

# IF1006 – DevOps Software delivery way: SDLC Challenges

Fish

@fisholito

jfsc@cin.ufpe.br



This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.



## What did we learn until here?

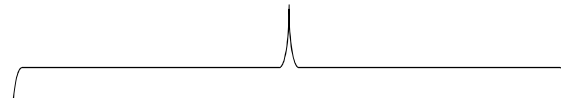
---

- Digital transformation requires new ways to do software (product)
- The customer needs are changing in a fast way and we need to provide rapid responses;
- Cloud is a key element for Digital transformation;
- Cloud helps software delivery;
- We need to eliminate the DevOps Bubble.
- Testing is a fundamental part of Software delivery

# What is software development?

---

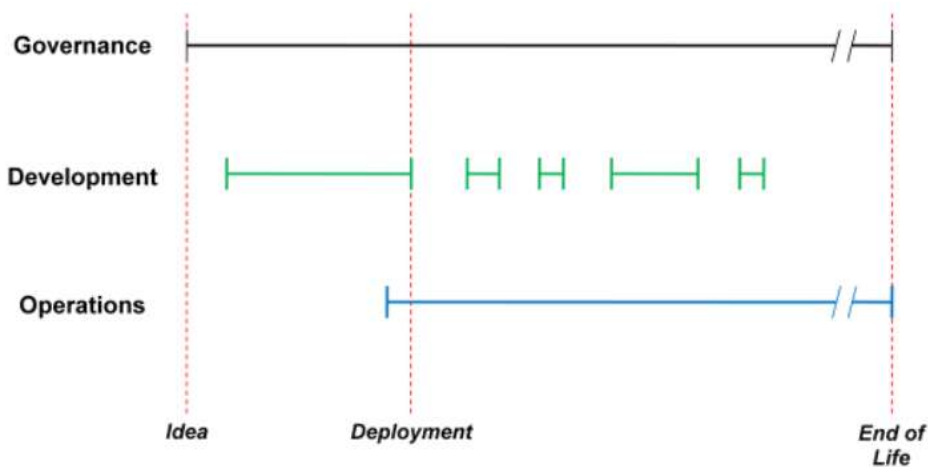
IDEA -> \*PROCESS -> CODE

The way we do software

\*In a general context

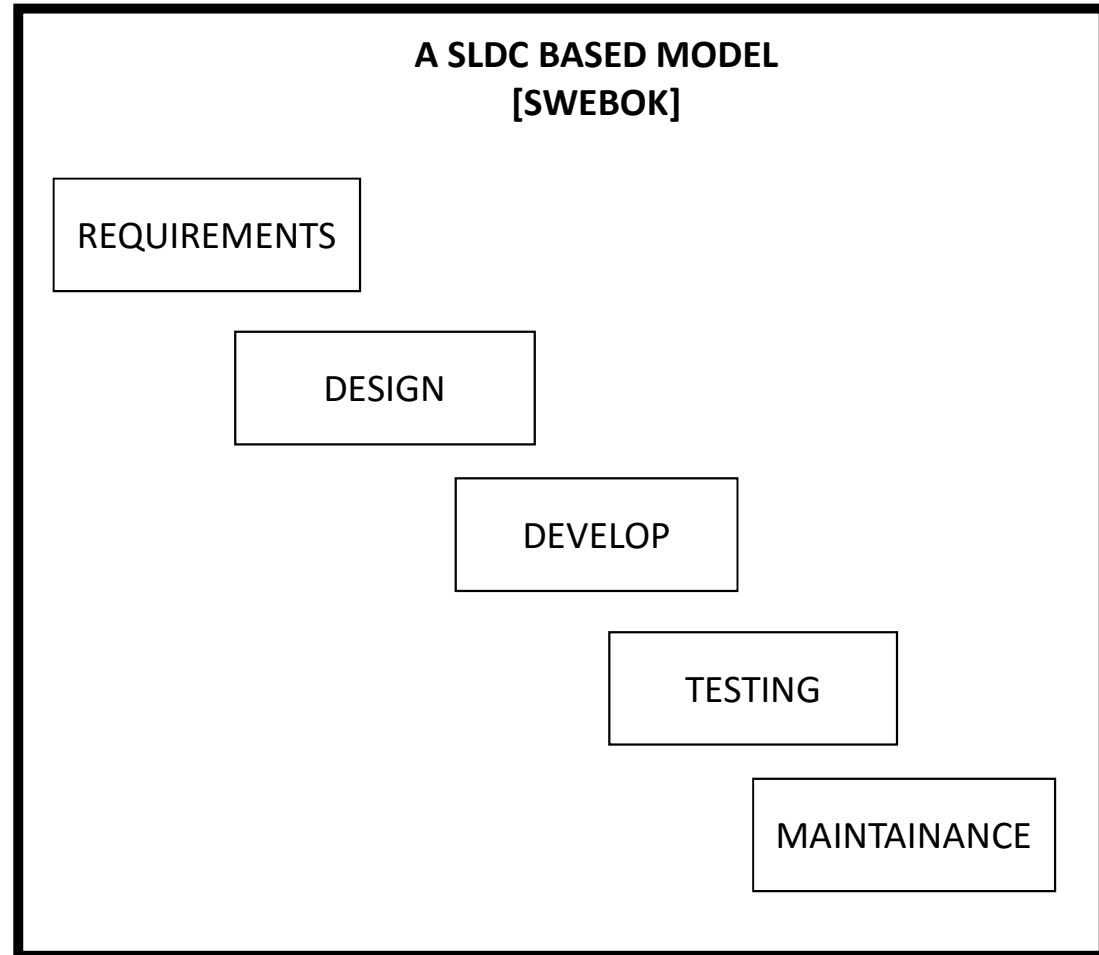
# Models to see our way to develop/deliver/manage software

**ALM MODEL  
[DAVIDCHAPPELL]**



**WE'LL ADOPT THAT FOR THIS COURSE**

**A SLDC BASED MODEL  
[SWEBOK]**



# Last years in software delivery (2012)

## Knight Capital Says Trading Glitch Cost It \$440 Million

By NATHANIEL POPPER AUGUST 2, 2012 9:07 AM 356

Runaway Trades Spread Turmoil Across Wall St.



Errant trades from the Knight Capital Group began hitting the New York Stock Exchange almost as soon as the opening bell rang on Wednesday. Brendan McDermid/Reuters

1 of 4



Wrong decisions through whole SDLC  
turn deployments a nightmare.

# Last years in software delivery (2013)

Failures in software development projects inflicts negative impacts in our daily lives.

cheat sheet · Do IT certifications matter? · Microsoft pulls plug on Win 10's debut version · Newsletters · Resources/White Papers


**ERWORLD**  
FROM IDG

Home > Vertical IT > Healthcare IT

NEWS ANALYSIS

## Healthcare.gov website 'didn't have a chance in hell'

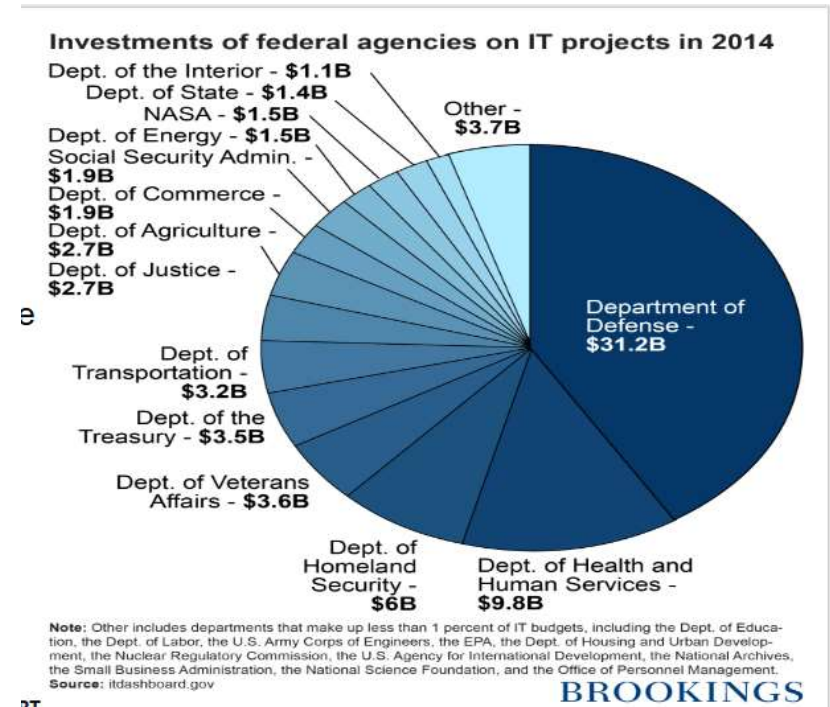
The failure rate for software development projects is high generally, particularly large ones like Healthcare.gov, says Standish Group data



Healthcare.gov website 'didn't have a chance in hell'  
<https://goo.gl/GngXFy>

# Last years in software delivery (2014)

75.6 billion dollars on IT projects in 2014. Only 6.4% of software is delivered to federal agencies.



# Last years in software delivery (2015)

The complexity in/out software  
Has broken projects.

## The Telegraph

Home Video News World Sport Business Money Comment Culture Travel Life Wo  
Politics Investigations Obits Education Science Earth Weather Health Royal Celebrity

2.800 racks e 13,3 MW de  
energia disponível em 16 mil m<sup>2</sup>

Saiba mais ▶

HOME » NEWS » UK NEWS » IMMIGRATION

### Home Office's 'flagship' £350m immigration computer system ditched

National Audit Office says IT contract failed to deliver, as it discloses backlogs are still a problem in the immigration service



Home Office's 'flagship' £350m immigration computer system ditched (2014)  
<https://goo.gl/toCt2a>

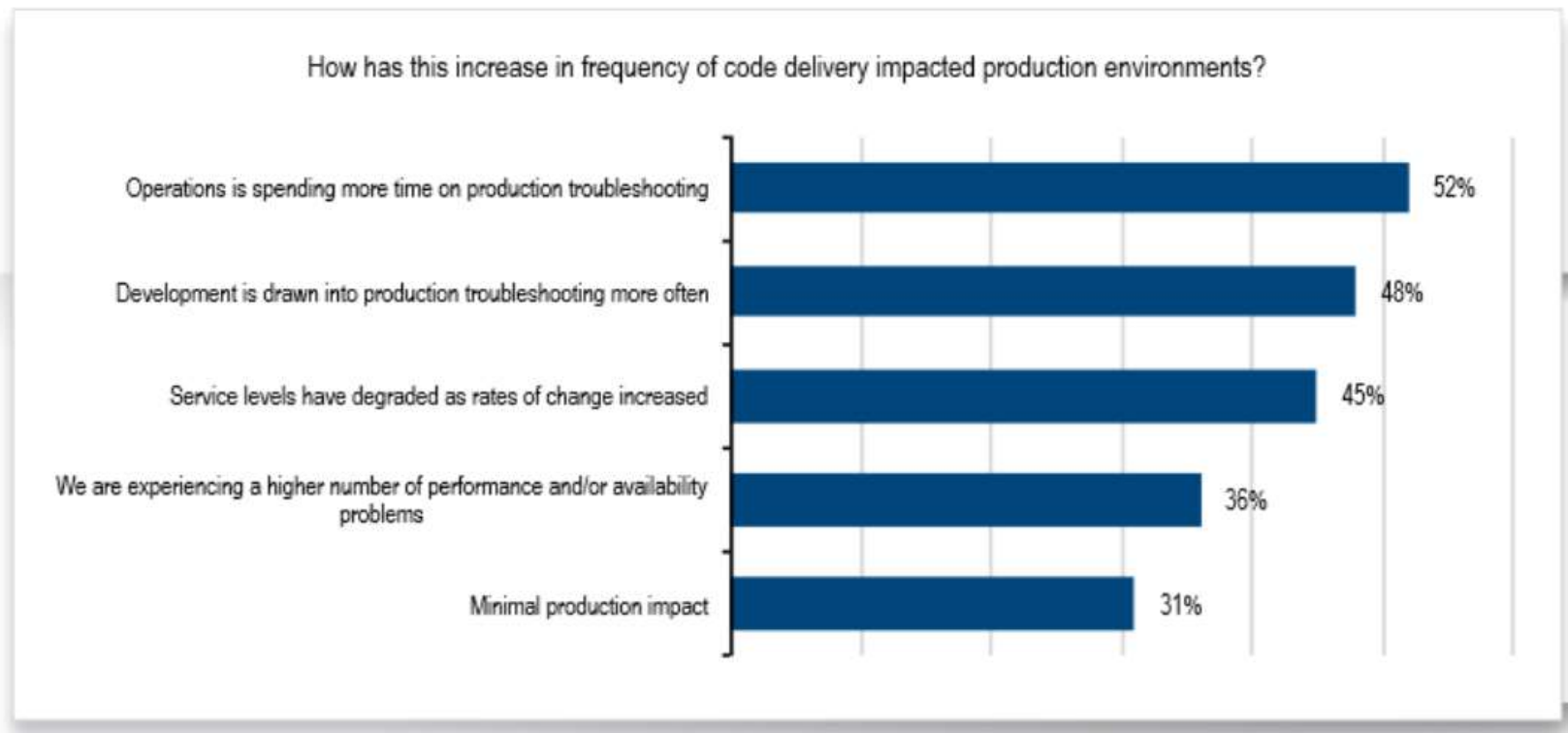


# Last years in software delivery (2016)

---

from their core function of writing code. 69 percent of study participants share the view that developers, at their North American organizations, spend 20 to 60 percent of their time on non-functional requirements such as authorization, scaling, high availability, and provisioning. By automating these relatively trivial tasks, each developer could reclaim 1 – 3 days per week with which they could write more code and add more business value.

# Last years in software delivery (2016)



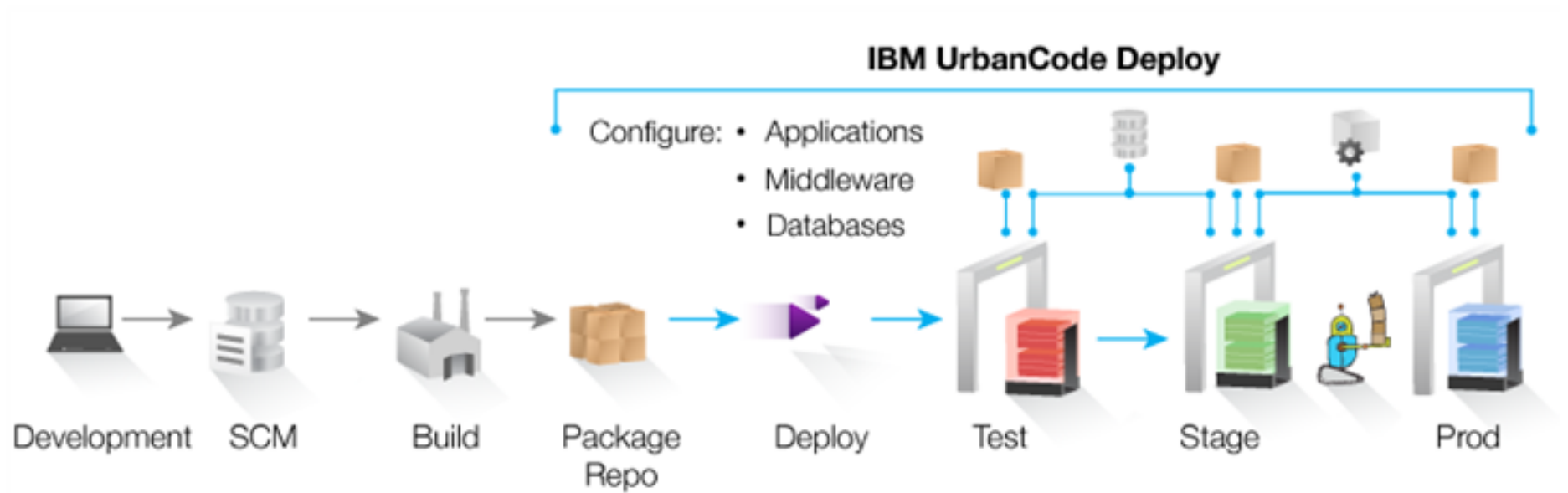
EMA. DevOps, Continuous Delivery, and the Cloud Journey: an evolutionary model.

What do you think about current software delivery reality?

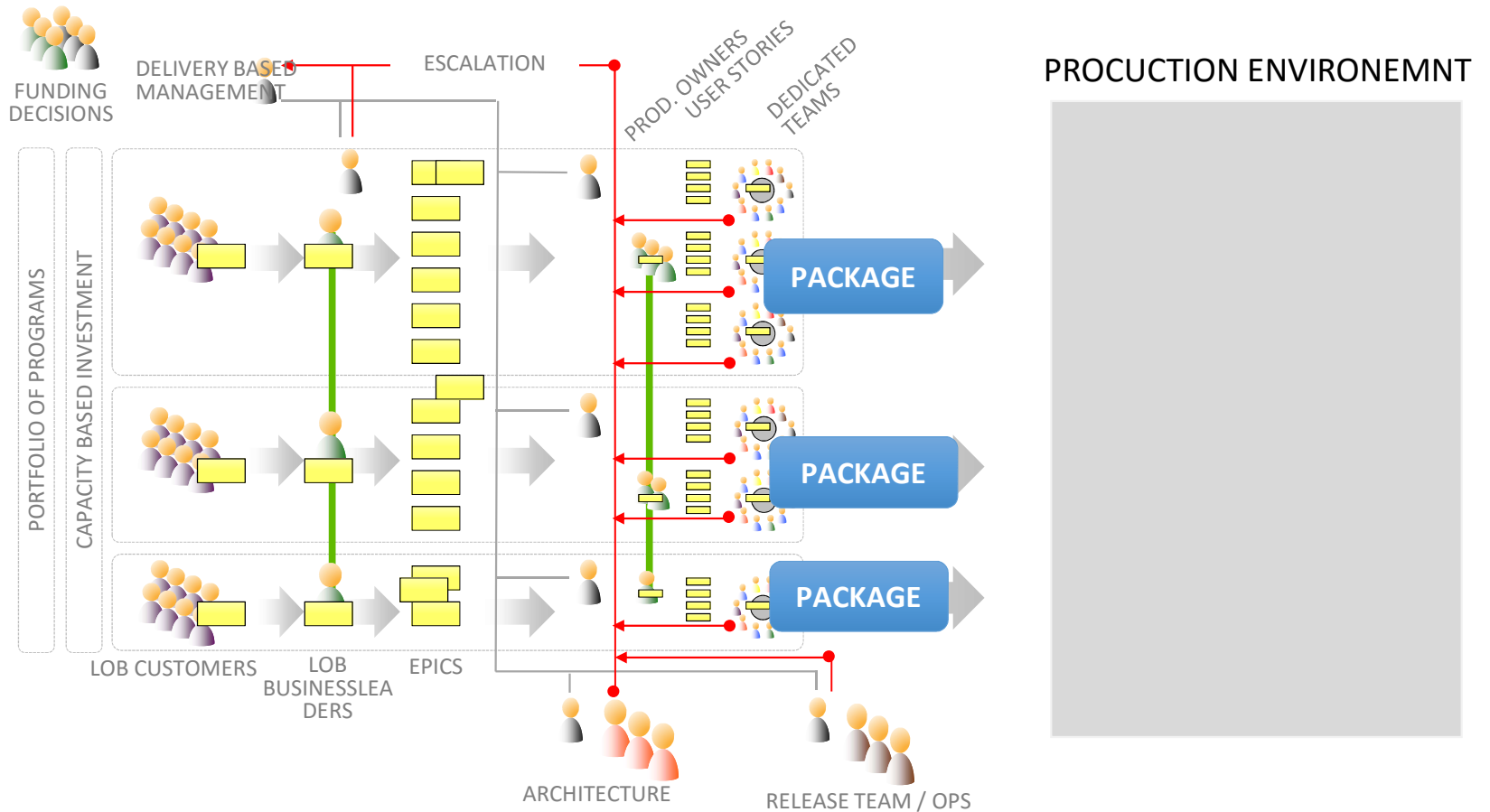
---

?

# Code path



# Lets look at this development enviroment



## Exercise

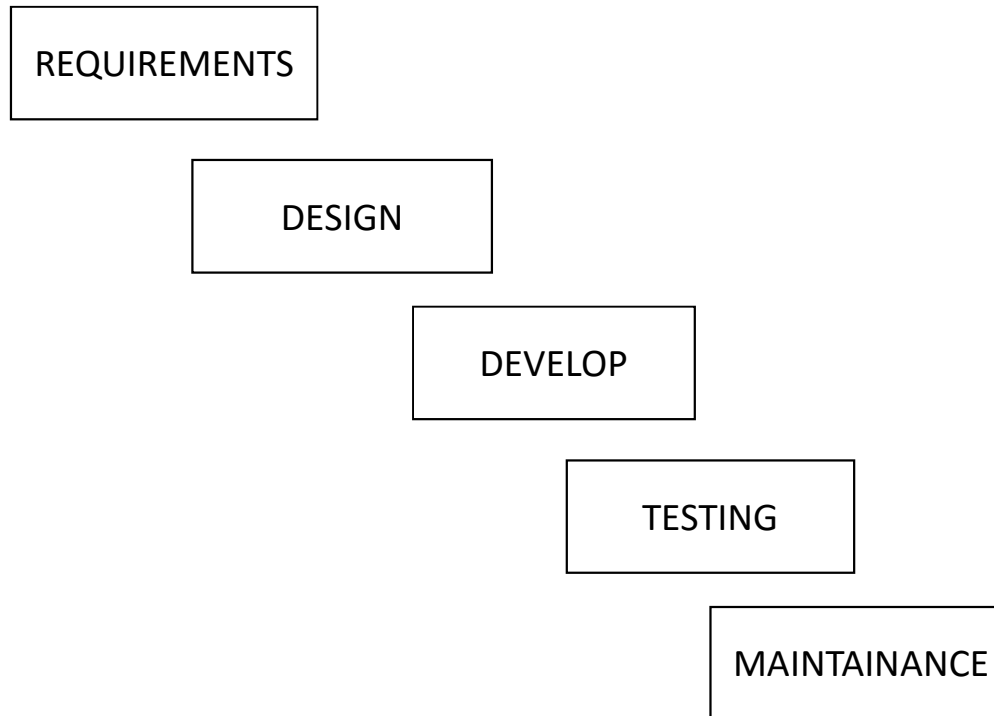
---

The team is in very high pressure due to deadline.

What do you do to mitigate the risk of wrong package deployments?

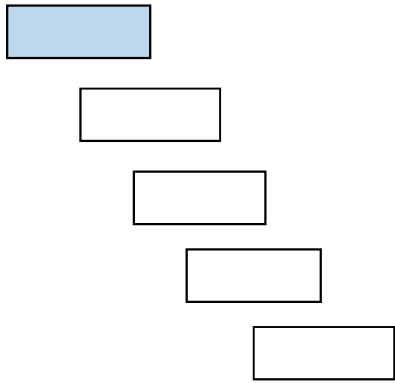
# Where is the problem ?

---



# Requirements

---

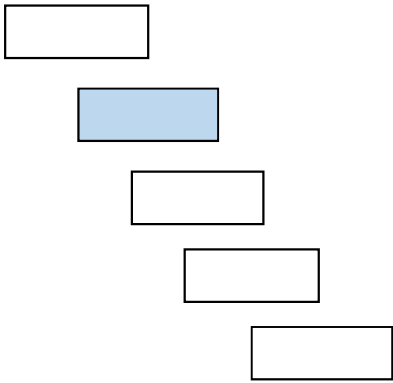


- Lack of detail
- Invisible requirements: even well document documents suffer with lack of strategic content (business);
- Isn't ease to client talk about their needs



# Design

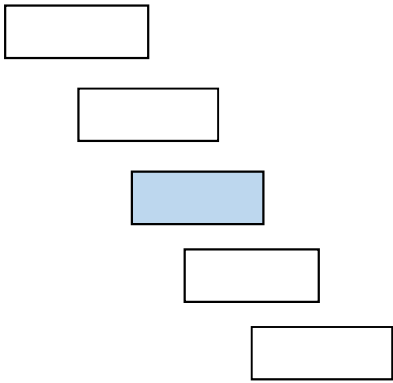
---



- Estimation;
- Flexibility of plans;
- Software architecting;

# Code

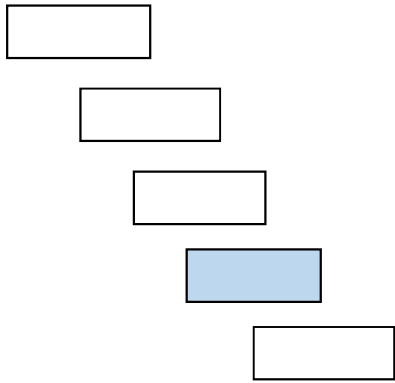
---



- Skill;
- Creative process;
- Slow;

# Testing

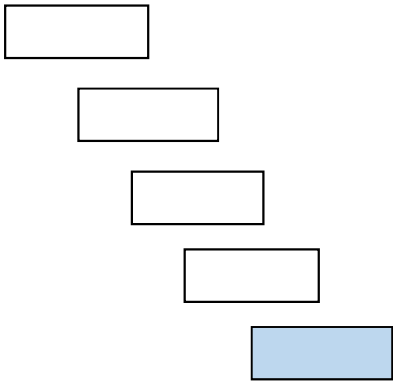
---



- Understand other ideas;
- Distribution of systems;
- Multidisciplines (today they need test infrastructure);

# Maintenance

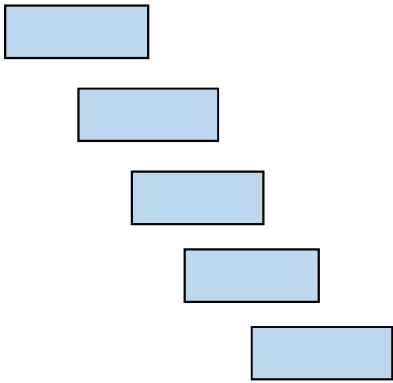
---



- Receives all decision from others stages;
- Heterogeneity of devices;
- Massive distribution;

# SDLCs are under constant pressure

---



- Time-to-Market;
- New and growing requirements;
- User is more exigent;

# Result of misalignment inside organizations

---

- We are spent more Money than needed;
- Development environments are more expensive;
- We are empowering silos and historical barriers.
- We lack opportunities;

# Result of misalignment inside organizations

---

- Dev + Ops [THIS IS OUR FOCUS]
- Buz+Dev+Ops
- Dev+Test+Ops
- Dev+Sec+Ops
- ...