

**Institute** of  
**Data**

---

2025

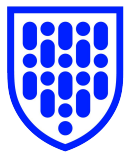


# Software Engineering

---

## Presenting Skills

---

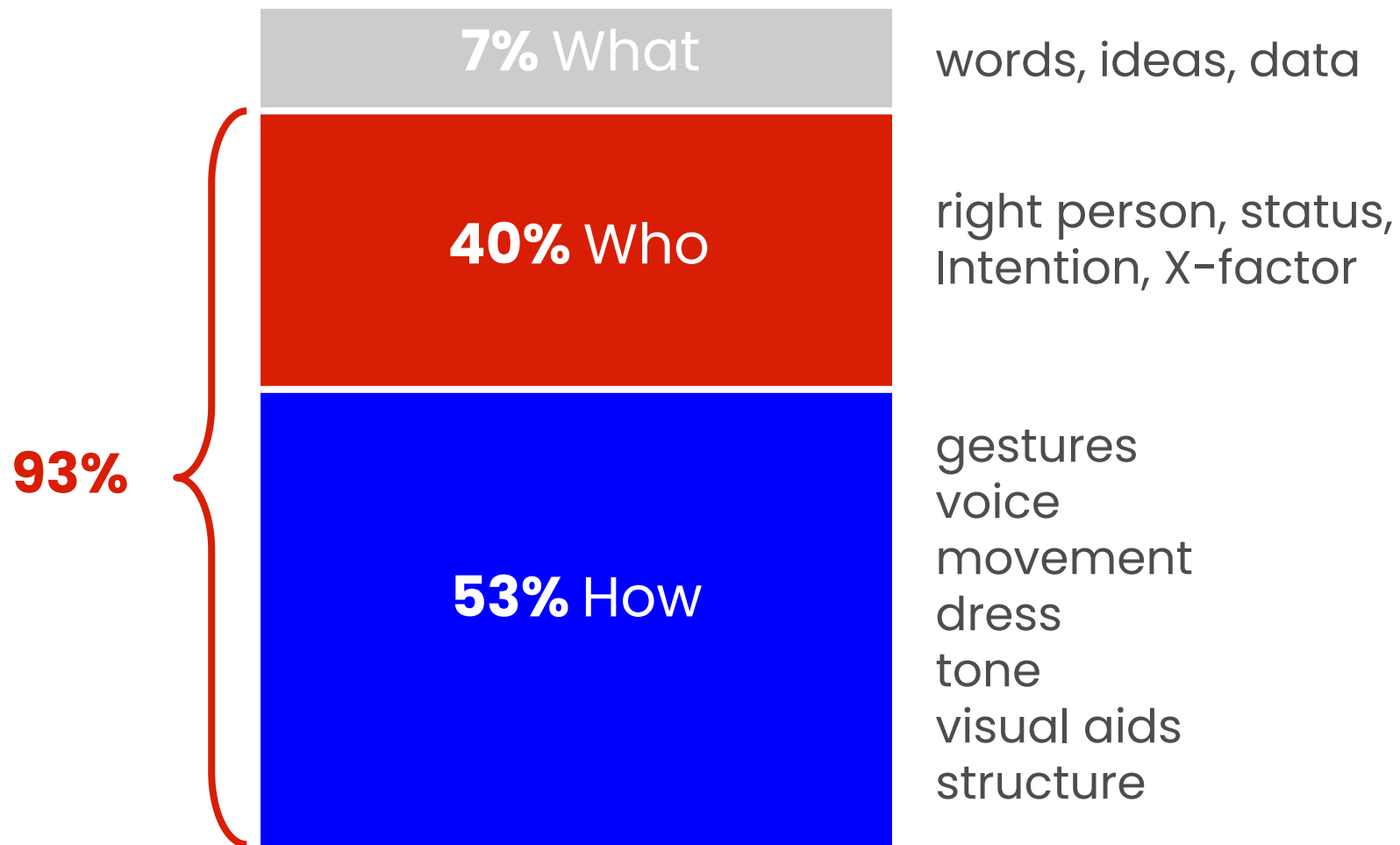


# Agenda

1. Planning frameworks
2. Delivery techniques
3. Practical exercises

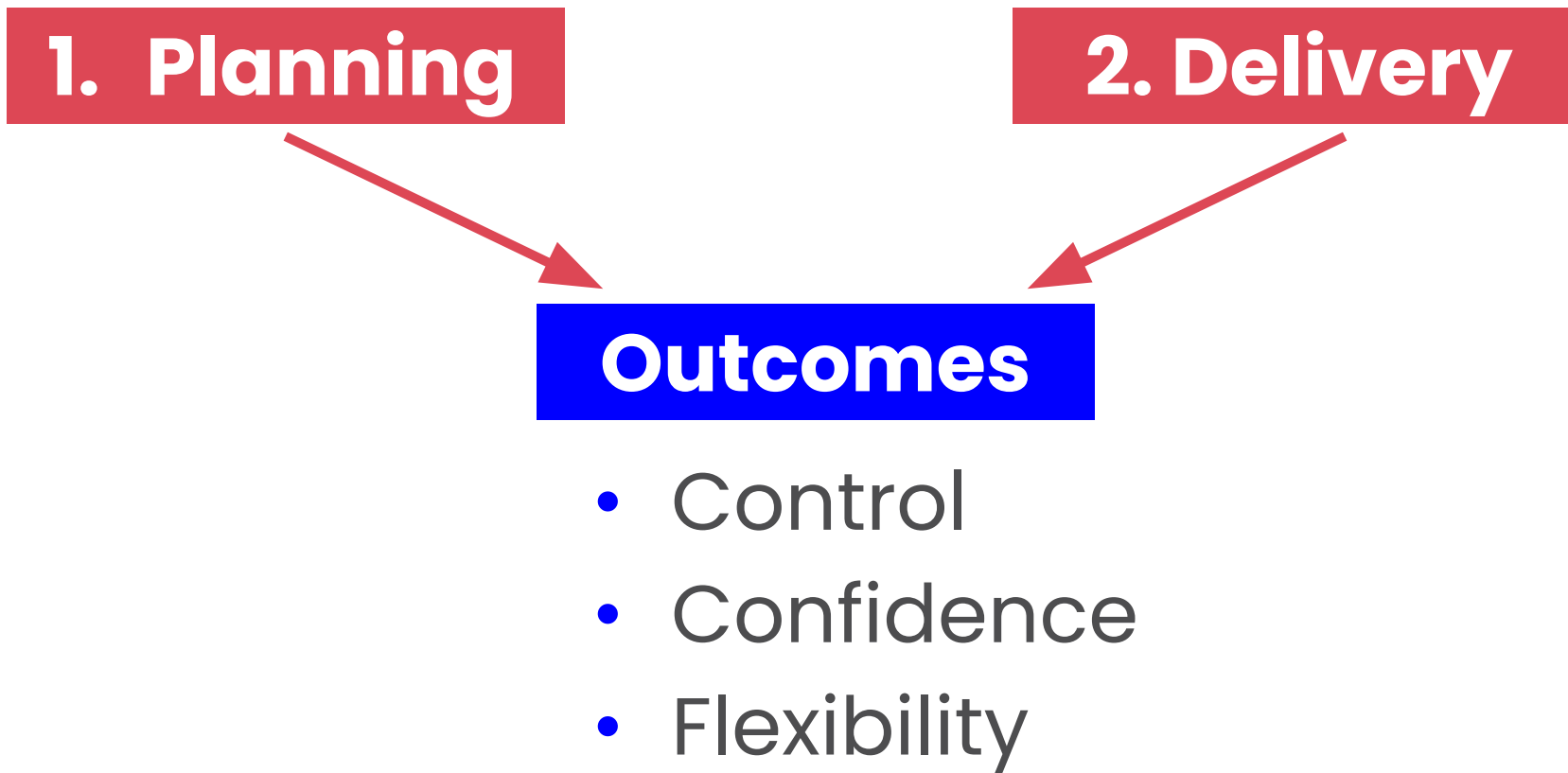


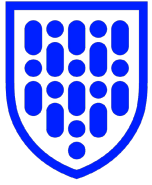
# Impact and Effectiveness





# The two elements of a good presentation





Institute of  
Data

# Planning Frameworks



# Presentation Formats

- Impromptu Outline
- Short Presentation Outline
- Formal presentation structure



# Essential Stages of Presenting

**1** Make sure the listener is ready to receive the message **Beginning**

**2** Communicate the message **Middle**

**3** Make sure the listener has received & understood the message **End**





# Impromptu Outline

**1** Tell 'em what you're going to tell them

**Beginning**

**2** Tell 'em

**Middle**

**3** Tell 'em what you just told them

**End**



# Short Presentation Outline

**1** Subject – State it

**2** Agenda – Areas to be covered

**Beginning**

**3** Body – Cover each area

**Middle**

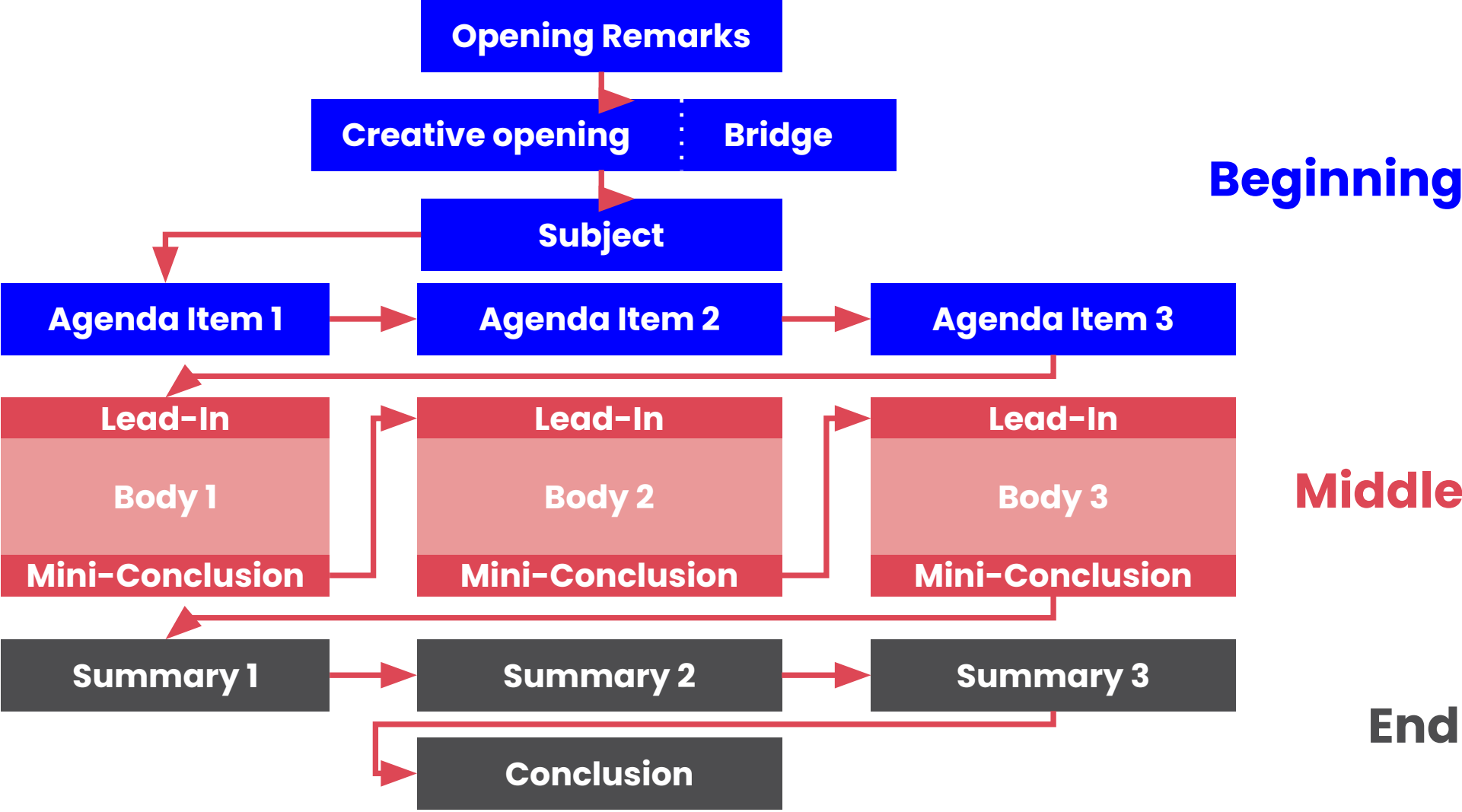
**4** Summary – Key idea from each area

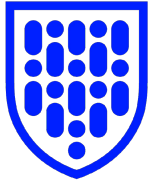
**5** Conclusion – Main idea, next steps,  
recommendation

**End**



# Formal Presentation Structure





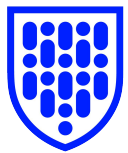
Institute of  
Data

# Delivery Techniques

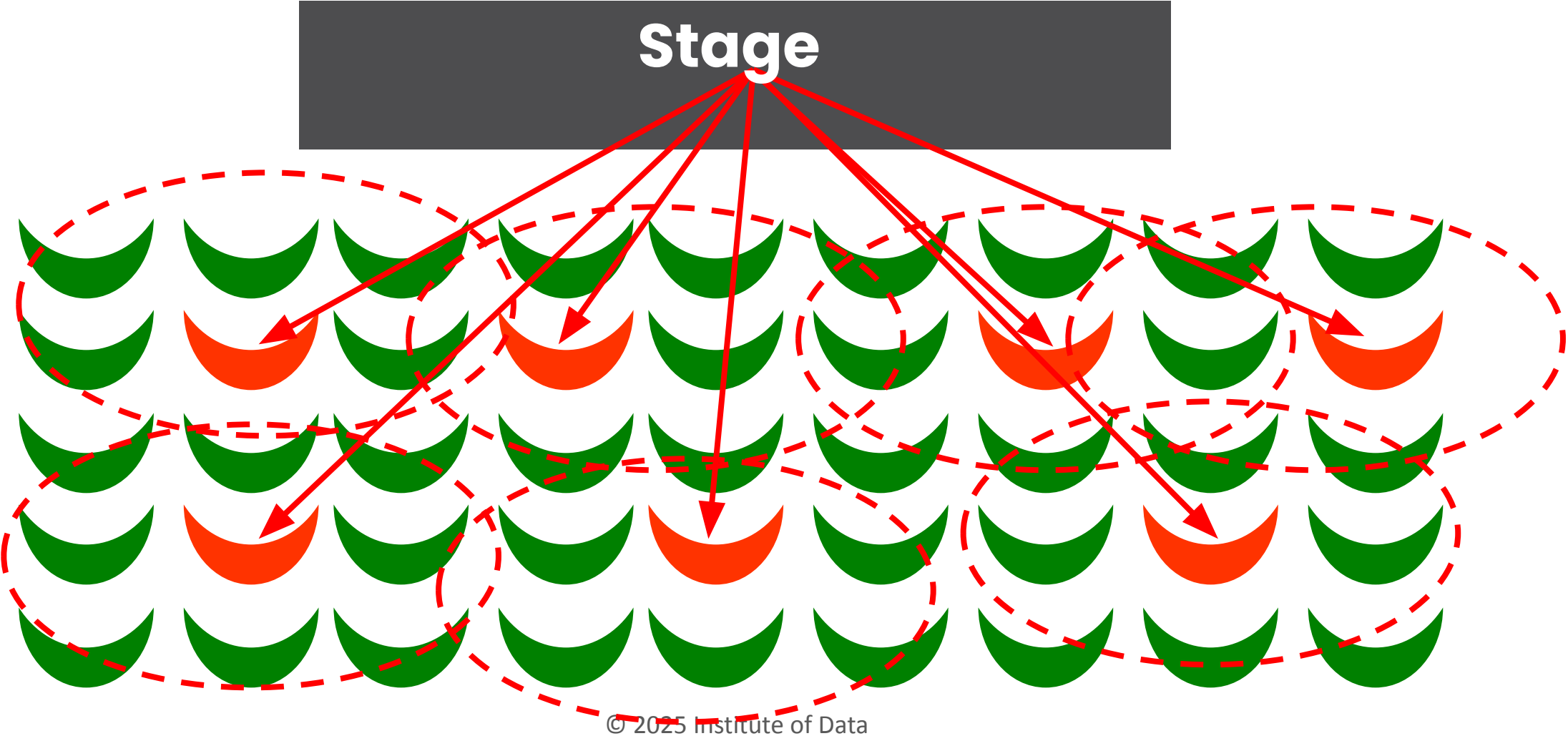


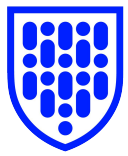
# The Impact of Eye Contact

Good Use	Poor Use
Trust Genuine Confidence You care Engage Show interest Competence Enthusiasm Authority	Dodgy Distrust
	Too Much
	Intimidating Unsettling Rude Overbearing



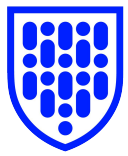
# Public Speaking Eye Contact Clusters





# Confident Manner

1. Eye contact
2. Use pauses and silence to your advantage - Say nothing instead of “Um”
3. Control your pace
4. Use hand gestures appropriately
5. Move to a new position to indicate a new idea



# Exercise: Touch, Turn, Talk

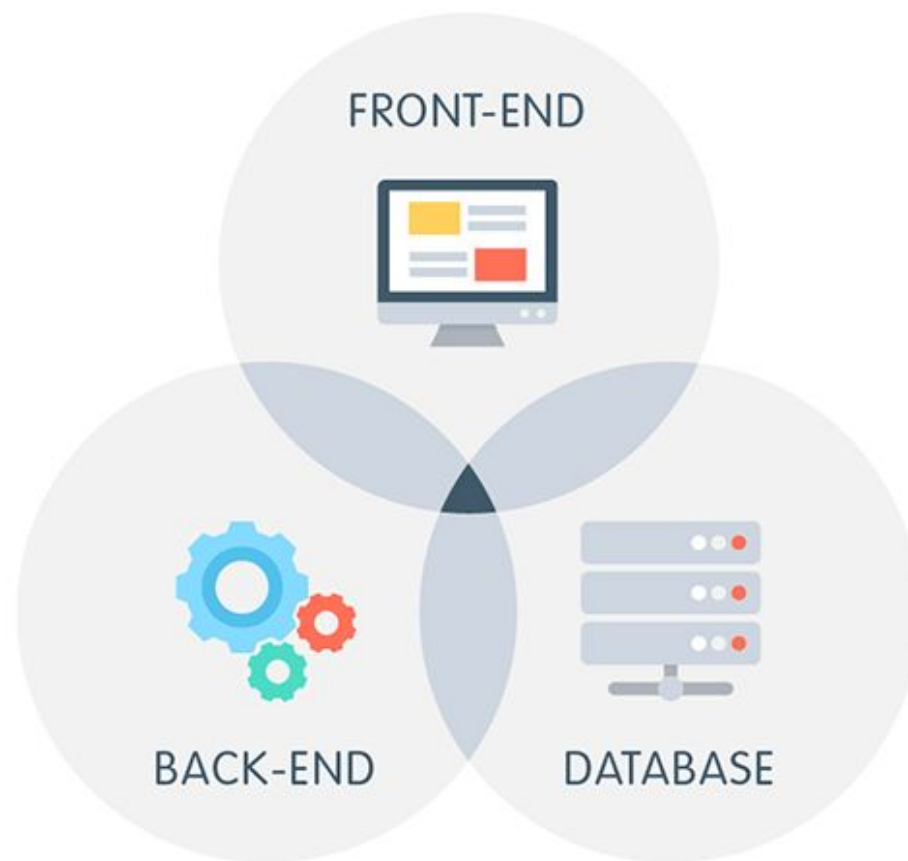
- Touch - click the clicker
- Turn - to the audience
- Talk - make eye contact and deliver the message

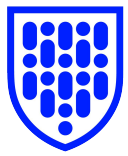
Present from one or more of the following slides



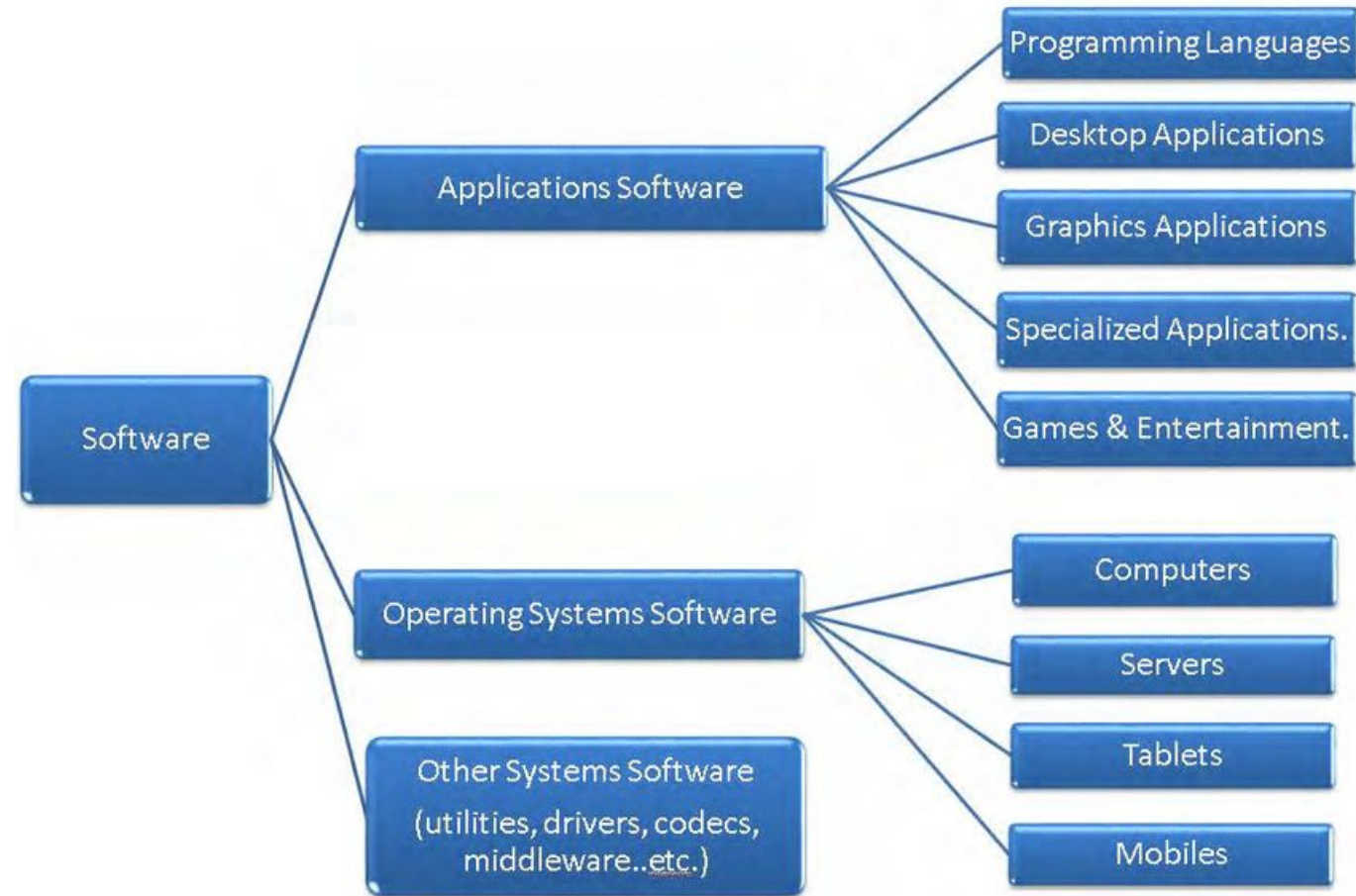


# FULL-STACK DEVELOPMENT



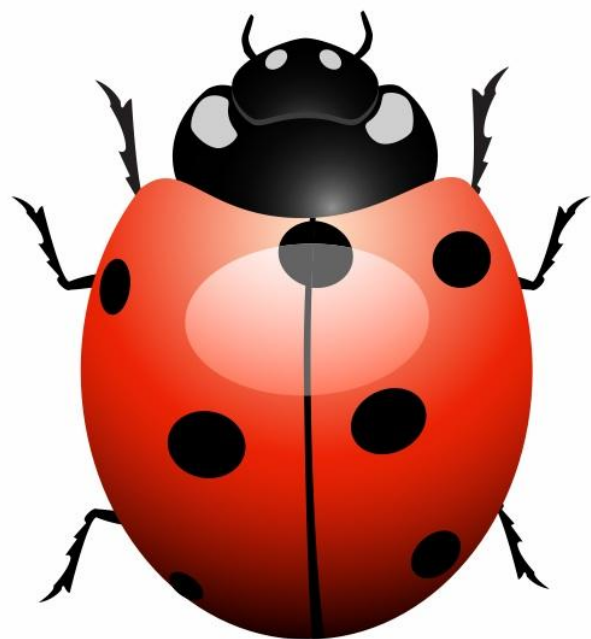


# Types of Software



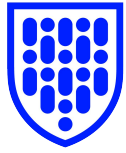


## Why do bugs arise?



- Unclear Requirements
- Incorrect Requirements
- Complex design
- Poor skilled developer
- Complicated logic
- Insufficient time
- Less knowledge about product
- Poor skilled tester
- Changing requirements often
- Environment & System issues

 [www.letzdotesting.com](http://www.letzdotesting.com)



# 7 COMMON TYPES OF SOFTWARE TESTING

## **BLACK BOX TESTING**

... test output of a given input...

## **WHITE BOX TESTING**

... test the source code ...

## **ACCEPTANCE TESTING**

... test what is expected vs. what actually happens

## **AUTOMATED TESTING**

... recurring standardized tests with scripts ...

## **REGRESSION TESTING**

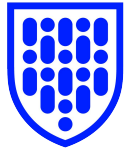
... test to verify that the system still works the way it did before ...

## **FUNCTIONAL TESTING**

... test all things related to functionality ...

## **EXPLORATORY TESTING**

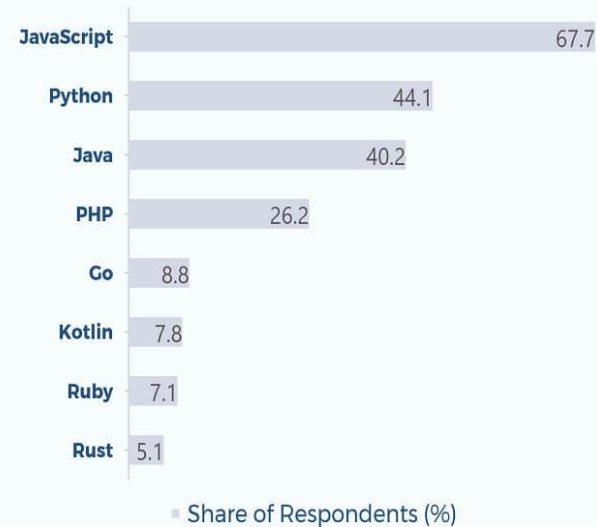
... test within certain areas, no specified test cases ...



# The Best Backend Programming Languages



## Most Used Programming Languages Among Developers Worldwide

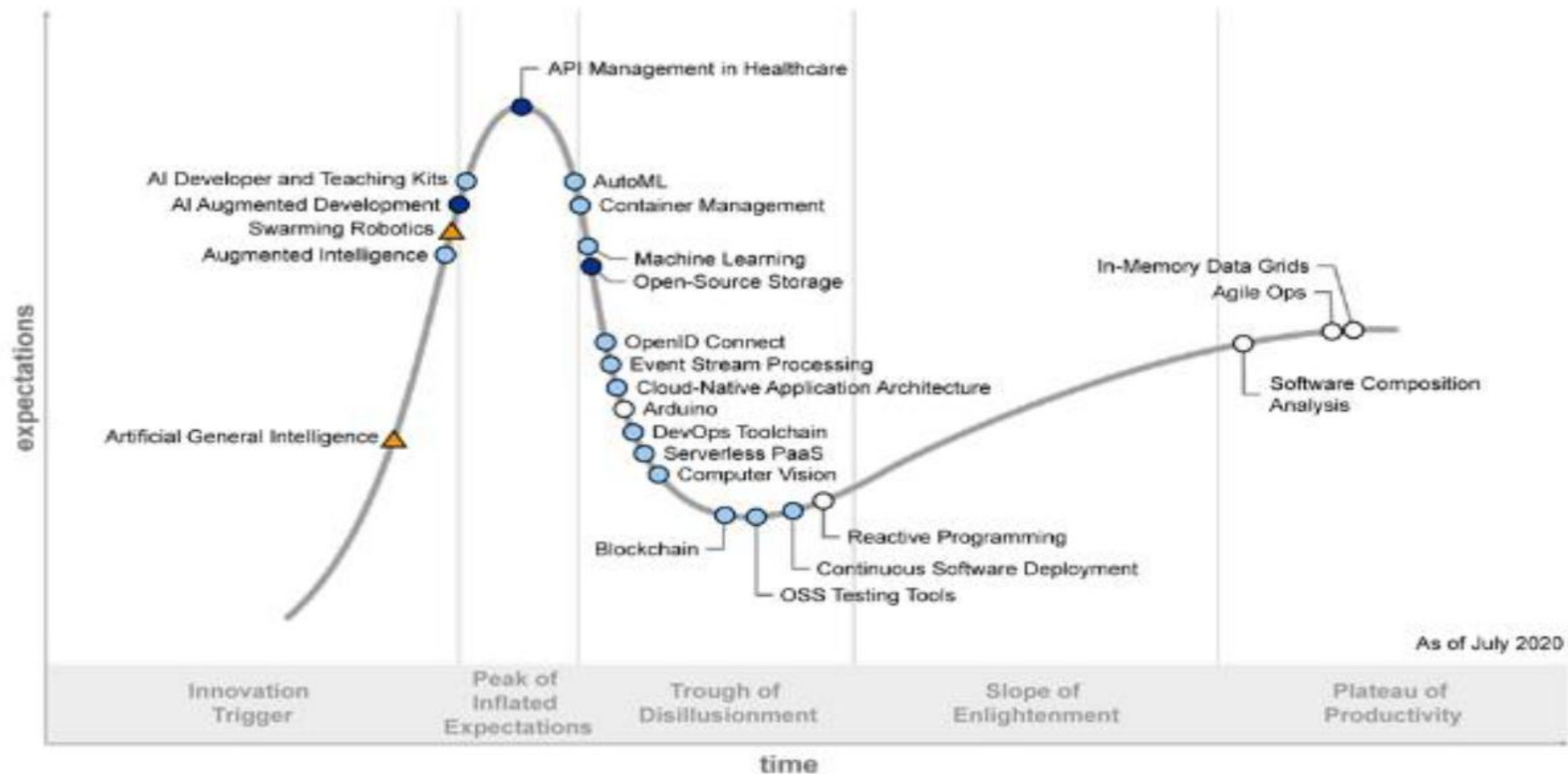


Source: A Survey Published by Shanhong Liu (2020)





# Hype Cycle for Open-Source Software, 2020



Plateau will be reached:

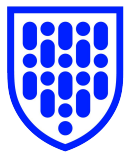
○ less than 2 years    ● 2 to 5 years    ● 5 to 10 years    ▲ more than 10 years    ✗ obsolete before plateau

Source: Gartner  
ID: 450395



# 3 things I like about Cats





# Wrap up

Today we covered:

1. Planning frameworks
2. Delivery techniques
3. Practical exercises

Do that and you'll present with Impact and Effectiveness.





# Exercise: Capstone Project Presentation

- Revise Capstone Presentation in light of short presentation outline or formal presentation structure
- Present!