2. Individual and Social Sustainability

Sustainable Software Engineering
CS4295

Luís Cruz
L.Cruz@tudelft.nl

1. Individual
2. Social
3. Assignment

Individual Sustainability

- *Well-being of the individuals in an organisation*

- It is difficult to **recruit** and **retain** good people
- Productivity requires **respecting** people
- It is important that responsibilities **reflect** their **skills** and **experience**.

What Software managers can do about individual sustainability?

Previous work shows that the **manager** is perceived as the most important factor in **job satisfaction**.

– Towards a Theory of Software Developer Job Satisfaction and Perceived Productivity (2019)

Relationship with the Manager

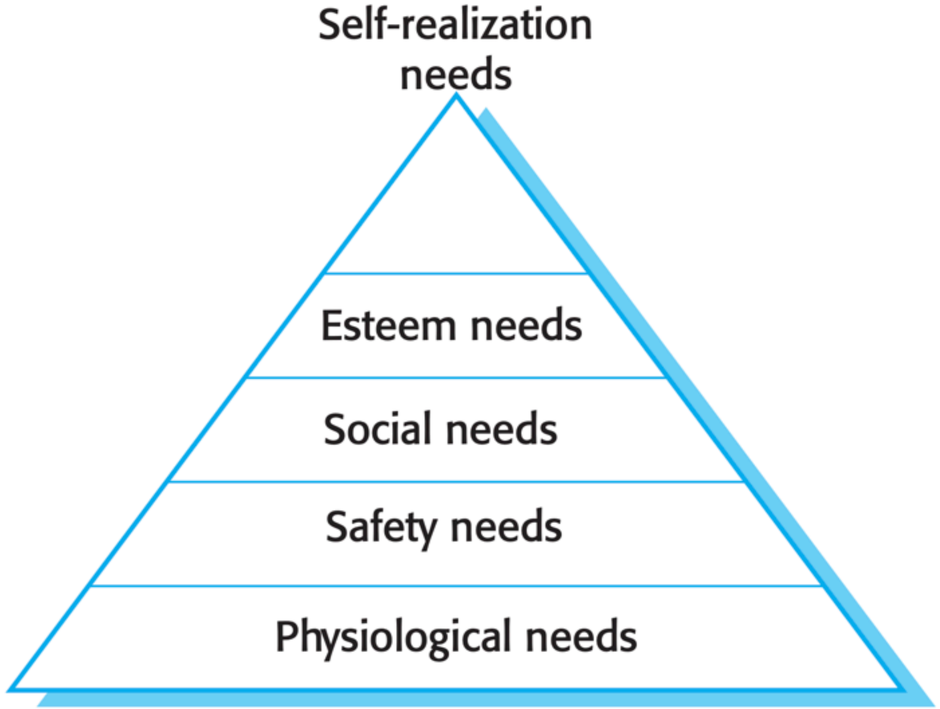
- **Consistency**. All people should be treated in a comparable way. Rewards are not identical but people should feel **appreciated for they work**.
- **Respect**. People have **different skills** and that's okay.
- **Inclusion**. People contribute effectively when they feel that others **listen** to them and **take account** of their **proposals**
- **Honesty**. Always be honest about what is going well and what is going badly in the team. E.g., be honest about your technical skill, be willing to **defer to staff with more knowledge** when necessary. If you try to cover it up, eventually you will lose the respect of the group.

Story

Motivating People

- If people are not motivated, they will work **slowly**, be more likely to make **mistakes**, and will not contribute to the **broader goals** of the team or organization.
- To encourage people we need to understand **what drives people** ➡ Maslow's pyramid

Maslow’s pyramid (1954)



Software Engineering, Ian Sommerville

Model that suggests that people are motivated by **satisfying their needs**.

Psychological personality type
(⚠️ model)

- Psychological personality type also influences motivation.
- **Task-oriented people**. Motivated by the intellectual challenge.
- **Self-oriented people**. Motivated by personal success and recognition.
- **Interaction-oriented people**. Motivated by the interaction between co-workers. (Usually great in soft skills)
- All types are important. Teams are typically more successful when all these types can work together. Getting the perfect balance is tricky though.

?

Work in teams (I)

- Individuals can only excel if we consider the team, department, organisation where they work.
- Ideally, teams should be “designed” in a way that unlocks everyone’s full potential.
  - Right balance of **skills**, **experience**, and **personalities**.
  - Technical knowledge should not be the only factor. The “competing engineers” problem.
  - Importance of **interaction-oriented teammates** (**within reason**).

Work in teams (II)

- Good managers strive for **cohesive groups**.
  - Establish a sense of **group identity** (e.g., team name, promote social events, etc.)
  - Promote **inclusion**. Have everyone involved in most project-related updates and discussions.
  - Well-defined **processes** and **responsibilities**.
  - **Good communication**. Horizontal and vertical communications should be straightforwards.

Communication

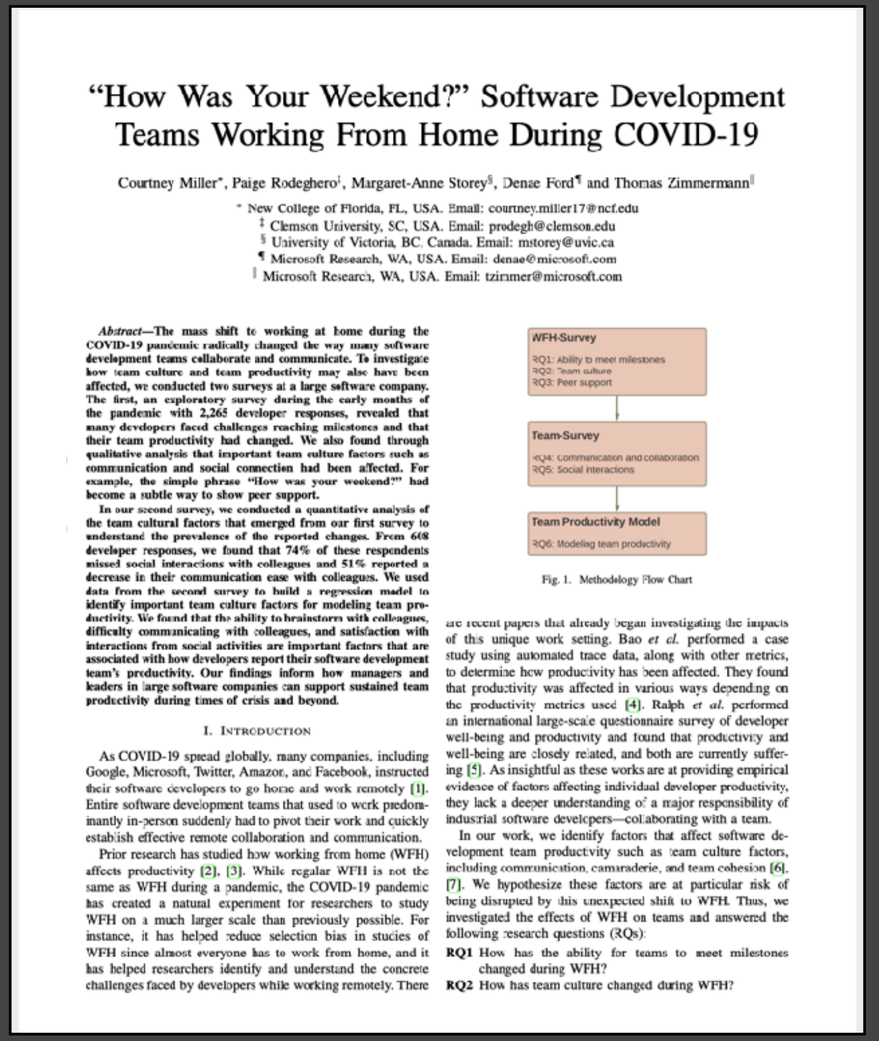
- Communication is quintessential!
- **Group size matters**. (experts say 4–6 is optimal)
  - E.g., teams size 8 will have 8x(8-1) = 56 possible communication pathways.
  - => Some people will **rarely communicate** with each other.
  - This is a problem:
    - Managers and experienced engineers tend to **dominate conversations**
    - Juniors might be reluctant to make **critical remarks** or **initiate conversations**.

Working space

- Developers with optimal workplace can **boost their productivity by 2x.** (DeMarco and Lister, 1985)
  - Ability to cut off interruptions
  - Being able to choose between a private space and an open space, depending on the task. (A bit expensive though 🤑) .
- **Flex desks**.
  - + Occupancy cost savings above 5%.
  - + You can easily book private space or shared space
  - + Perfect if you are mobile/remote and rarely at the office.
  - + Flattens the workplace hierarchy and encourages networking at all levels (does it?)
  - - Potentially hindering team and individual identity

Individual Sustainability during the pandemic

- Survey with 600+ developers
- Main takeaway:
  Pandemic remote work isn't remote work.
- *"We have lost somewhere between* **20%-40% effectiveness in use of time***. In order to keep up,* **people are working longer hours***. We are starting to see* **burnout***."*
- **A few proposed practices:**
  - Build and maintain **team culture**.
  - Include **social activities** as part of "work." (Coffee break failure example)
  - Be mindful of other **people's time**. (30-minute meeting policy)
  - Actively work to be **inclusive**.



"How Was Your Weekend?" Software Development Teams Working From Home During COVID-19

Courtney Miller*, Paige Rodeghero‡, Margaret-Anne Storey§, Denae Ford¶ and Thomas Zimmermann‖
* New College of Florida, FL, USA. Email: courtney.miller17@ncf.edu
‡ Clemson University, SC, USA. Email: prodegh@clemson.edu
§ University of Victoria, BC, Canada. Email: mstorey@uvic.ca
¶ Microsoft Research, WA, USA. Email: denae@microsoft.com
‖ Microsoft Research, WA, USA. Email: tzimmer@microsoft.com

***Abstract*—The mass shift to working at home during the COVID-19 pandemic radically changed the way many software development teams collaborate and communicate. To investigate how team culture and team productivity may also have been affected, we conducted two surveys at a large software company. The first, an exploratory survey during the early months of the pandemic with 2,265 developer responses, revealed that many developers faced challenges reaching milestones and that their team productivity had changed. We also found through qualitative analysis that important team culture factors such as communication and social connection had been affected. For example, the simple phrase "How was your weekend?" had become a subtle way to show peer support.**

In our second survey, we conducted a quantitative analysis of the team cultural factors that emerged from our first survey to understand the prevalence of the reported changes. From 608 developer responses, we found that 74% of these respondents missed social interactions with colleagues and 51% reported a decrease in their communication ease with colleagues. We used data from the second survey to build a regression model to identify important team culture factors for modeling team productivity. We found that the ability to brainstorm with colleagues, difficulty communicating with colleagues, and satisfaction with interactions from social activities are important factors that are associated with how developers report their software development team's productivity. Our findings inform how managers and leaders in large software companies can support sustained team productivity during times of crisis and beyond.

Fig. 1. Methodology Flow Chart

I. Introduction

As COVID-19 spread globally, many companies, including Google, Microsoft, Twitter, Amazon, and Facebook, instructed their software developers to go home and work remotely [1]. Entire software development teams that used to work predominantly in-person suddenly had to pivot their work and quickly establish effective remote collaboration and communication.

Prior research has studied how working from home (WFH) affects productivity [2], [3]. While regular WFH is not the same as WFH during a pandemic, the COVID-19 pandemic has created a natural experiment for researchers to study WFH on a much larger scale than previously possible. For instance, it has helped reduce selection bias in studies of WFH since almost everyone has to work from home, and it has helped researchers identify and understand the concrete challenges faced by developers while working remotely. There are recent papers that already began investigating the impacts of this unique work setting. Bao *et al.* performed a case study using automated trace data, along with other metrics, to determine how productivity has been affected. They found that productivity was affected in various ways depending on the productivity metrics used [4]. Ralph *et al.* performed an international large-scale questionnaire survey of developer well-being and productivity and found that productivity and well-being are closely related, and both are currently suffering [5]. As insightful as these works are at providing empirical evidence of factors affecting individual developer productivity, they lack a deeper understanding of a major responsibility of industrial software developers—collaborating with a team.

In our work, we identify factors that affect software development team productivity such as team culture factors, including communication, camaraderie, and team cohesion [6], [7]. We hypothesize these factors are at particular risk of being disrupted by this unexpected shift to WFH. Thus, we investigated the effects of WFH on teams and answered the following research questions (RQs):

RQ1 How has the ability for teams to meet milestones changed during WFH?
RQ2 How has team culture changed during WFH?

(Miller et al., 2021)
Preprint: https://arxiv.org/pdf/2101.05877.pdf

Mental Health/Wellbeing

- Mental Health Support
- Campaigns about mental wellness
- Mental health day off (e.g., Google)
- A lot to be done yet…
- **Mental Health ≠ Mental Wellbeing**

Social Sustainability

Values

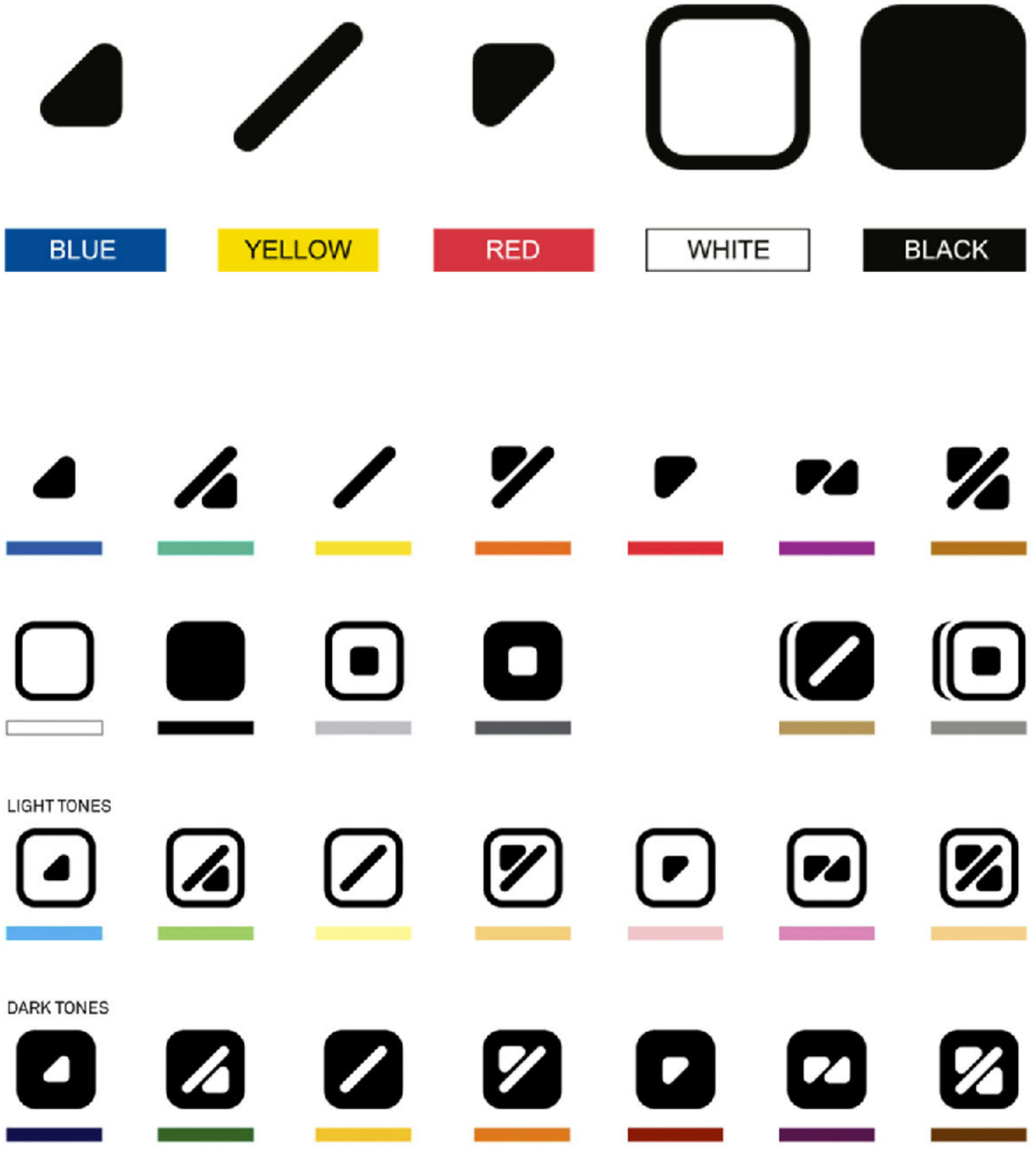
- Democracy and Justice
- **Human Wellfare**
- Inclusiveness
- Presence
- **Privacy**
- **Security**
- **Ownership**
- Regulation
- Responsibility
- Safety
- Trust
- Accountability and Transparency
- …

Inclusiveness

- Accessibility features
  - Color-blind users
- Elderly people
- People without digital presence
- …
- Gender, Culture, Age, Backgrounds…

ColorADD

The Color Alphabet



Gender-biased AI models

- Amazon’s recruiting system based on AI models.
- Automatically processing CV’s to do a first-stage screening
- Feedback loops: the training data contains **unwanted biases**. I.e., the software was amplifying the male dominance across the tech industry.
- (Fixed in 2015)
- **How can we prevent such biases?**

Inclusive Language in Technology

- Replace potentially offensive language in code bases and documentation
- General Guidelines When Writing Code or Documentation **(based on Academy Software Foundation)**
  - 1. Avoid using terms that have social history.
  - 2. Avoid using idioms and jargons.
  - 3. Write inclusive examples. (Avoid culturally-specific examples)
  - 4. If you’re unsure, ask.

Common terms and replacements (I)

Inclusive Language in Technology

- **Socially-charged language**
  - Master, slave → primary/main, secondary/replica
  - Owner, master → lead, manager, expert
  - Blacklist → deny list, exclusion list, block list, banned list
  - Whitelist → allow list, inclusion list, safe list
  - Native feature → core feature, built-in feature
  - Culture fit → values fit
  - Housekeeping → cleanup, maintenance

Common terms and replacements (II)

Inclusive Language in Technology

- **Gendered Language**
  - Man hours → labor hours, work hours
  - Manpower → labor, workforce
  - Guys (referring to a group) → folks, people, engineers/artists
  - Girl/Girls (referring to women age 18 and older) → woman/women
  - Middleman → middle person, mediator, liaison
  - Gendered pronouns (he/him/his, she/her/hers) → they, them, theirs

Common terms and replacements (III)

Inclusive Language in Technology

- **Ableist Language** (superiority)
  - Crazy, insane → unpredictable, unexpected
  - Normal → typical
  - Abnormal → atypical

Common terms and replacements (IV)

Inclusive Language in Technology

- **Ageist Language**
  - Grandfather, grandfathering, legacy → flagship, established, rollover, carryover
- **Violent language**
  - Crushing it, killing it → elevating, exceeding expectations, excelling

?

Examples

Inclusive Language in Technology

- Github (master -> main)
- Linux kernel’s official coding-style
- Apple’s official coding style: https://help.apple.com/applestyleguide (whole chapter on inclusive writing)
- Twitter Engineering

Twitter Engineering @TwitterEng · Jul 2, 2020

Inclusive language plays a critical role in fostering an environment where everyone belongs. At Twitter, the language we have been using in our code does not reflect our values as a c
We want to chan

> **regynald** @
> Twitter eng rec
> language in ou
> want to speak
> so far.
>
> cnet.com/news

Twitter Enginee
@TwitterEng

We're starting with
away from using in
such as:

Avoid non-inclusive language	⇒	Prefer inclusive versions
Whitelist	⇒	Allowlist
Blacklist	⇒	Denylist
Master/slave	⇒	Leader/follower, primary/replica, primary/standby
Grandfathered	⇒	Legacy status
Gendered pronouns (e.g. guys)	⇒	Folks, people, you all, y'all
Gendered pronouns (e.g. he/him/his)	⇒	They, them, their
Man hours	⇒	Person hours, engineer hours
Sanity check	⇒	Quick check, confidence check, coherence check
Dummy value	⇒	Placeholder value, sample value

6:52 PM · Jul 2, 2020

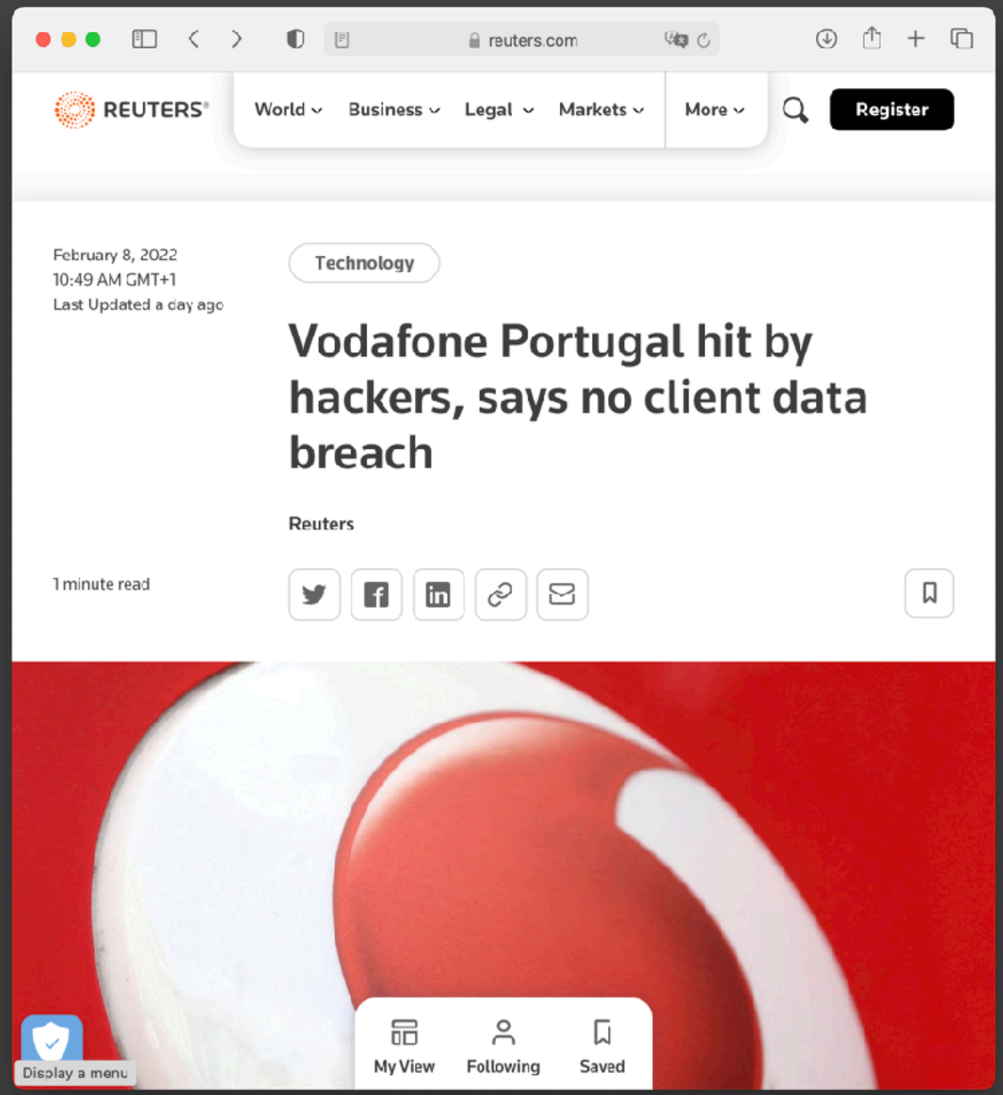
https://twitter.com/TwitterEng/status/1278733305190342656

Privacy

- Considered a basic human right (Movius and Krup, 2009)
- Linking personal information is easy with the increasing computing power, data storage, and ubiquitous collection of personal data.
- 3 main aspects
  - (**1**) being left alone and **free from intrusion**, (**2**) being able to **control personal information** about oneself, and (**3**) not being tracked, followed, or watched in **private space**.
- The line that divides public from private space is a bit blurry. (E.g., do users perceive their Facebook page as private/public?

Security

- Particularly important when your software company/ product stores **sensitive data**.
- People store data in the cloud without being aware of the security of their systems. How to make sure people are not harmed by security attacks?
- This is a value with **enough investment**. There's a huge cost when security is not handled
- **Vodafone PT** was shut down for a few hours this week (Feb 8 2022) due to a **cyber attack**
  - They claim **no personal data had been compromised**… **-trust, transparency..-**



Human Welfare

- **Physical**
  - Important for healthcare software, self-driving cars.
- **Material**
  - Important for banking systems, self-driving cars
- **Psychological**
  - E.g., cyberbulling, catfishing, etc.

Ownership

- Easy to understand when it comes to physical objects
- Digital data can be **saved**, **shared**, **published**, **removed**, etc.
- When you upload media content, do you understand who owns that content?
- Instagram, Facebook, Tiktok, Dropbox, Flickr, Google Photos, etc.

Dropbox case

Ownership

- *“you grant us (and those we work with to provide the Services) worldwide, non-exclusive, royalty-free, sublicenseable rights to use, copy, distribute, prepare derivative works (such as translations or format conversions) of, perform, or publicly display that stuff to the extent we think it necessary for the Service.” – Dropbox Terms of Service, July 2 2011*
- In other words: **We own your stuff**🤡
- *“You must ensure you have the rights you need to grant us that permission.”*
- In other words, if you upload something you **legally bought for personal use**, you are giving dropbox permission to distribute it worldwide, **and it’s your fault**🤡
- (One week later Dropbox updated their ToS; they were really quick to fix the issue and regain trust)

How do we include these values in the software lifecycle?

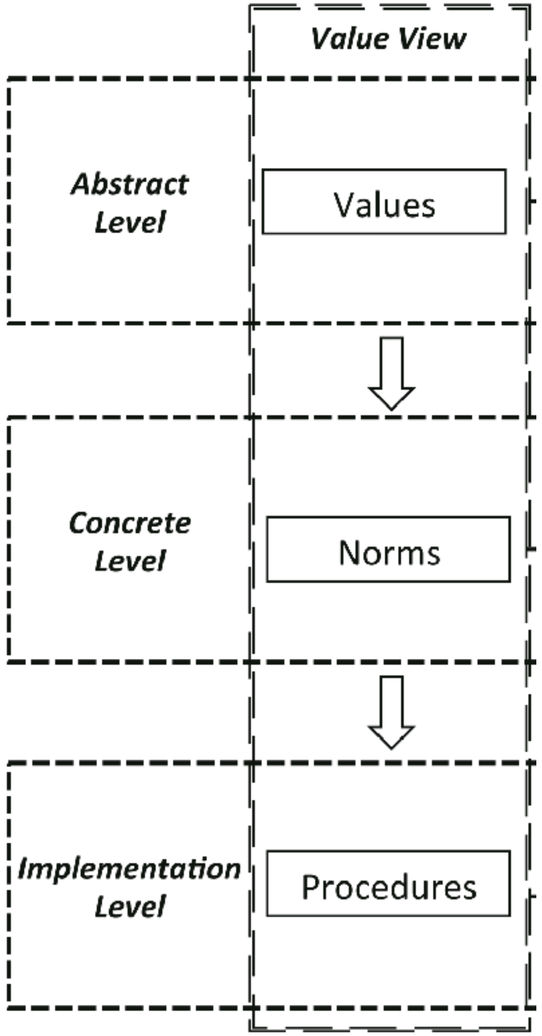
(Software) Design for Values

- **Global aims** and **policies** should be explicitly defined.
- Enforcement is context based and should be negotiated between **stakeholders**.
- **Design decisions** should be **formulated explicitly** rather than being implicit in the procedures and objects.
- We need a way of translating **abstract level values** into **verifiable specifications** (I.e., natural language to formal language).
  - Still an open problem.
  - There are a few frameworks out there.

VSSD

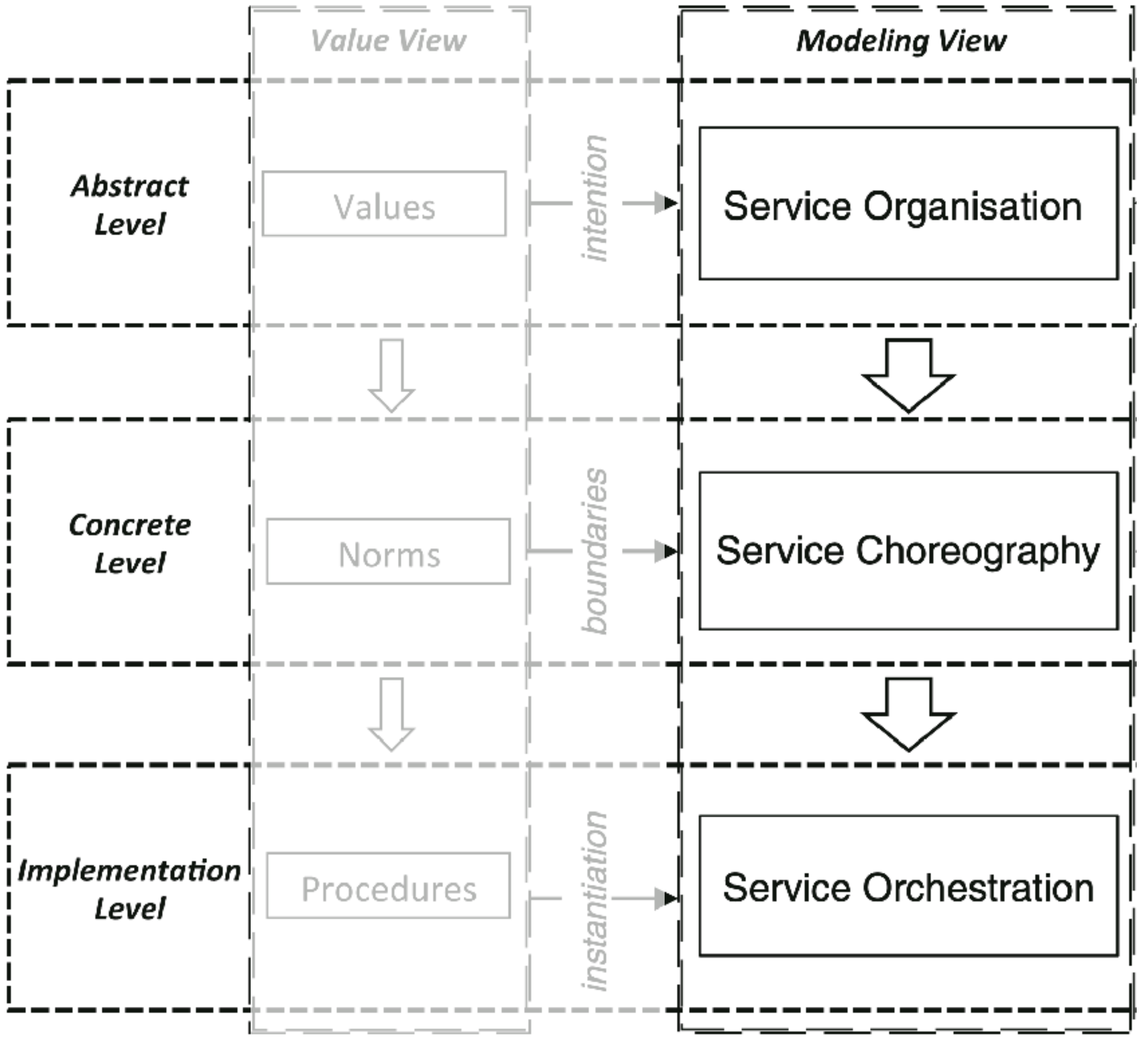
- Value-Sensitive Software Development Framework (VSSD)
- By Huib Aldewereld and Virginia Dignum (2014)
- From TU Delft
- Particularly useful when you **port your software to different platforms** (Web, iOS, Android, etc.). Keeping the same values might be tricky.

VSSD



Focused on Values definitions and their operationalisation

VSSD

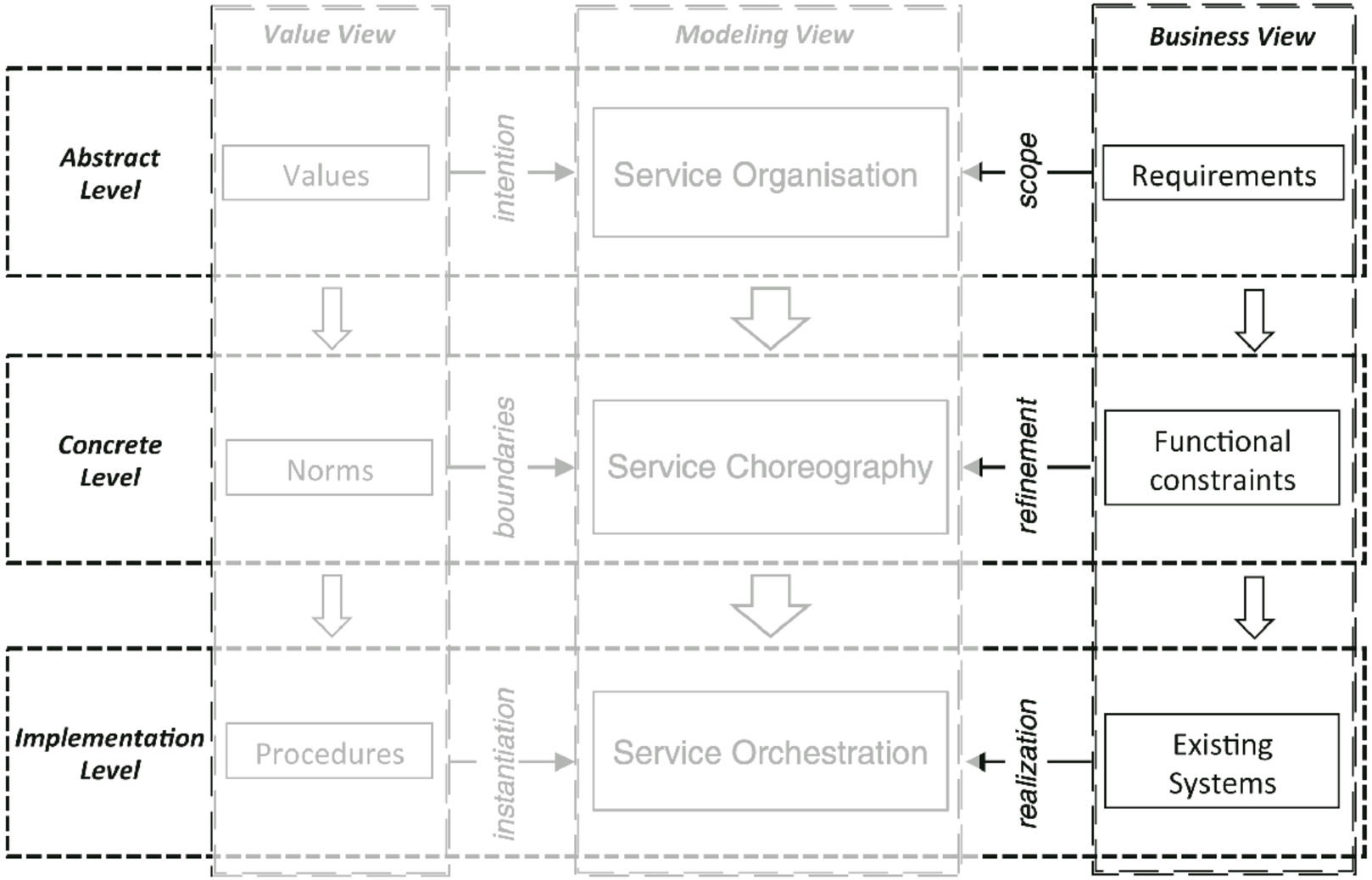


The system architecture

VSSD

No concrete specification of software architecture/ component

Concrete model components are specified



Contextual aspects of the domain

?

Thoughts about VSSD?

Open questions

- If you are a task-/self-/interaction-oriented person, is VSSD motivating?
- How do you fit VSSD into 2-week sprints and a fast pace software engineering team?

Other methods/frameworks to approach values

- Stakeholder Analysis (direct and indirect stakeholders)
- Value Elicitation
- Value Scenarios
- Envisioning Cards and Security Cards ->

Other social-centred subfields in SE/ICT
(Leaving it here for future reference)

- Computer Ethics
- Social informatics
- Computer-Supported Collaborative Work
- Participatory Design and Co-design
- Values in Design
- Values at Play
- Value-Sensitive Design
- Privacy by Design
- Worth-Centered Design

Assignment

Literature Review 📚

- Groups of **4** (exceptionally **3)**
- Select **1 academic paper** that talks about social or individual sustainability in the context of software engineering. (I will also accept economical sustainability.)
  - Google Scholar; DBLP
  - Look for forward and backward references of the papers mentioned in the class.
  - **Double check** with the lecturer whether your paper is good enough.
- Write a **short summary** about it (min 200 words; max 500).
- Prepare a **presentation** for next Tuesday (Feb 21).
  - 5 min + 1 min clarification question.
    - Presentation should **emphasise** and **discuss takeaways** of the paper.
- Presentation is 7.5% of the grade. (But you can recover in the presentation of project 2)
- Submission by pull request to the website. Instructions will follow today.
- **Deadline Tuesday, Feb 21**.

Further Reading

- **Software Engineering** by Ian Sommerville.
- Jeroen van den Hoven, Pieter E. Vermaas, Ibo van de Poel (2015). **Handbook of Ethics, Values, and Technological Design**. Springer. https://doi.org/10.1007/978-94-007-6970-0
- Courtney Miller, Paige Rodeghero, Margaret-Anne Storey, Denae Ford and Thomas Zimmermann (2021). **“How Was Your Weekend?” Software Development Teams Working From Home During COVID-19**. ICSE
- Movius LB, Krup N (2009) **US and EU privacy policy: comparison of regulatory approaches**. Int J Commun 3:19