













## How can we lower the odds of YOUR project failing?

How can we make YOUR project suck less?



# Gather Information 31% Check expectations vs reality 21% Discover/reduce risk 21%

\* Credit To: Trish Khoo, \*Transparency: a Hollistic Approach to software Quality\*

#### Work towards the team goal 31% Look for Edge cases 21% Break the application 16%

\* Credit To: Trish Khoo, \*Transparency: a Holistic Approach to software Quality\*



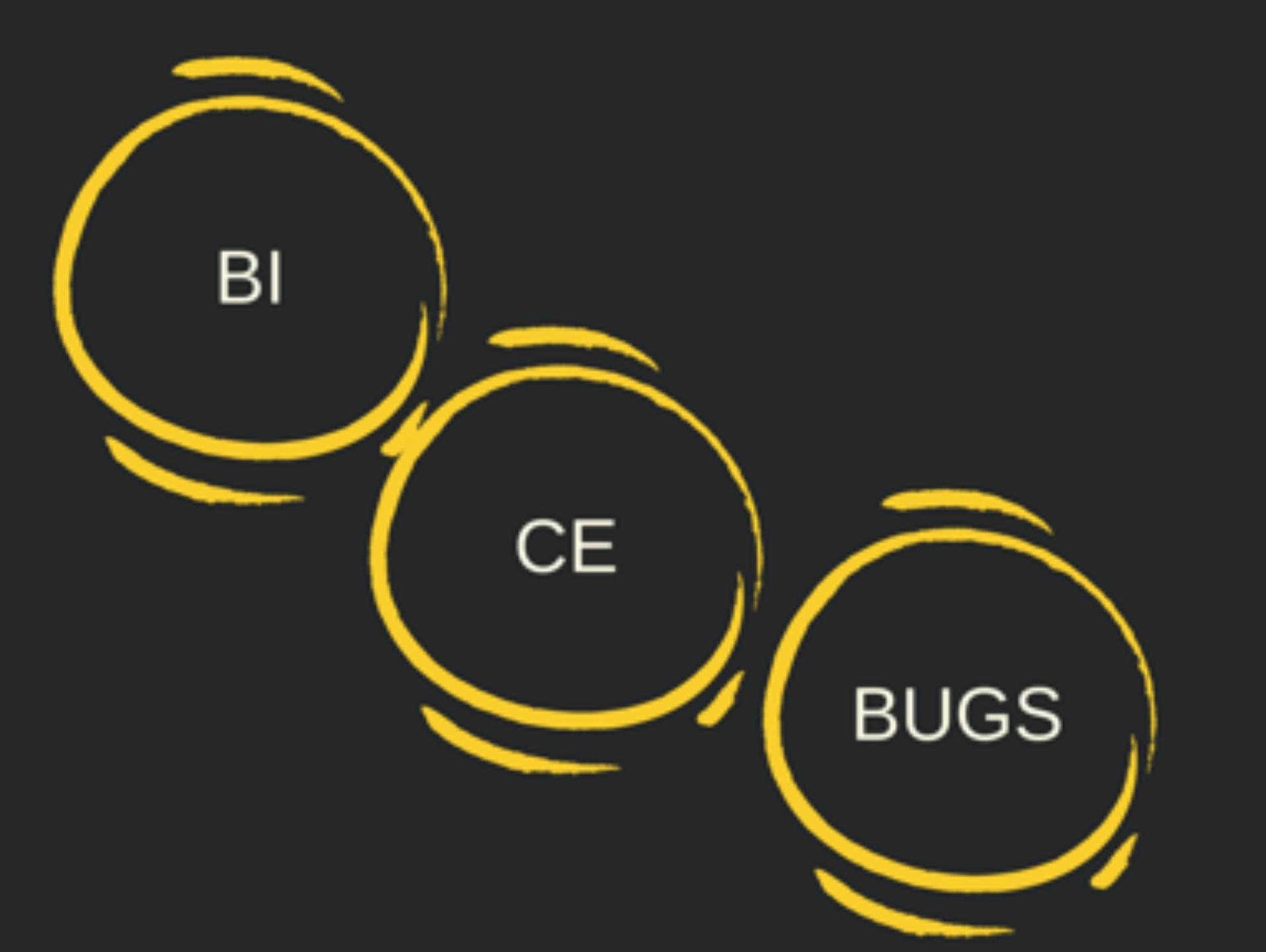




Does it solve a problem?





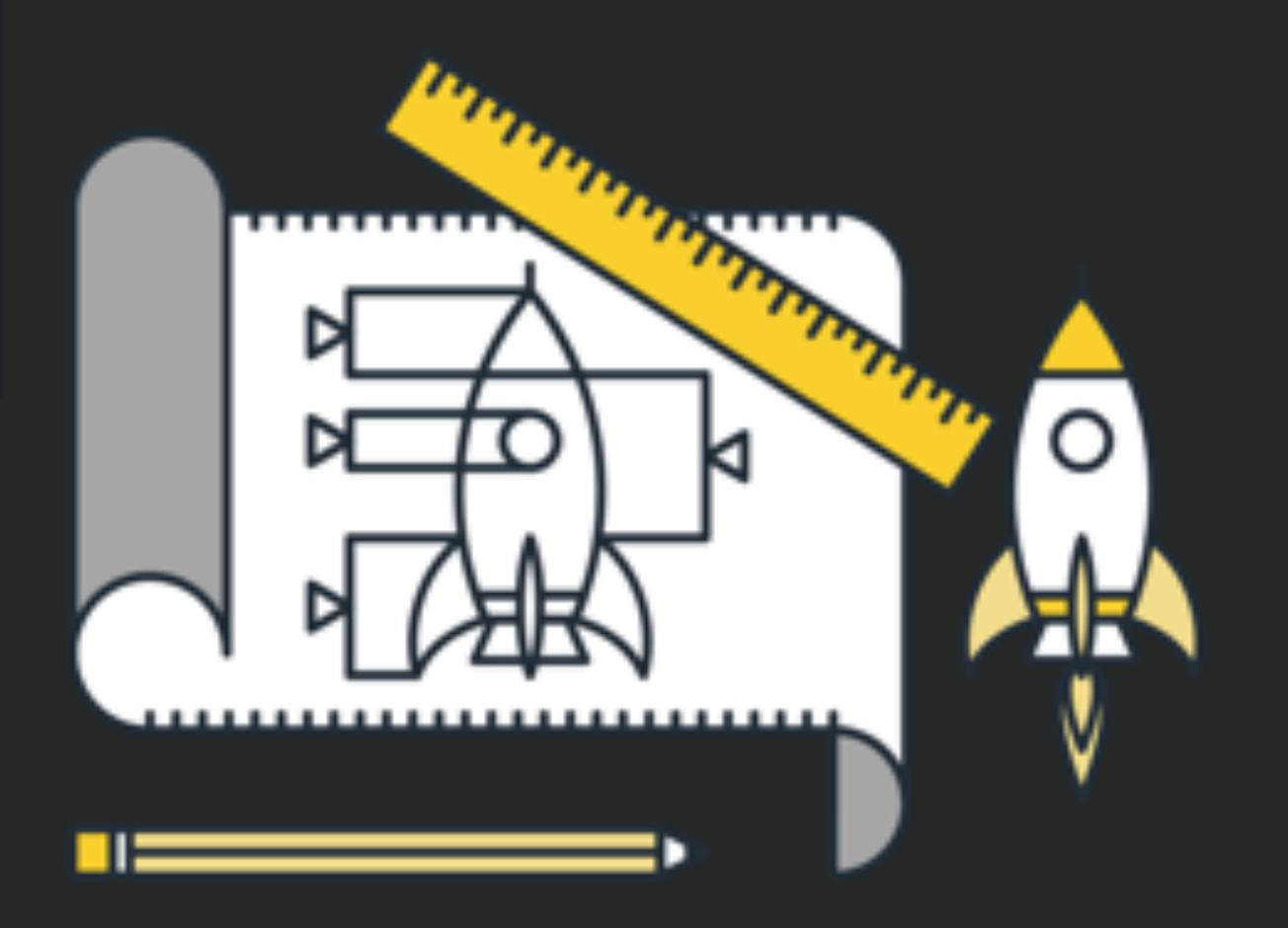


@anastasiachicu

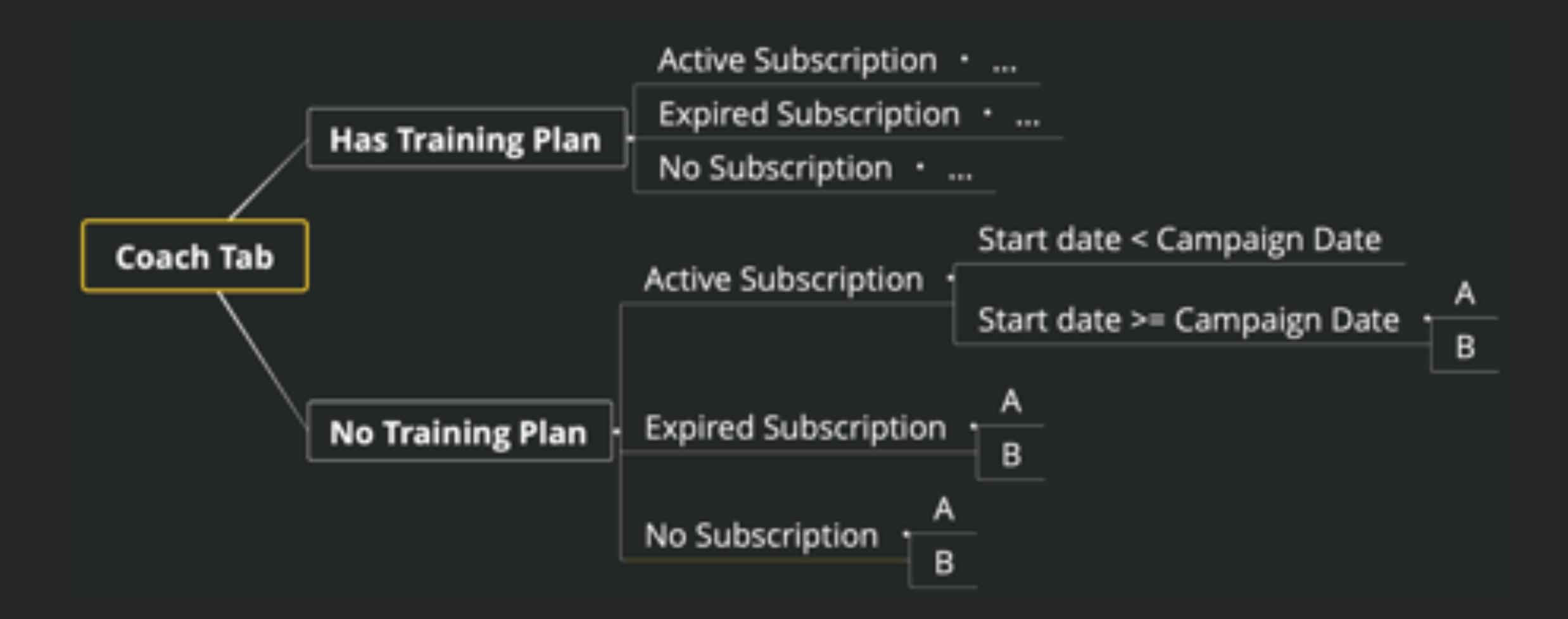
Does it solve a problem?

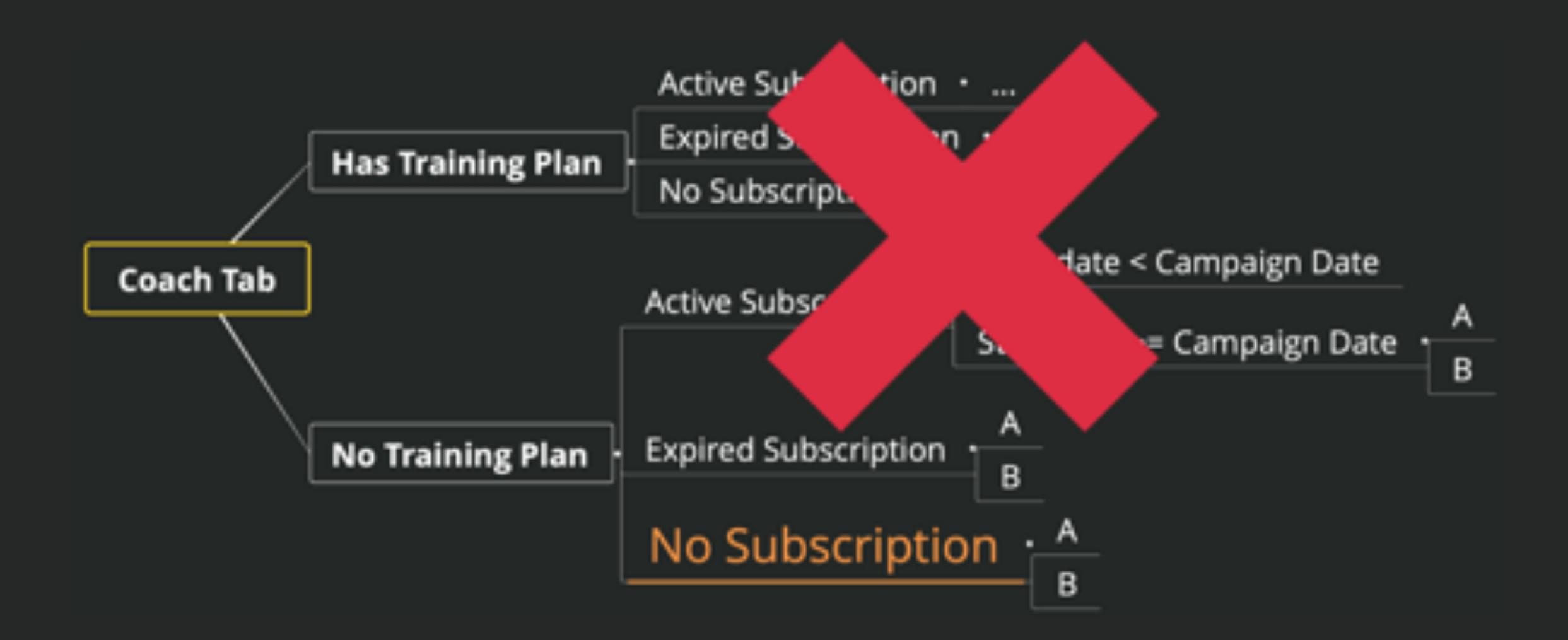
## Does it solve a problem? Is the problem clear?

# Does it solve a problem? Is the problem clear? How will the story add value?



## Functional Tests Examples Mind Maps



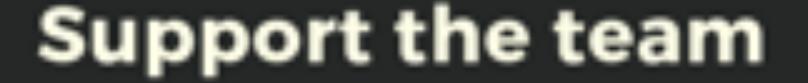






#### **Test Goals**







Criticise the product



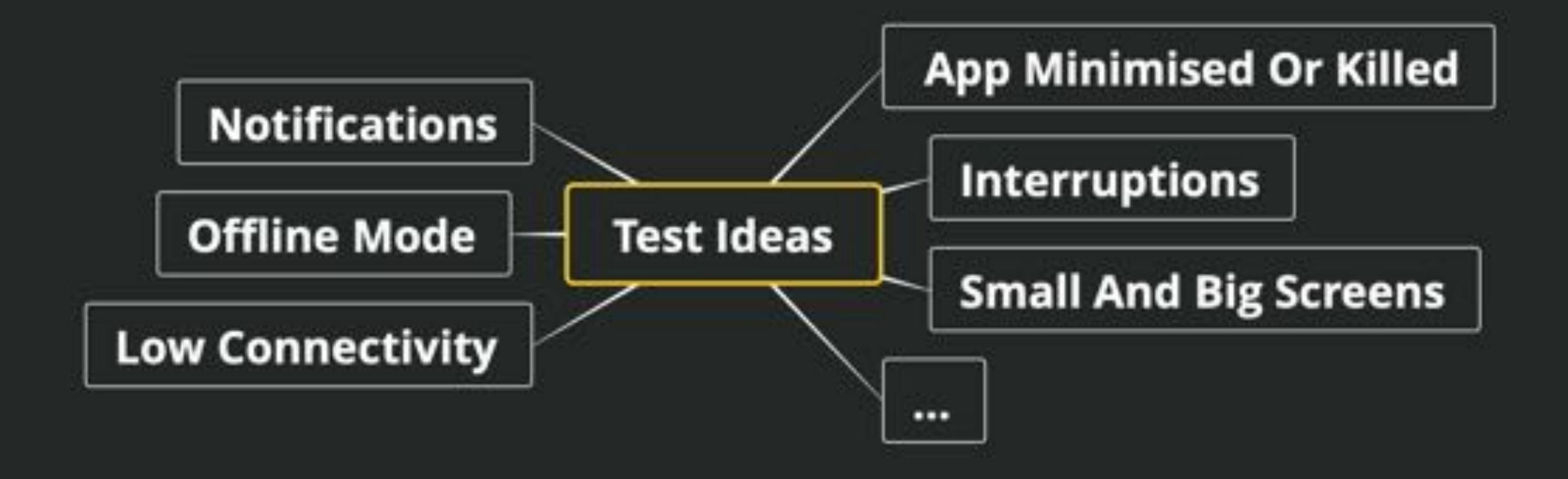
**Unit & Component tests** 



#### Exploratory Testing & Personas



- Jack, the training junkie
- Mafusail aged 57+
- Mary, the young mum

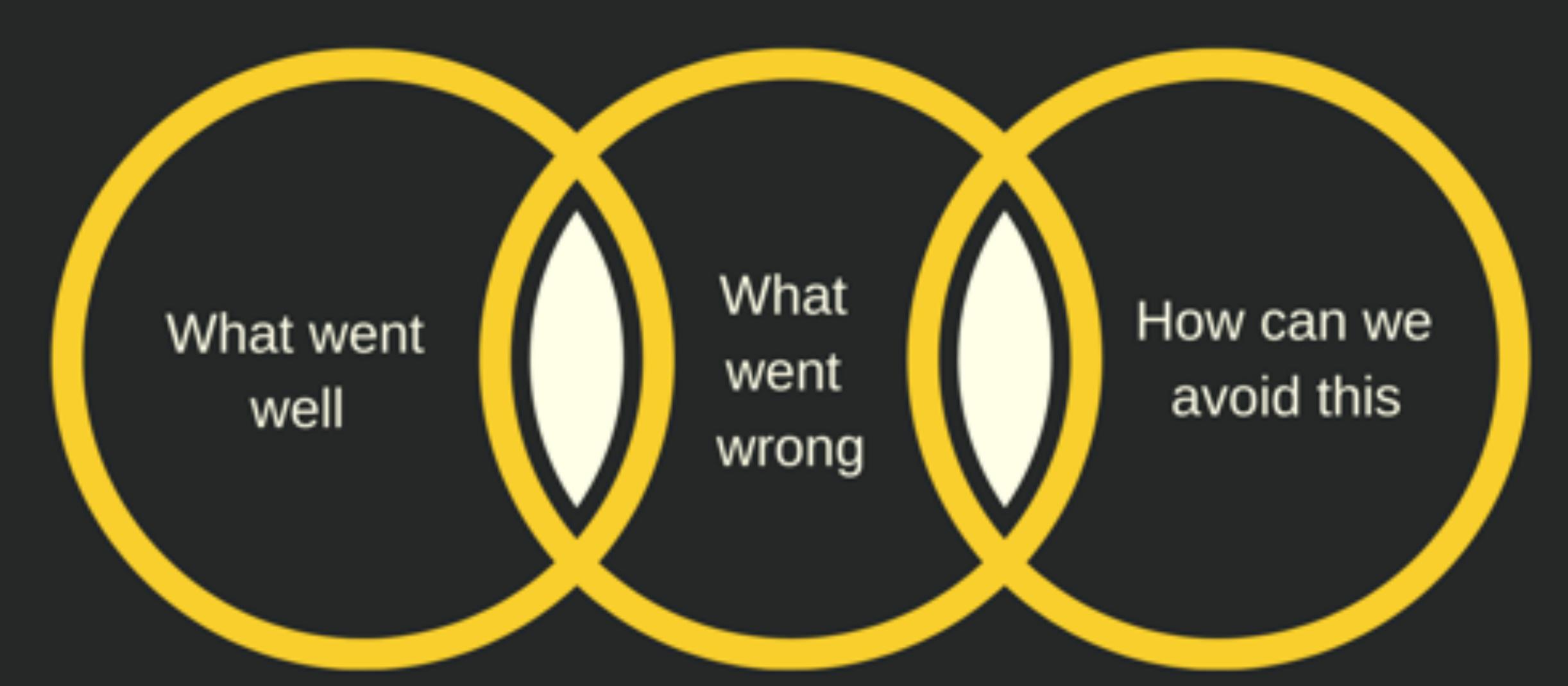








**Post-mortems** 





### Gather data

Quality Goals

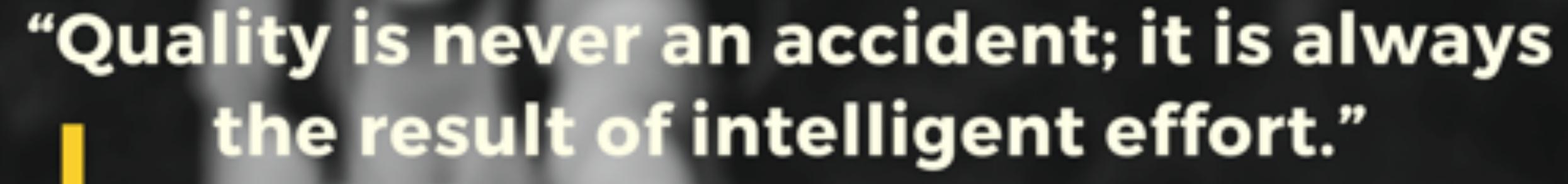
Story Mapping

Pair Testing

### Examples and Personas

Postmortems

Exploratory Sessions



- John Ruskin



#### REFERENCES

- https://martinfowler.com/bliki/TestPyramid.html
- https://www.iag.biz/business-analysis-benchmark-2008/
- Trish Khoo's talk about transparency https://www.youtube.com/watch? v=0MjkciWfQow
- Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations by Gene Kim, Jez Humble, and Nicole Forsgren
- https://github.com/dastergon/postmortem-templates