



Fabian Gilson

Software Engineering II

SENG301 - Requirement analysis

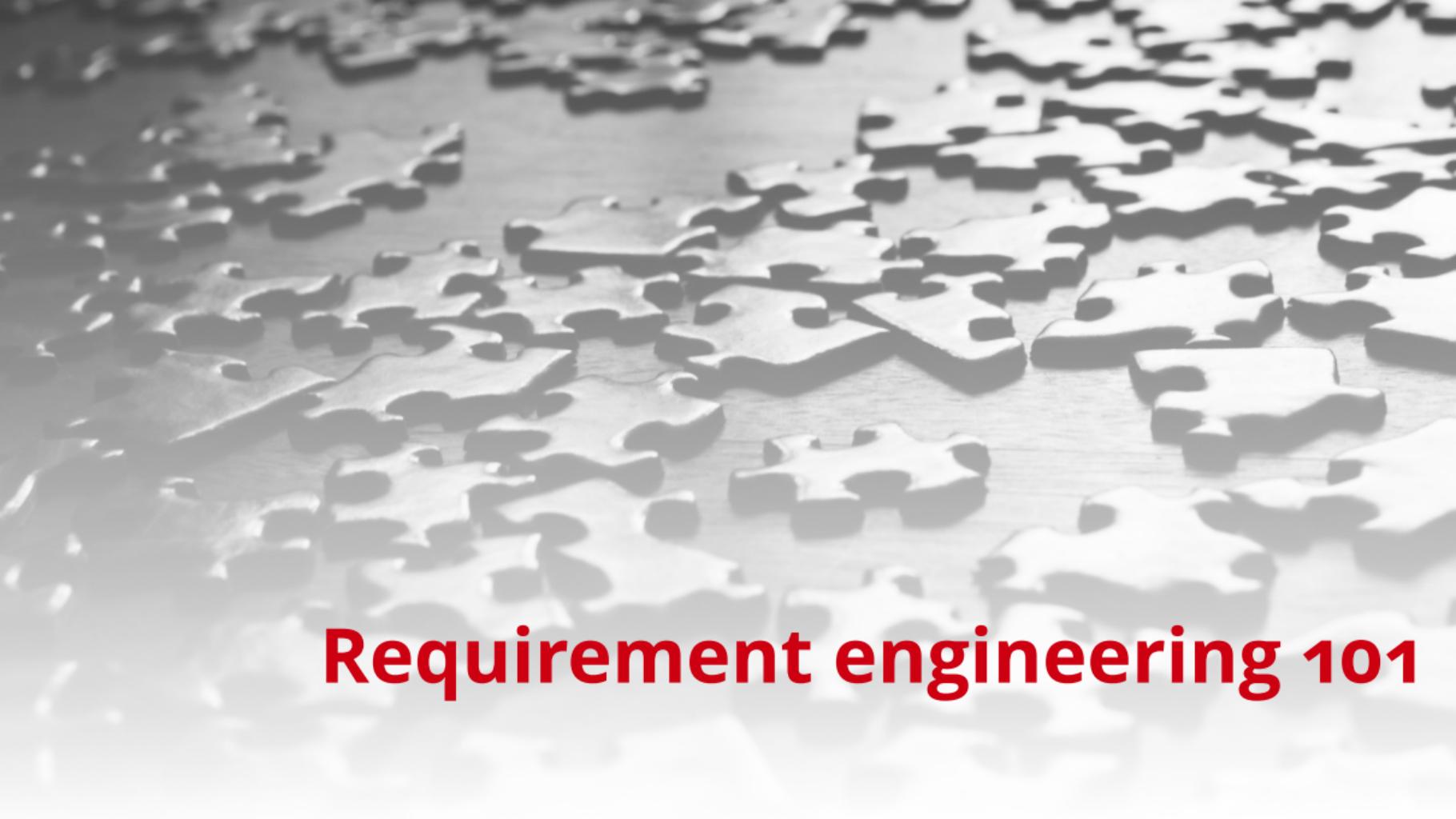
Lecture 4 - Semester 1, 2020



A primary cause of complexity is
that software vendors uncritically
adopt almost any feature that
users want.

– Niklaus Wirth

A Plea for Lean Software – 1995



Requirement engineering 101

From ideas to software



Understand the world you're entering

Stakeholders, *i.e* interested parties

- can have different **practices**
- have divergent **objectives**
- have **hidden responsibilities** and power

Every business is different

- it has its own **practice**
- it uses its own **vocabulary**
- it follows its own **rules**

As a requirement engineer, you have to **decipher, analyse** and **identify**

A black and white photograph of a diverse group of approximately 20 young adults of various ethnicities, all smiling and looking towards the camera. They are dressed in casual attire, including t-shirts, hoodies, and button-down shirts. The background is slightly blurred, creating a sense of depth.

Analyse the people

Users are just users

" One most unfortunate product is the type of engineer who does not realize that in order to apply the fruits of science for the benefit of mankind, he must not only grasp the principles of science, but must also know the needs and aspirations, the possibilities and the frailties, of those whom he would serve. "

[Vannevar Bush]

Thinking in terms of generic users won't work

- complicated to extract **clear requirements**
 - hard to identify **system's added value**
 - very few systems target people with **homogeneous characteristics**
- thinking about users helps in **designing appropriate user interfaces**

Getting into users' mind

Alan Cooper's **Personas** – Cooper, *The Inmates are Running the Asylum*, 1999

- user **archetypes** synthesised from common characteristics
- well identified **behaviour patterns** and goals
- **fictional characters** with fictitious details

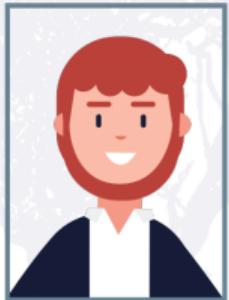
Represents behaviour, no real job description

- characteristics, i.e **skills**, **environment** and **goals** regarding the system
- possibility to have **more than one persona for one job**

But beware

- keep the **number of personas as limited as possible**
- bringing them to life does **not mean role playing**

Examples of *personas*



Cyle, 48
CIO
company founder
no IT skills
globetrotter



Lina, 35
accountant
13 years of exp
IT enthusiast
philatelist



Theo, 23
programmer
just joined
geeky-attitude
dev. games



Elena, 38
HO secretary
15 years of exp
basic IT skills
NPO admin



Celine, 52
CEO
prev. HoAccount.
reticent to change
play golf



Focusing on user profiles (*aka actors*)

Alternatively, focus on "*classes*" of users

- **physical** characteristics (*i.e* possible disabilities)
- **cognitive** characteristics (*i.e* disabilities and motivation)
- relevant social, ethnical or religious **specificities**
- **educational** background and competencies regarding the job
- task **experience**

Profiling creates *abstract* users

- highlight **discrepancies** between users
- helps in **focusing** user interfaces

As personas, profiles are defined from user interviews, **not your imagination**

Examples of user profiles



The background of the slide features abstract architectural elements, specifically large glass cubes with a grid pattern, set against a light blue sky. One cube is prominent on the left, and another is visible at the top right.

Analyse the context

Do we talk about pipes, pipes or pipes?



The context does matter

Each organisation is particular

- uses its own **terminology** and **concepts**
- always has **dissimilarities in similarities**

From interviews and discussion, requirement engineers must

- understand the **core** business **concepts** and their **relations**
 - specify their concrete **semantics**
- dictionary of concepts and attributes, e.g., entity-relationship, class diagrams

When entering a new organisation, interviews are also meant to

- identify the **key people** and **underground power relationships**
- adjust your **communication** and **behaviour** to fit the organisation's style

The fine art of conducting interviews

Three ways of interviewing stakeholders

conducted using a predefined set of questions

open brainstorming-like exploration of their needs

mixed a bit of both approaches

First interviews are critical

- don't get abused by **domain jargon** (it's ok to say "*I don't understand*")
- be **open-minded**, but remember this lecture's quote
- **rapid** sketching of **prototypes** are often useful

Remember **interviewees may not master the day-to-day** reality

Translate interviews into requirements

Natural language

- ordered **free-form** sentences with no particular characteristics

Structured natural language

- following a **template**, e.g., *as a <actor>, I want to <action>* [M. Cohn]

Graphical notations

- **diagrams**, e.g., UML Use Cases [[OMG](#)] or User Requirement Notation [[D. Amyot](#)]

Formal specifications

- **mathematical descriptions** based on, e.g., finite state machines [[E. F. Moore](#)]

Validate requirements

Ensure faithful and accurate translations (on top of **INVEST**)

valid does the requirement reflect the user's need?

consistent are there any conflicting requirements?

complete requirement definition should be self-contained

realistic does the current technologies and knowledge allow such feature?

verifiable aka INVEST's testable

Common validation practices

review sit with the stakeholders and systematically check

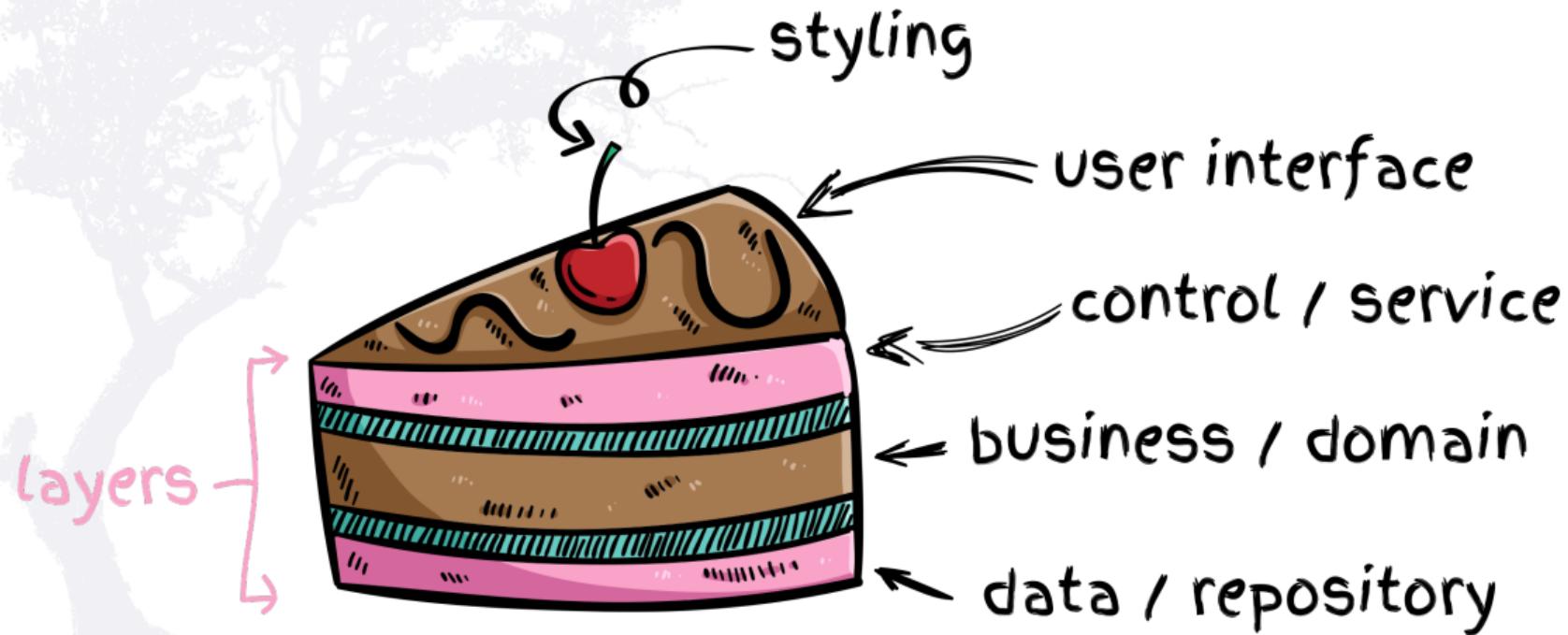
prototype either build a *Proof of Concept* or write some sketches

test cases list sample / fraudulent usages (should be obvious)

A round cake with a pink frosting base and a dark chocolate drip down the sides. The top is decorated with large, fluffy pink swirls and smaller, pointed white meringues with red stripes. Small white and pink pearls are scattered across the surface.

Break down the work

Slice it the right way





QUIZ TIME!



Weekly reading

Keep up the good pace

The right stand up for you:

<https://www.martinfowler.com/articles/itsNotJustStandingUp.html>

Remember to keep notes, even if you won't write a reflection on that one

- what is it about?
- did it seem **relevant** to you?
- what are your **takeaways**?



**TO BE
CONTINUED... ➤**

Let's put it all together

Can I pass the sources with a USB stick to my team mates?

Submission is in 5 minutes... Oh no, a merge hell!

You are more than your code

... come next lecture

A close-up, low-angle shot of the Atomium's spheres at night. The spheres are large, reflective hemispheres with a grid pattern and numerous small lights. They are connected by thick, light-colored beams forming a cube-like structure. The background is dark.

That's all folks!