❏ Can this system hurt its backends by making excessive requests?

Robustness

- ❑ Can this system autonomously drain capacity?
- ❑ Can this system autonomously initiate resource-intensive processes like large dataflows?
- ❑ Can this system create self-reinforcing phenomena (i.e. vicious cycles)?

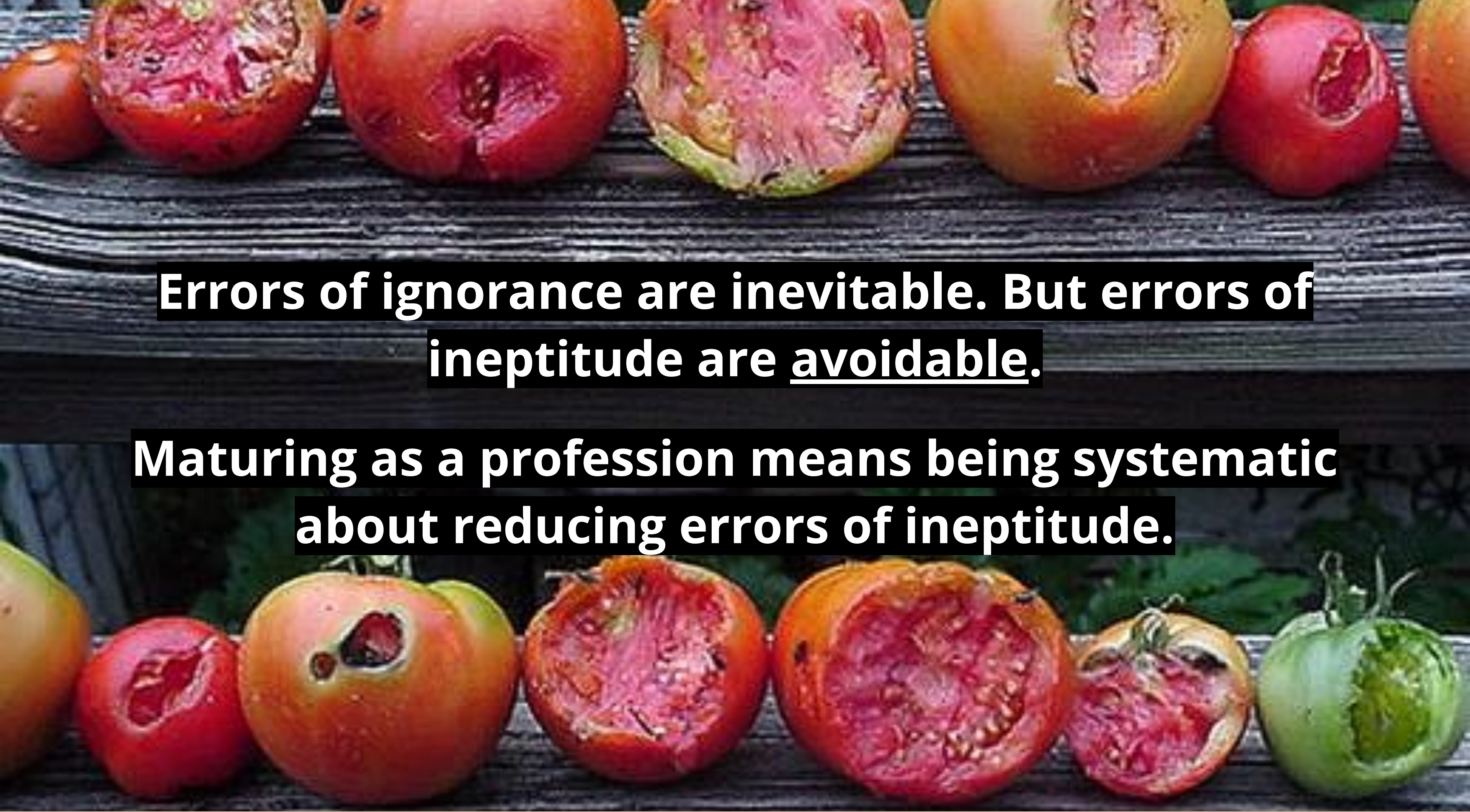
Checklist recap

- Who, what, and why
- Alternatives considered
- Stickiness
- Data
- Complexity
- Scale and performance
- Operability
- Robustness

Customize it

- This list should be adapted for local needs.
- Skip items that don’t make sense for a given project.

The goal of a good design is to understand tradeoffs and risks and to make deliberate choices.



Errors of ignorance are inevitable. But errors of ineptitude are avoidable.

Maturing as a profession means being systematic about reducing errors of ineptitude.

Find the checklist online

http://bit.ly/sredesign

We’re hiring!

slack

Slack is used by millions of people every day.
We need engineers who want to make that experience as reliable and enjoyable as possible.

https://slack.com/careers

Credits

Images are courtesy of pixabay.com, unless otherwise attributed on the slides where they appear.

“The Checklist Manifesto” by Atul Gawande.

Thank you to Tanya Reilly (@whereistanya) for reviewing the checklist.