

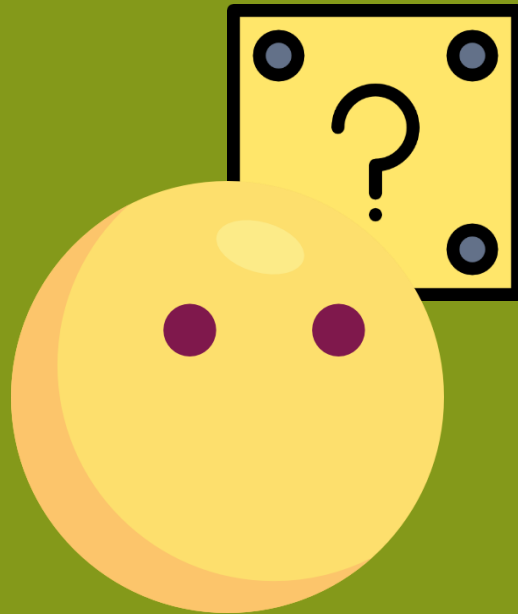
SOFTWARE ENGINEERING

Chapter 1: Key Concepts

MOTIVATION...

Programming is fun, but developing quality software is hard. In between the nice ideas, the requirements or the "vision," and a working software product, there is much more than programming [LARMAN]

What is software engineering?



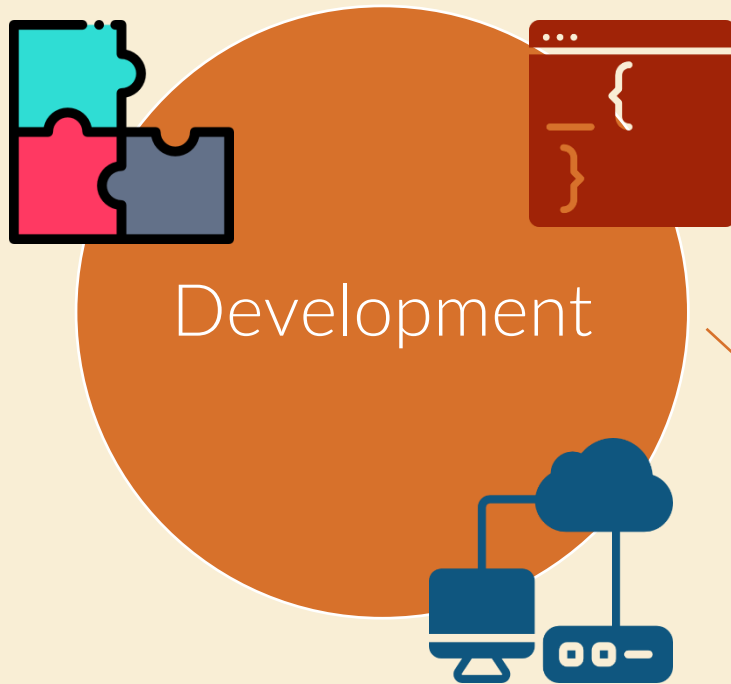
FIRST DEFINITION...

Software engineering is the usage of *proven practices and techniques* derived from systems engineering adapting them to address the challenges of software development. [SCHMIDTH]

SOMMERVILLE SAYS...

- It's a discipline that joins all the **software processes** involved to develop a working software product.
- It's **engineering** as you may have to do everything possible to make things work even with **limitations** (money, team, time...)

It is...

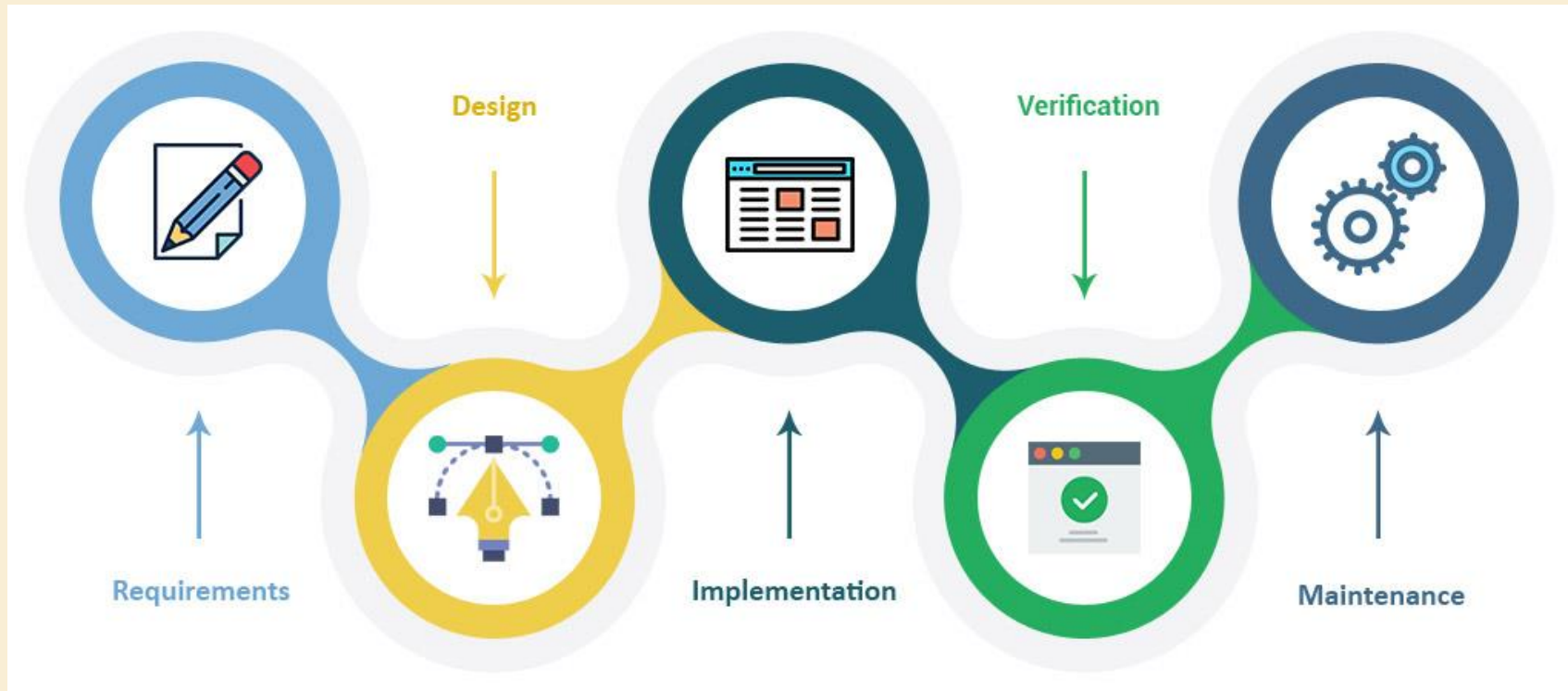


Software
Design

- Programming

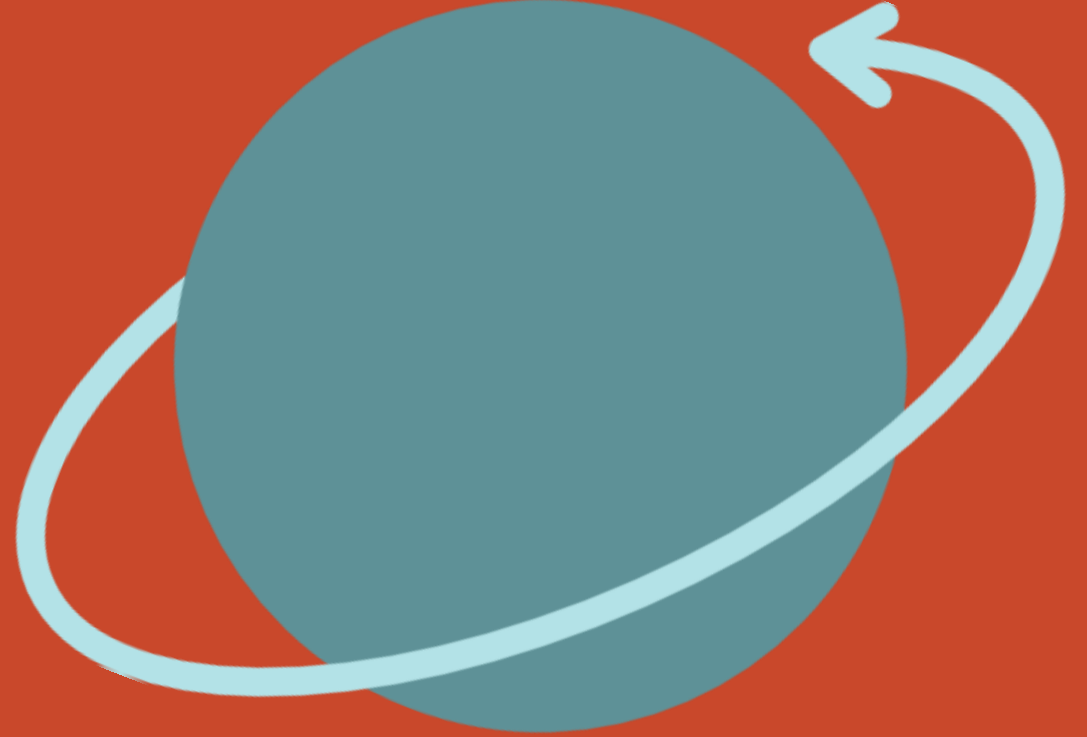


AND SOFTWARE DEVELOPMENT IS...

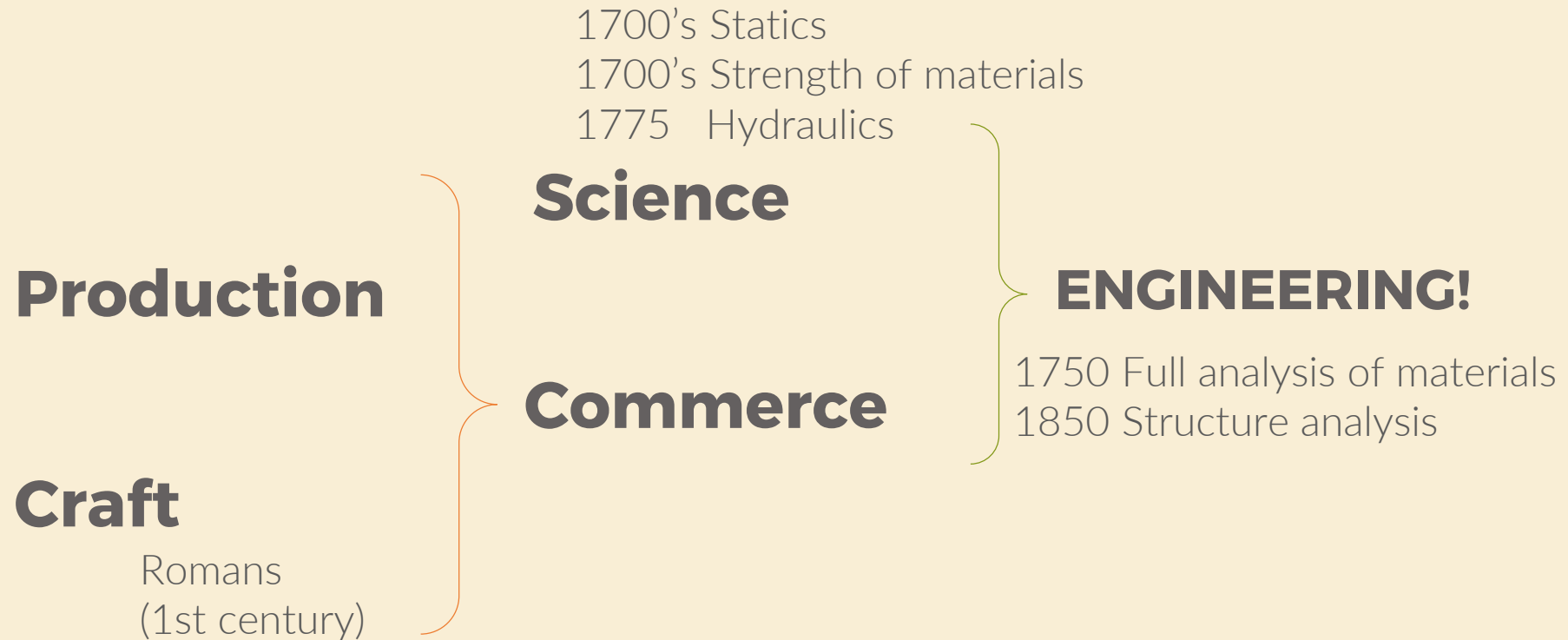


NOW...

Let's see some history, how did we get where we are now?



ENGINEERING?



ENGINEERING?

Production

Software development
methods (SDM)

Craft

Science

Emerging research!

Commerce

1990: Adoption of SDM's

ENGINEERING!

Here we are...

Why not... a timeline?



TIMELINE I

1940: First
digital
computers



1950:
Programming
languages



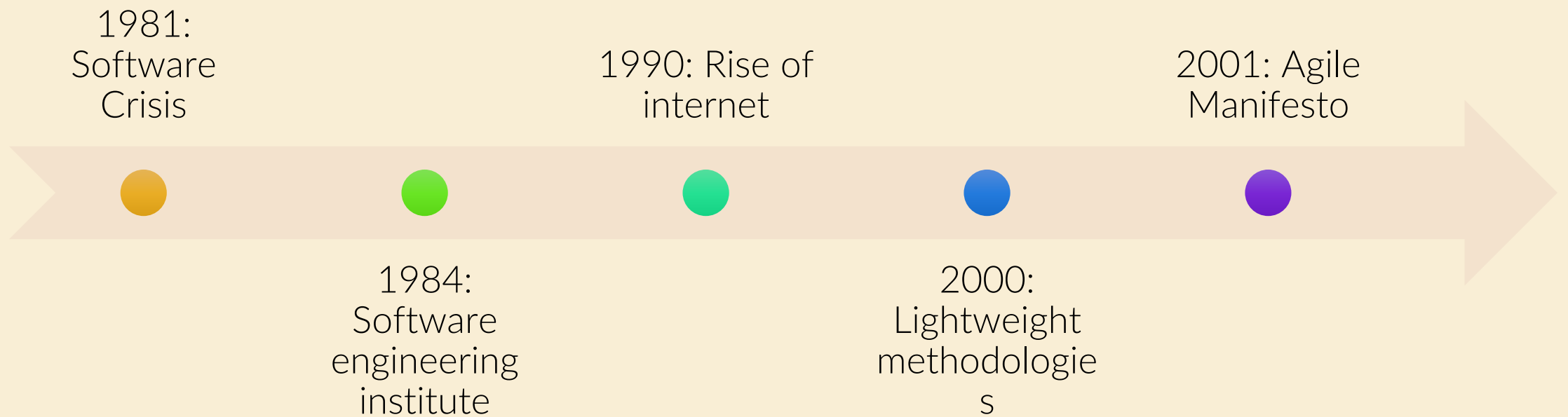
1961:
Modularity



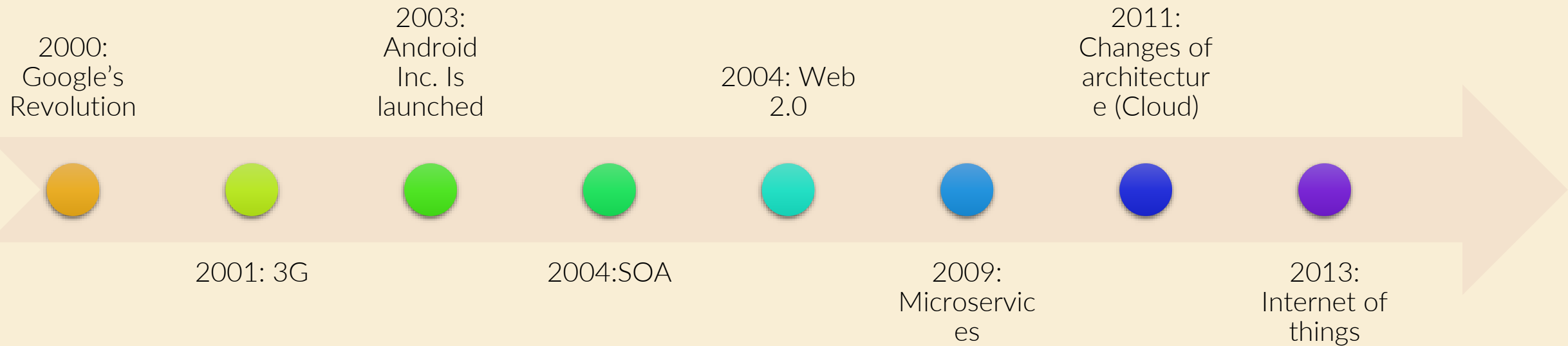
1981: Software
engineering



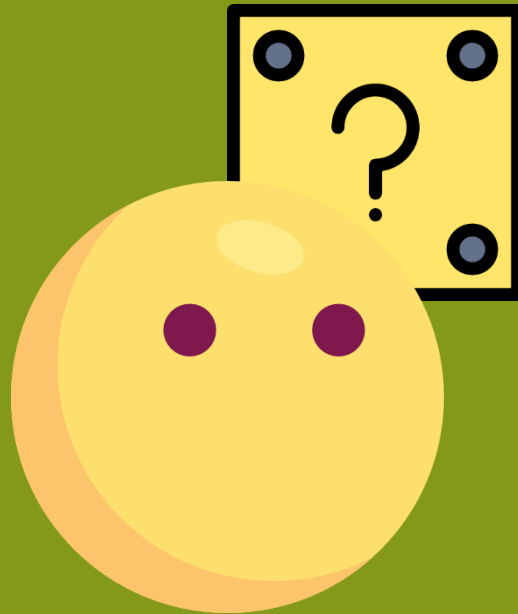
TIMELINE II



TIMELINE II

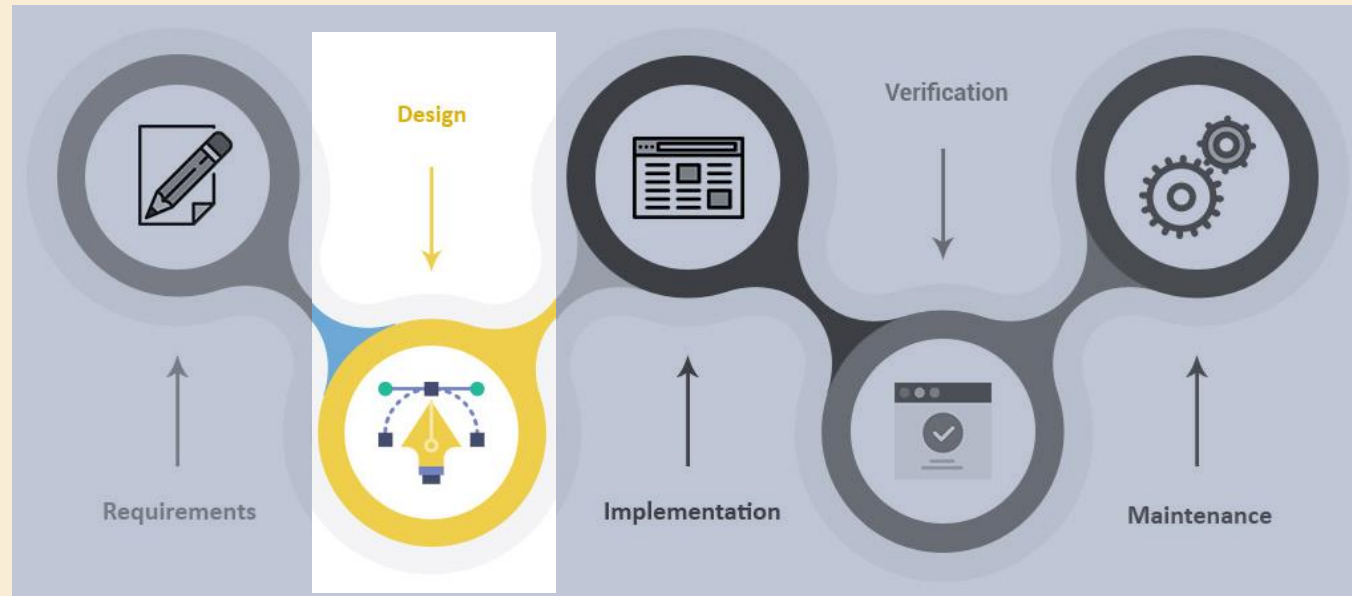


What is a software process

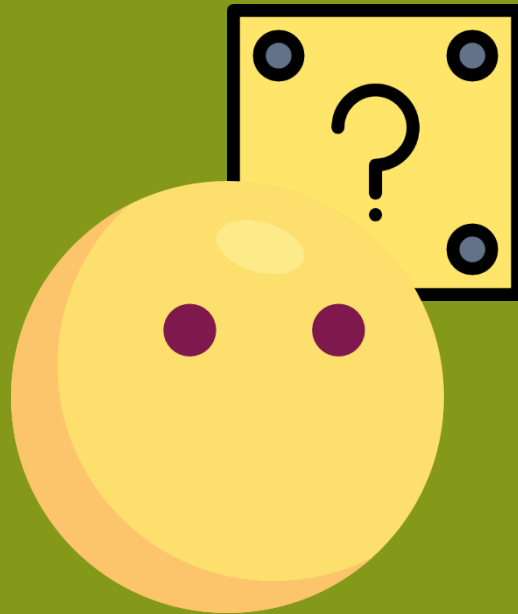


SOFTWARE PROCESS

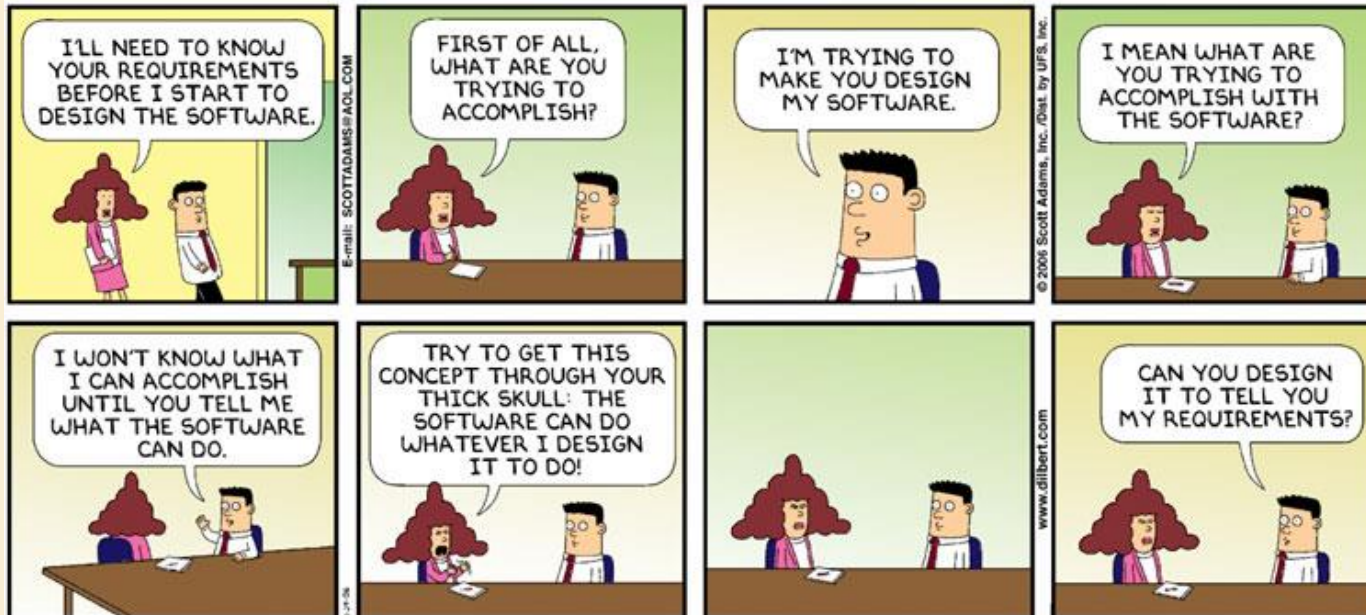
- It's the set of activities needed for getting a result...



**Could you define
everyone of them?**



REQUIREMENTS



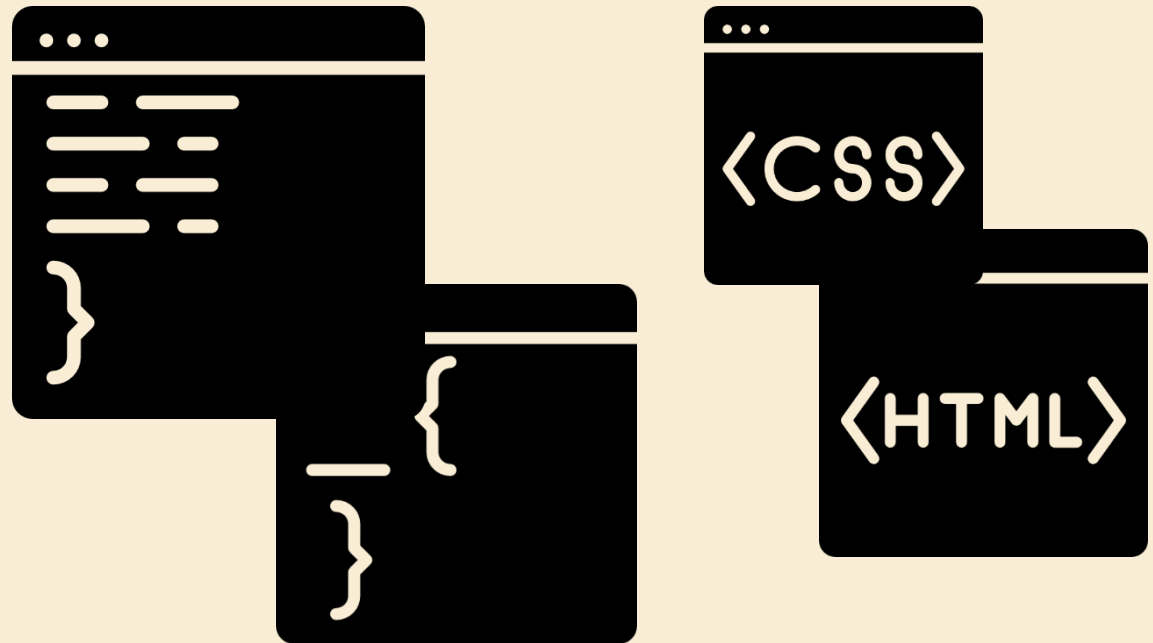
DESIGN

- Design shows every single piece of software that needs to be done, that means, databases, interactions, interfaces
- What data composes your system and how they interact
- This is the guide that programmers will implement



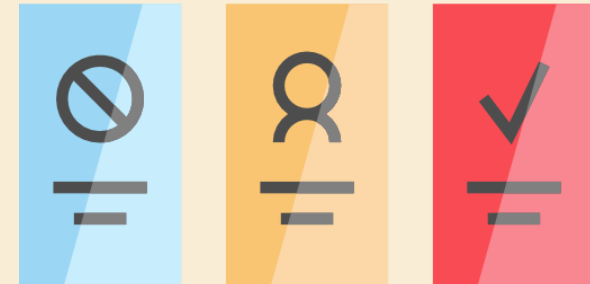
IMPLEMENTATION

- Well, let's write some code!
- Designs become code
- You may need IDE's
- Source code control
- Profilers
- Static analysis tools
- Refactoring



TESTING

- Developers do not put bugs intentionally
- Developers test their code
- Non-developers test the software [QA]
- After everything is integrated, the whole package is tested



Are there bugs?, will any optimization be needed?, what if my connection constantly falls?, How can I make this thing to fail?

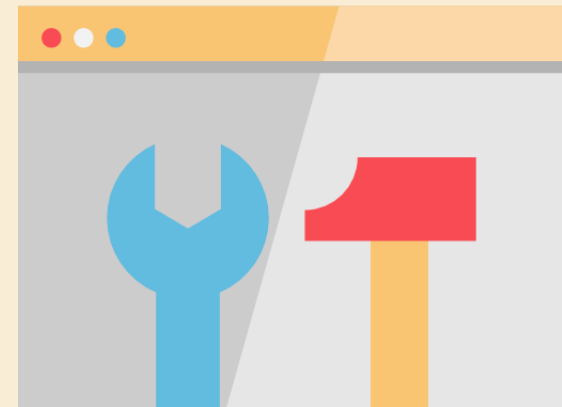
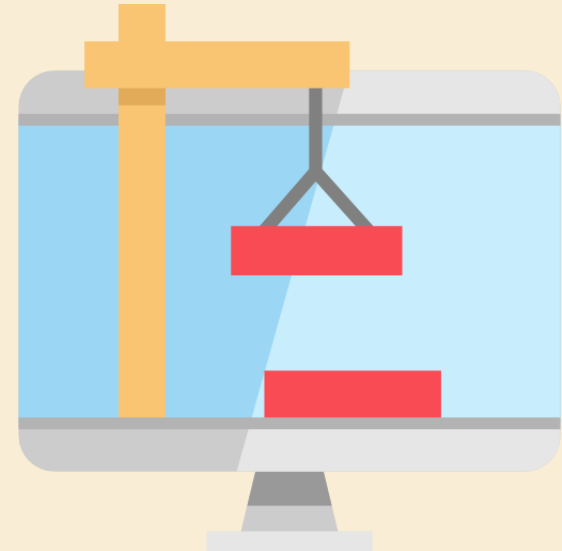
DEPLOYMENT

- It's the process of putting the final application in the user's hands.
- You may need an installation manual!
- Tasks:
 - Environments
 - Hardware resources
 - Server configuration
 - Cloud configuration
 - Software dependencies

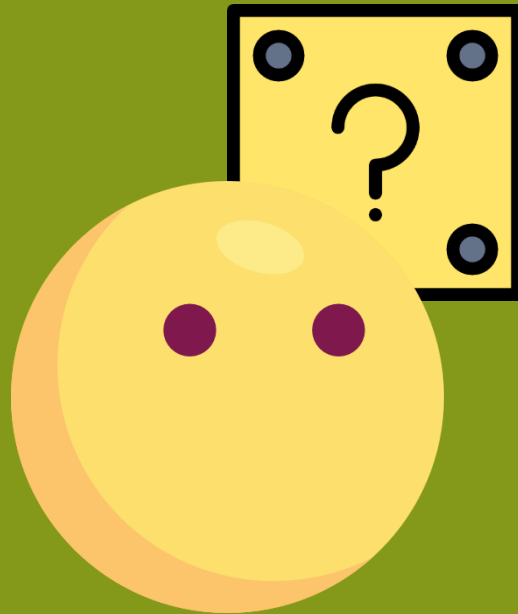


MAINTENANCE

- Every single effort that is put to every single piece of software after it has been deployed:
 - Patches
 - Security Updates
 - Compatibility with newer hardware
 - Optimization



Is there another approach?



Software Processes... Generic



Specification



Development

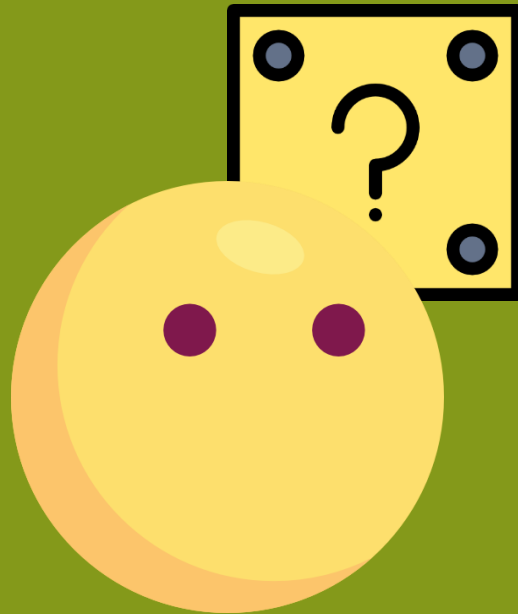


Validation



Evolution

What about process models?



A PROCESS MODEL..

- Well, it's how you manage to organize and prioritize activities...
- It's what we normally call software development methodologies

PROCESS MODELS

Predictive



- Waterfall
- Waterfall with feedback
- Sashimi
- Incremental Waterfall

Iterative:

Spiral
UP
RUP



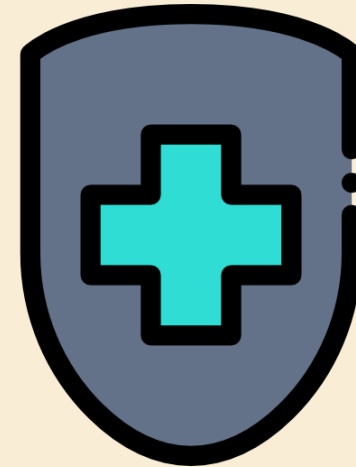
PROCESS MODELS



Agile

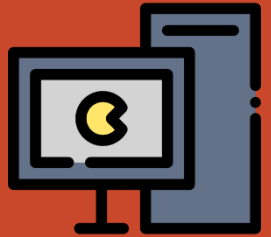
- XP
- Lean
- Crystal
- FDD

- AUP
- DAD
- Kanban



Software Engineering Challenges

Challenges



1) Heterogeneity:



2) Delivery:



3) Trust:

Different
computers

More
support

Different
Programmin
g Languages

Faster is
better!

Users can
rely on
software

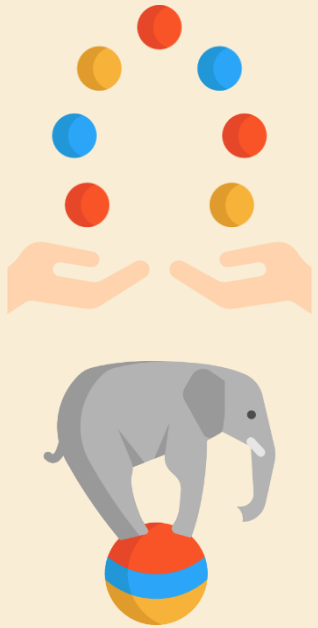
**And the reason why this is
difficult is...**

You need to be careful with

- Capturing the functional and non functional requirements
- The choice of the right development and deployment tools
- Manage hardware properly
- Doing good designs
- Having recovery strategies
- Choosing development methodologies



And you may need a lot of skills!



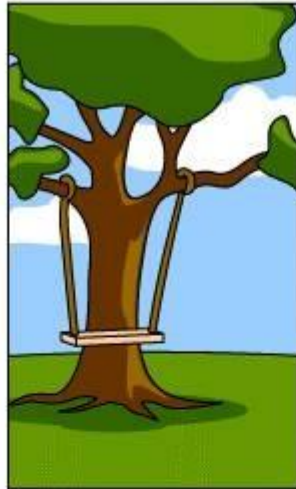
- Specification fundamentals
- Design concepts
- Performance considerations
- Resource management

- Good practices
- Software development process
- Security
- Coding
- Quality





How the customer explained it



How the Project Leader understood it



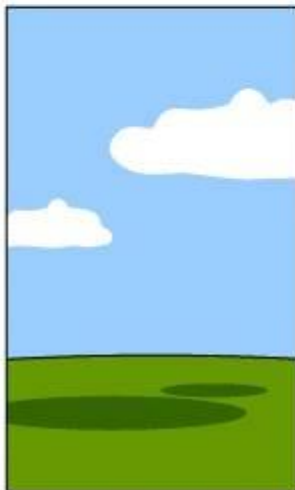
How the Analyst designed it



How the Programmer wrote it



How the Business Consultant described it



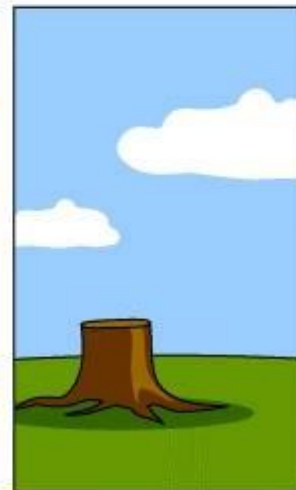
How the project was documented



What operations installed



How the customer was billed



How it was supported



What the customer really needed

**And if we take into account
every single aspect...**



How the customer explained it



How the project leader understood it



How the analyst designed it



How the programmer wrote it



What the beta testers received



How the business consultant described it



What the digg effect can do to your site



The disaster recover plan



The Open Source version



How it performed under load



How patches were applied



How the project was documented



How it was supported



What marketing advertised



When it was delivered



What operations installed



How the customer was billed

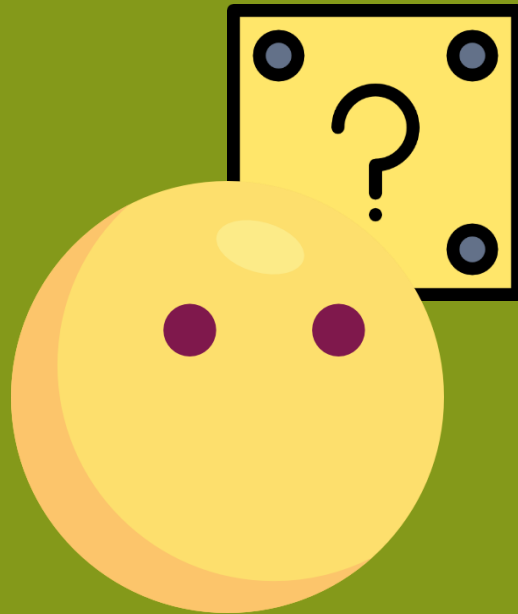


What the customer really needed

This is worse...



Can we talk a bit about software?



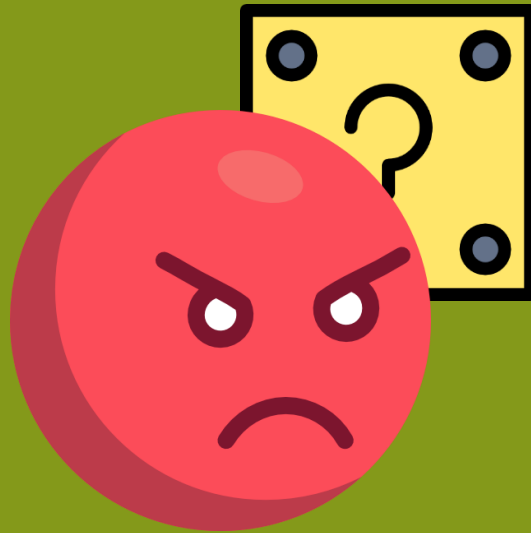


Software Definition!

Is the set of computing programs, processes, rules, documentation and associated data that becomes part of the operations of the system

IEEE Standard 729

Another one please!!





Software Definition 2!

Computer programs and associated documentation. Software products may be developed for a particular customer or may be developed for a general market.

SOMMERVILLE

GENERIC SOFTWARE VS CUSTOM

Generic



Custom



4 Key Elements for Software

1) Efficiency

- Consuming just the necessary!

2) Trustability

- Security, Protection, Stability

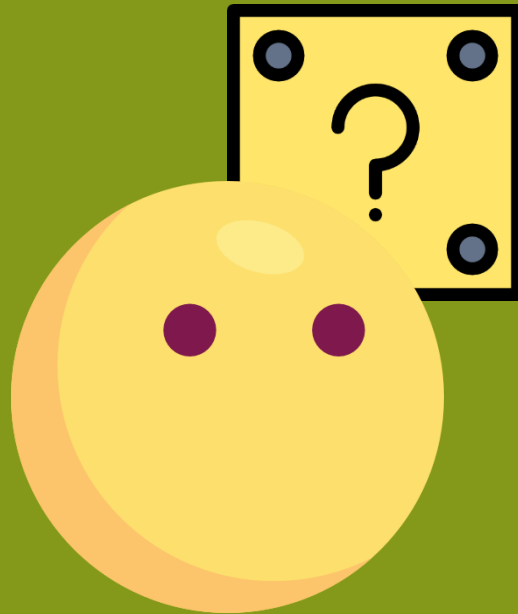
3) Usability:

- Intuitive?, easy to use

4) Evolution

- Software will always change, even after deployed

What about Colombia?



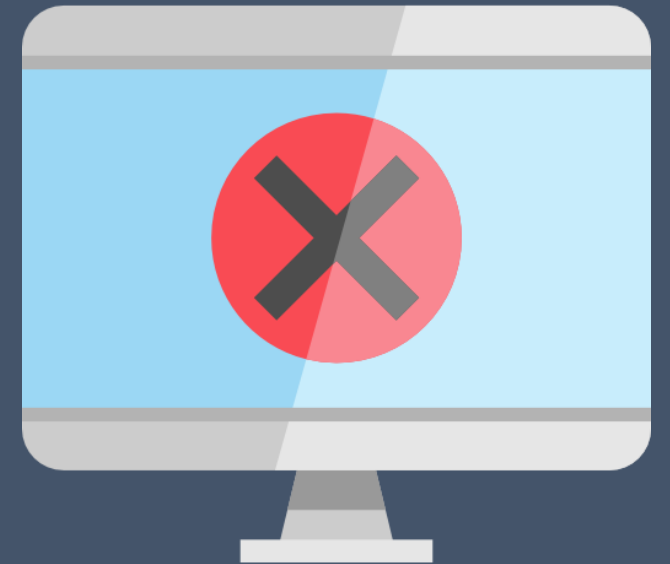
Well...

https://disqus.com/home/discussion/bunnyincblog/colombia_doesnt_have_any_software_engineers_really_not_even_one/

- It's a fact that we have skilled and good engineers, but... on the other hand our companies prefer technologists and technicians rather than engineers.



WE NEED BETTER COMPANIES!



The bad engineering culture is closing the gap (in a bad way) between technicians and engineers

Let's see an example!



Smiley and the data system! PT 1



Smiley has won an scholarship that allows him to stody abroad

He just needs to put all of his information into a data system.

Smiley and the data system! PT II



1) Time loads were extremely high!

2) After Loading and adding all the data into the forms, the system won't ever validate the info.

Usability and efficiency --

Smiley and the data system! PT III

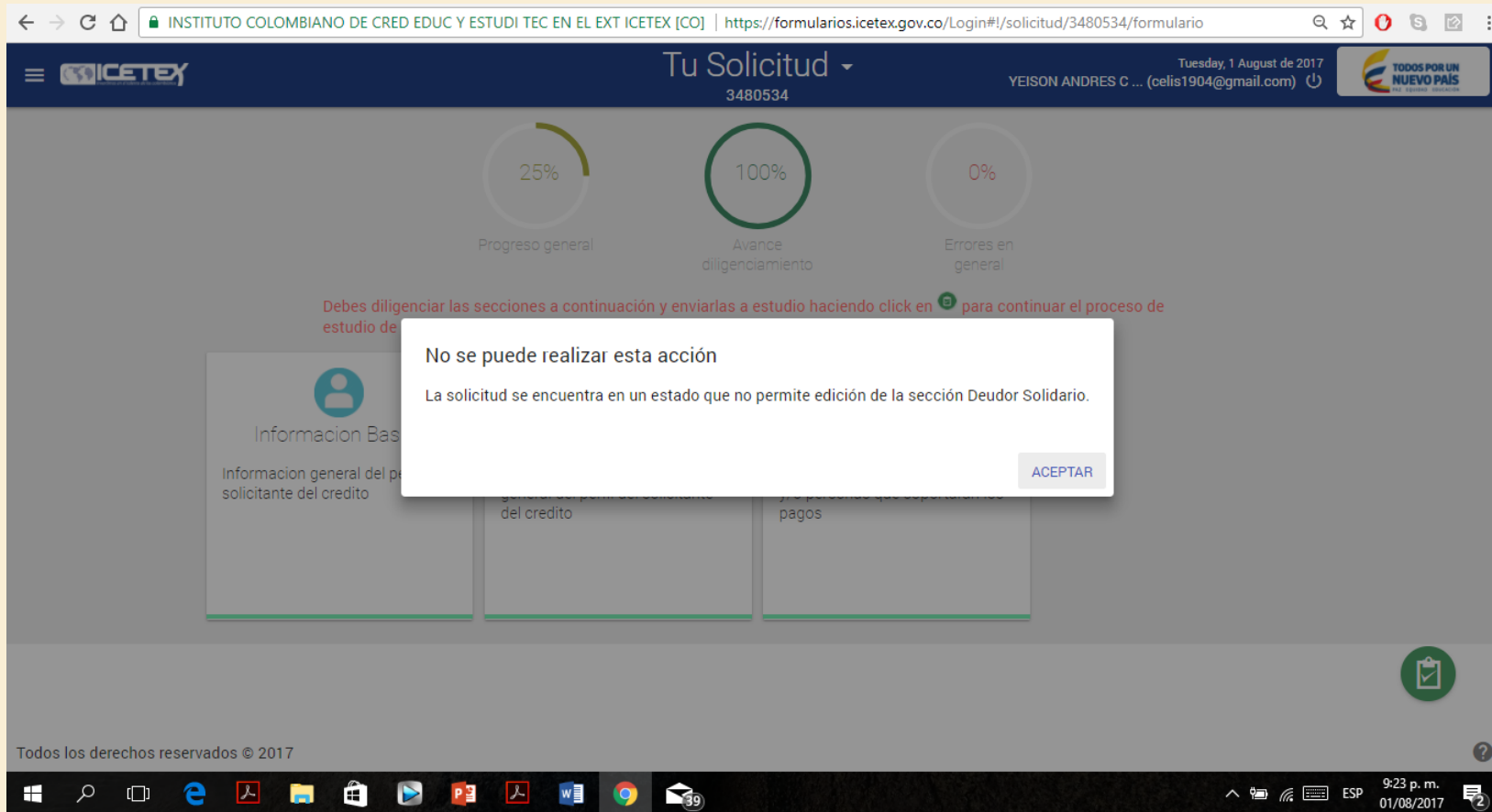


After several sessions of phone calling they confirmed that the issue is that *some parts of the form are not visible for the users*



They resolved the issue, and smiley tried the system again

Smiley and the data system! PT IV



1) One form became totally inaccessible

2) The system started to crash with errors (like 404), and the data validation became impossible

Nice fix!

Trustability and efficiency --

Smiley and the data system! PT V



After more sessions of emailing and phone calls their decision was to

RE DO ALL THE SYSTEM



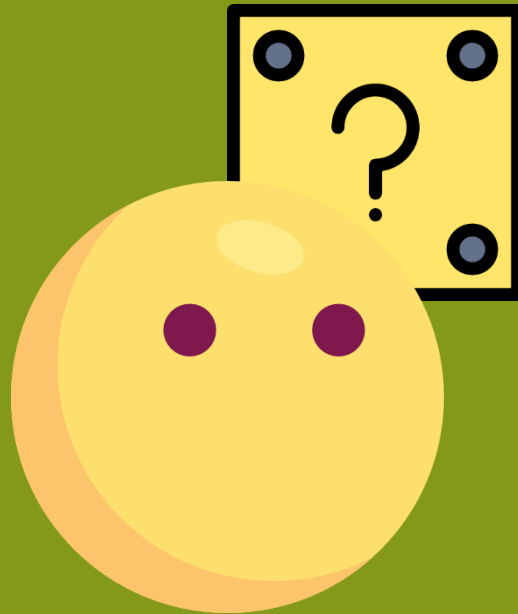
So.. Think about their Budget, timing, management, and software processes



Things may change if...

- Software engineering became a MUST discipline
- We had respect for the quality of every software produced
- We had genuine interest and software contributions

What about Ethics in Software Development?



4 Basic principles



Confidentiality

- Respect to clients and employers



Competence

- Don't lie about you



Intellectual property rights

- Libraries... copyright...



Computer misuse

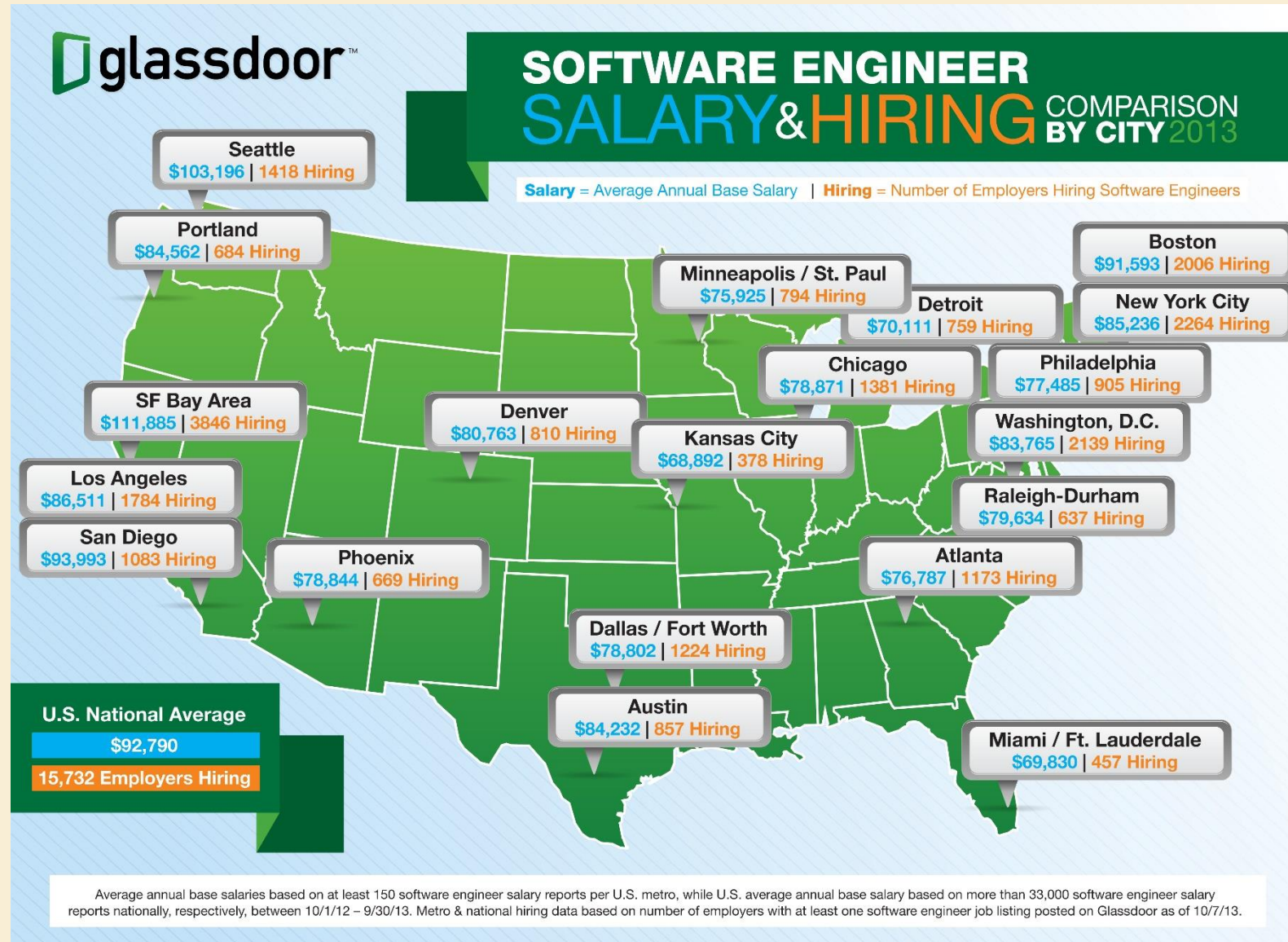
- Playing games?
- Pouring malware into someone's machine





Let's talk about money!

What about other countries?



What about programming languages?



One Shot Review



Review!

- What is software engineering?
- What is a software process?
- What is a process model?
- Why is software engineering difficult?
- Which are the challenges of software engineering?

References

- [SOMMERVILLE] Ian Sommerville. *Software Engineering 9th Edition*
- [SCHMIDT] Richard Schmidt. *Software Development Architecture-Driven Software Development*
- [KUMARAN] Why Software Engineering is complex?
- [STEPHENS] Beginning Software Engineering. 2015
- [CROOKSHANKS] Software Development Techniques. 2015



Class has died... for today!