

## Development Methodologies

- coding is only ~10%
  - we need to deliver product
  - the faster we do - the better.
  - deliver features faster.

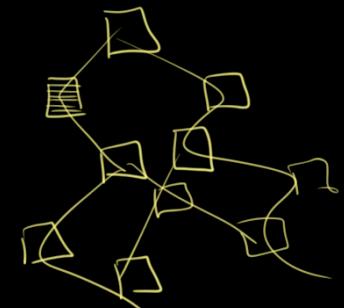
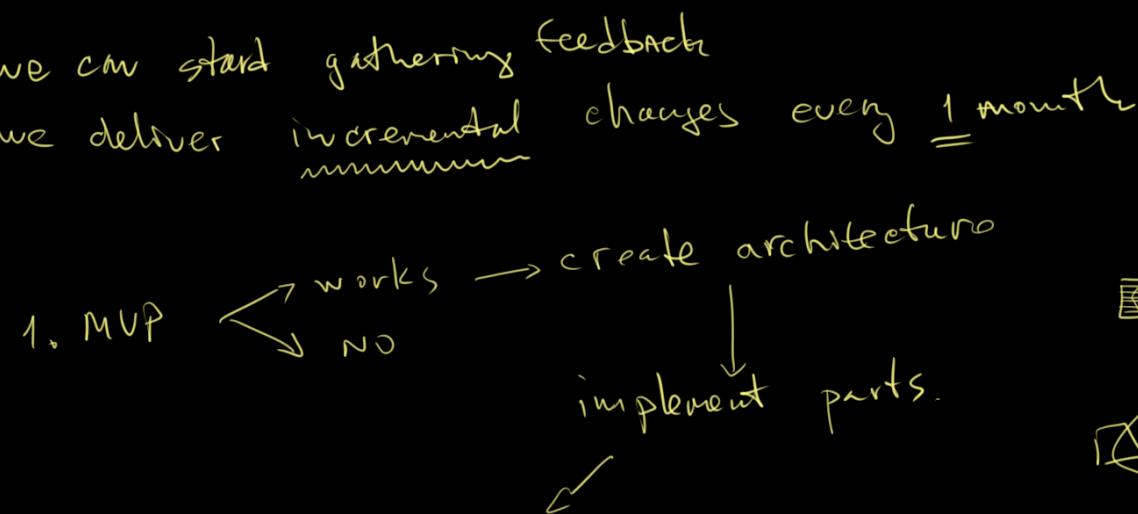
- ④ nobody is able to write good spec
- ⑤ nobody is able to estimate the resources
  - \* whether is still in demand.

1970 Mythical Man-Month

flexible

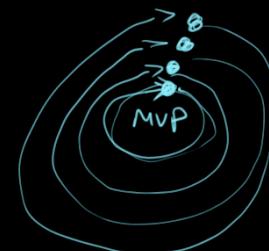
## Agile-based Methodologies

- nobody want to wait 5-10 years
- we introduce MVP - minimum Viable product.
- we introduce incremental changes



try Agile  
Not Waterfall

- 1. gathering requirements
- 2. estimating resources
- 3. implementation
- 4. deployment
- 5. retrospective



- we decide, vote, what needs to be implemented
- we estimate throughput (number of days)
- we estimate each task
- we pack only tasks to the current sprint (one iteration)
- sprint is finished when f. is delivered
- keep and meet release cycle. not skip delivery
- retrospective

$\Rightarrow$  continuous delivery

$\nearrow$  tech

$\searrow$  management



the properly you estimate - the more goals you meet  
 by understanding user problems - we can do next sprint better.  
velocity cost by voting. → if all tasks  $\approx$  similar  
 & have met before

how would you estimate task  
 if you never done it

- ↳ decompose to smaller tasks until they more/less understandable
- ↳ you understand complexity better.

scale  $\times 1.5$   $\sqrt{2}$  1 1 2 3 5 8 13 21

1d / 10d / 100d

The bigger task → the bigger uncertainty

after decomposing → task can be done in parallel

$$8 \rightarrow 1+1+3 \rightarrow 5$$

$$8 \rightarrow 3+5+5 \rightarrow 13$$



## Retrospective

- what done well
- what can be improved
- immediate steps
- blockers.
- how previous solutions were implemented

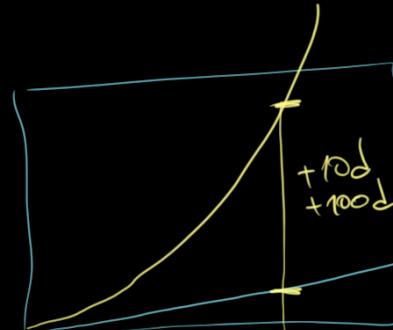
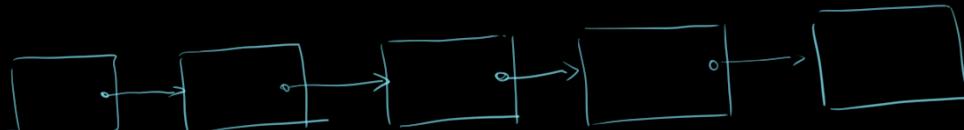
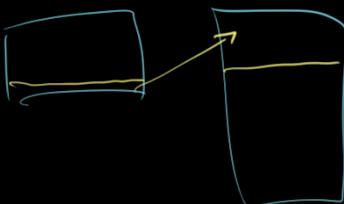
reuse good working approaches

learning process to make next better.

what is the next thing to improve

to eliminate min

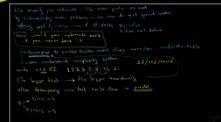
Who is responsible



## - Technical Debt

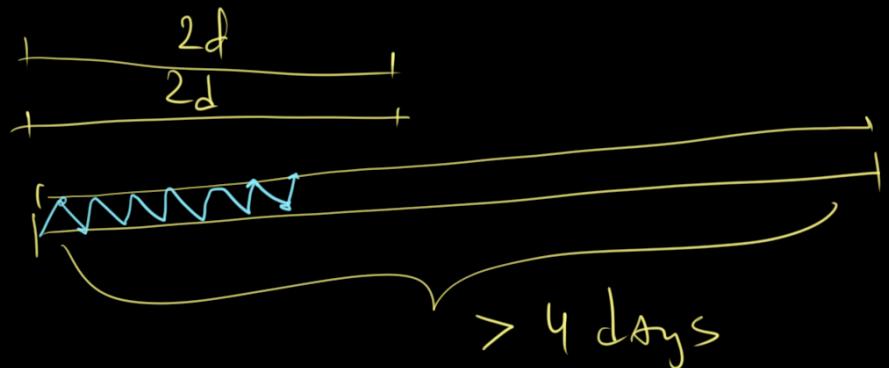
↳ wrong architecture  
 ↳ wrong implementation  
 ↳ whatever done  
 ↳ in a wrong way due to  
 ↳ deadlines

to keep tech debt on digestable level

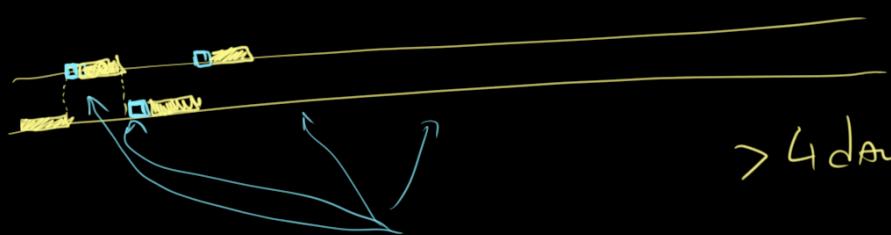


limit WIP (Work In Process)

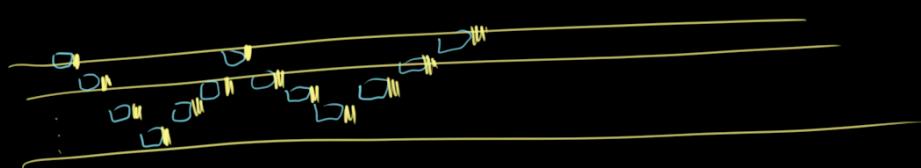
one task at a moment, no exceptions



definitely.



> 4 days ~5 days



} you'll never provide the result  
because of constant switch

