Software Engineering Essentials

Prototyping

Bernd Bruegge, Stephan Krusche, Andreas Seitz, Jan Knobloch
Chair for Applied Software Engineering — Faculty of Informatics

Learning goals

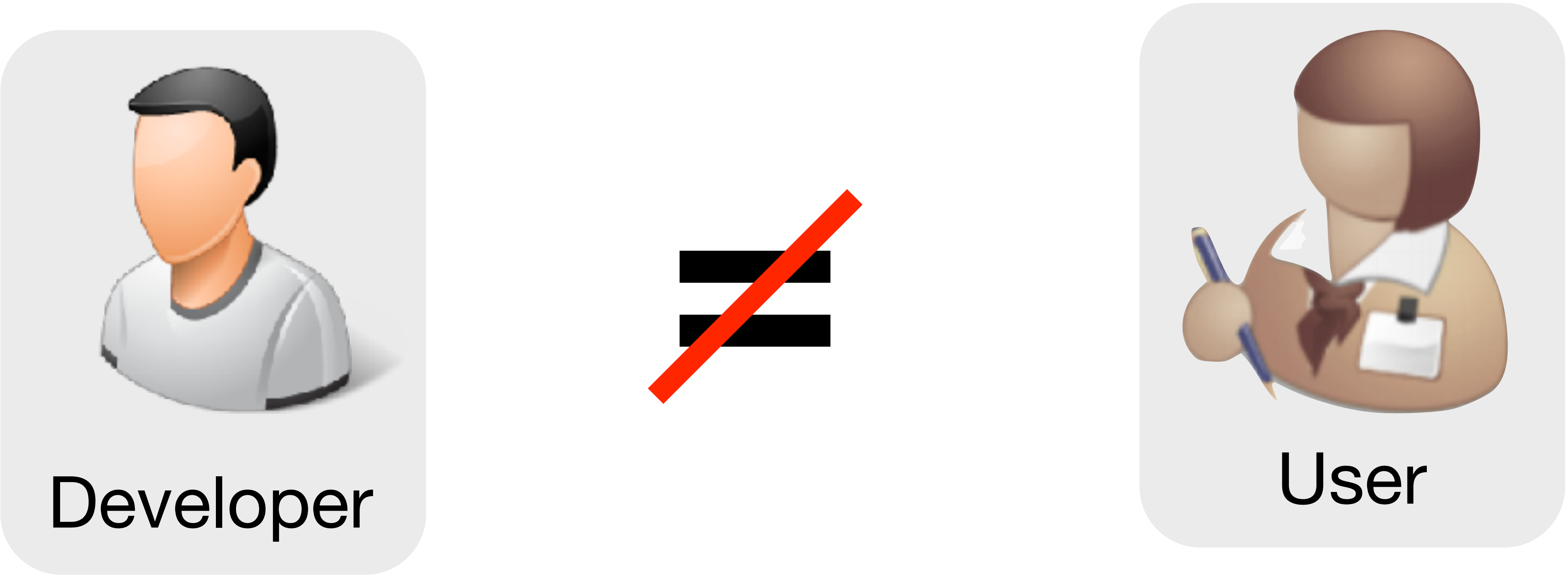
1) Understand the importance of prototyping for software projects

2) Explain the iterative prototyping approach from low to high fidelity prototypes

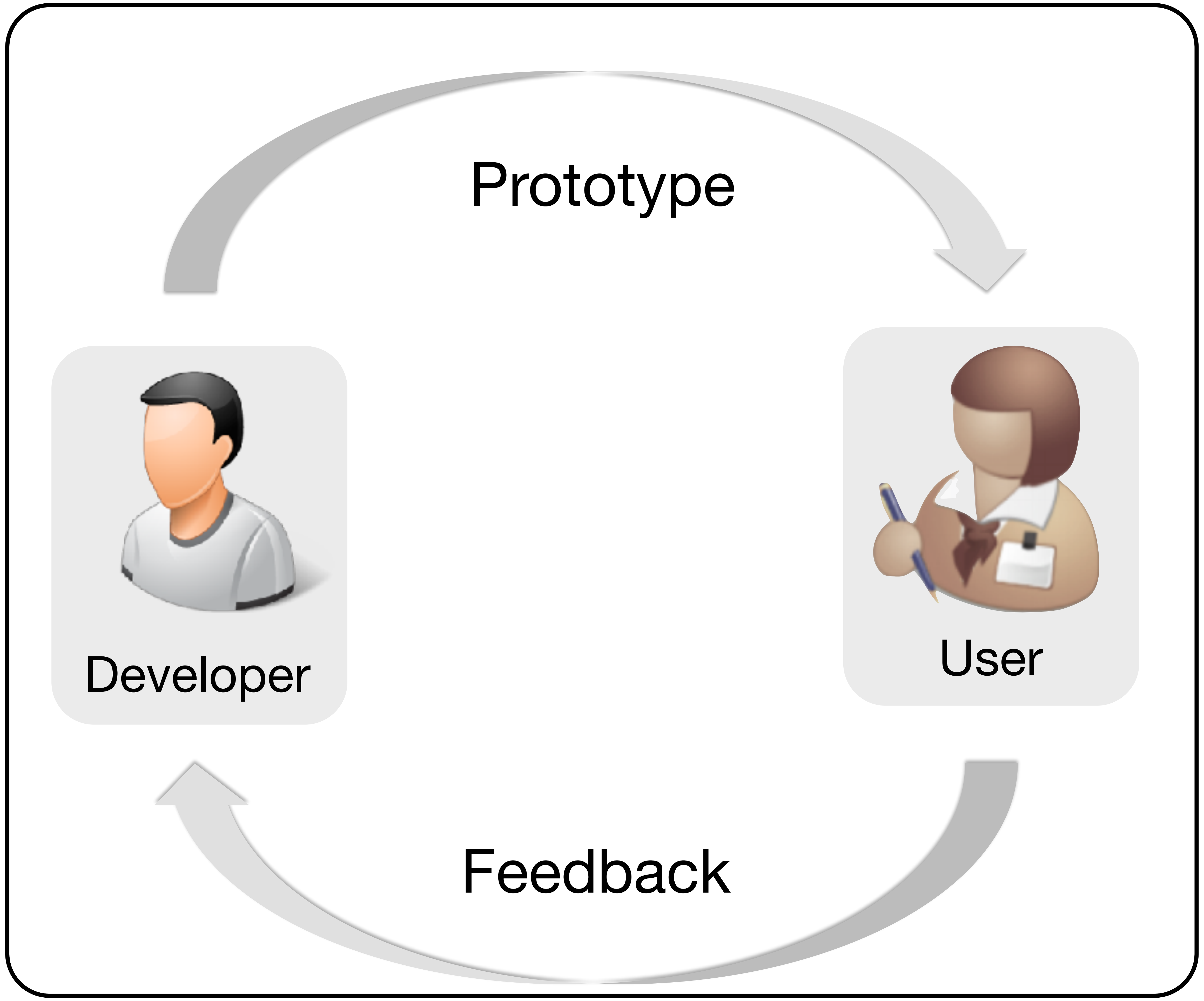
3) Compare different types of prototypes, e.g. horizontal vs. vertical

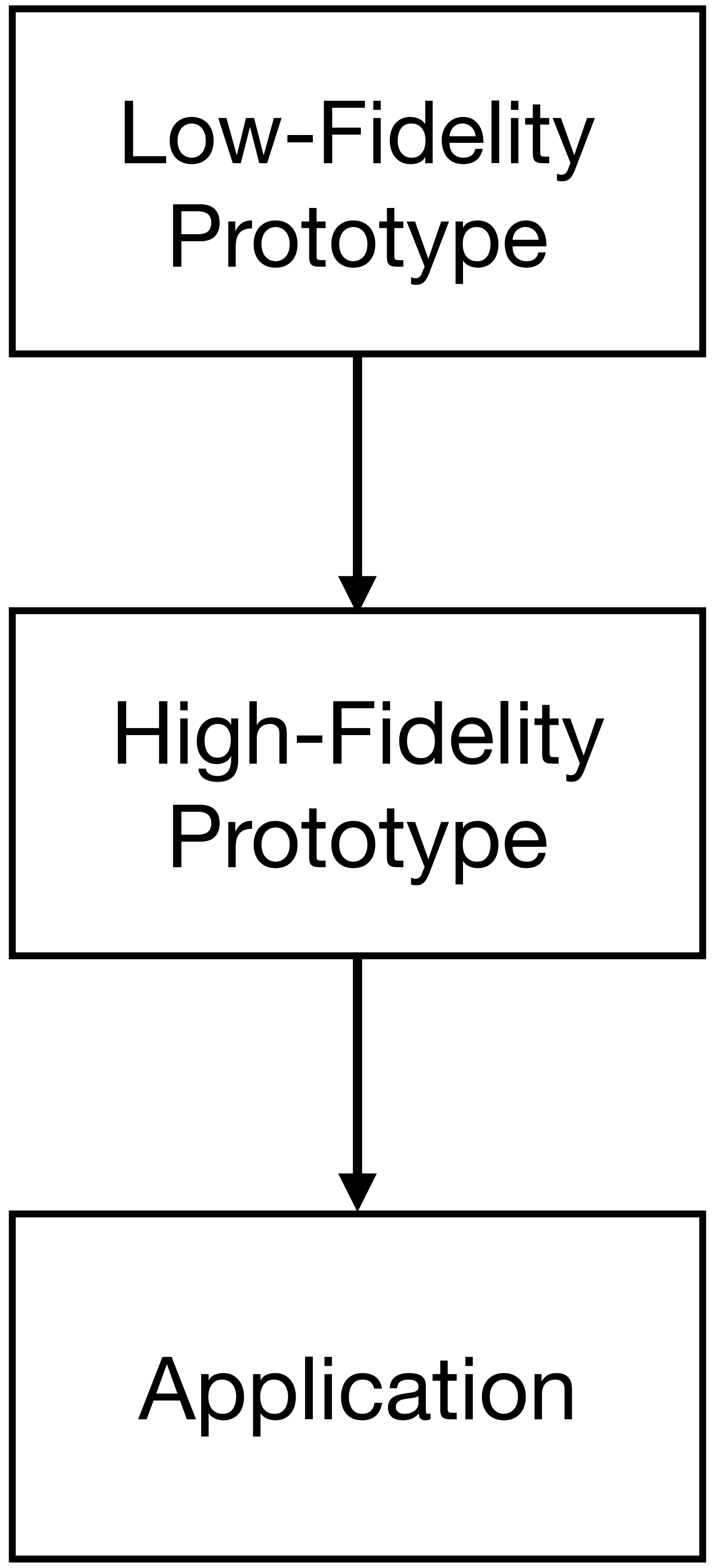
Motivation - why do we need prototyping?

- Complex systems and user interfaces are hard to design and to develop
- Communication between developers and users is necessary
- Several iterations are required to meet user expectations

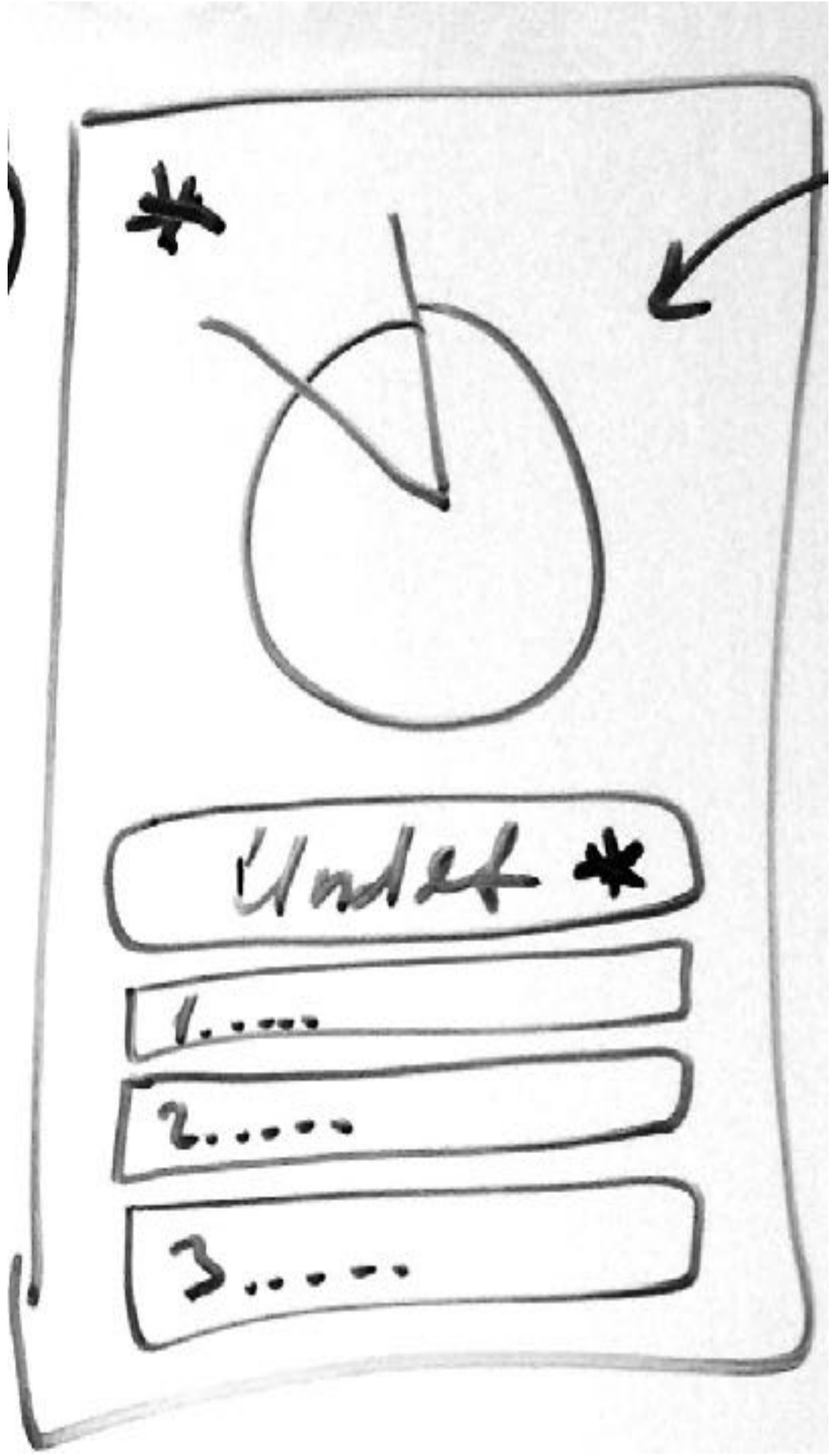


Prototyping is an iterative approach

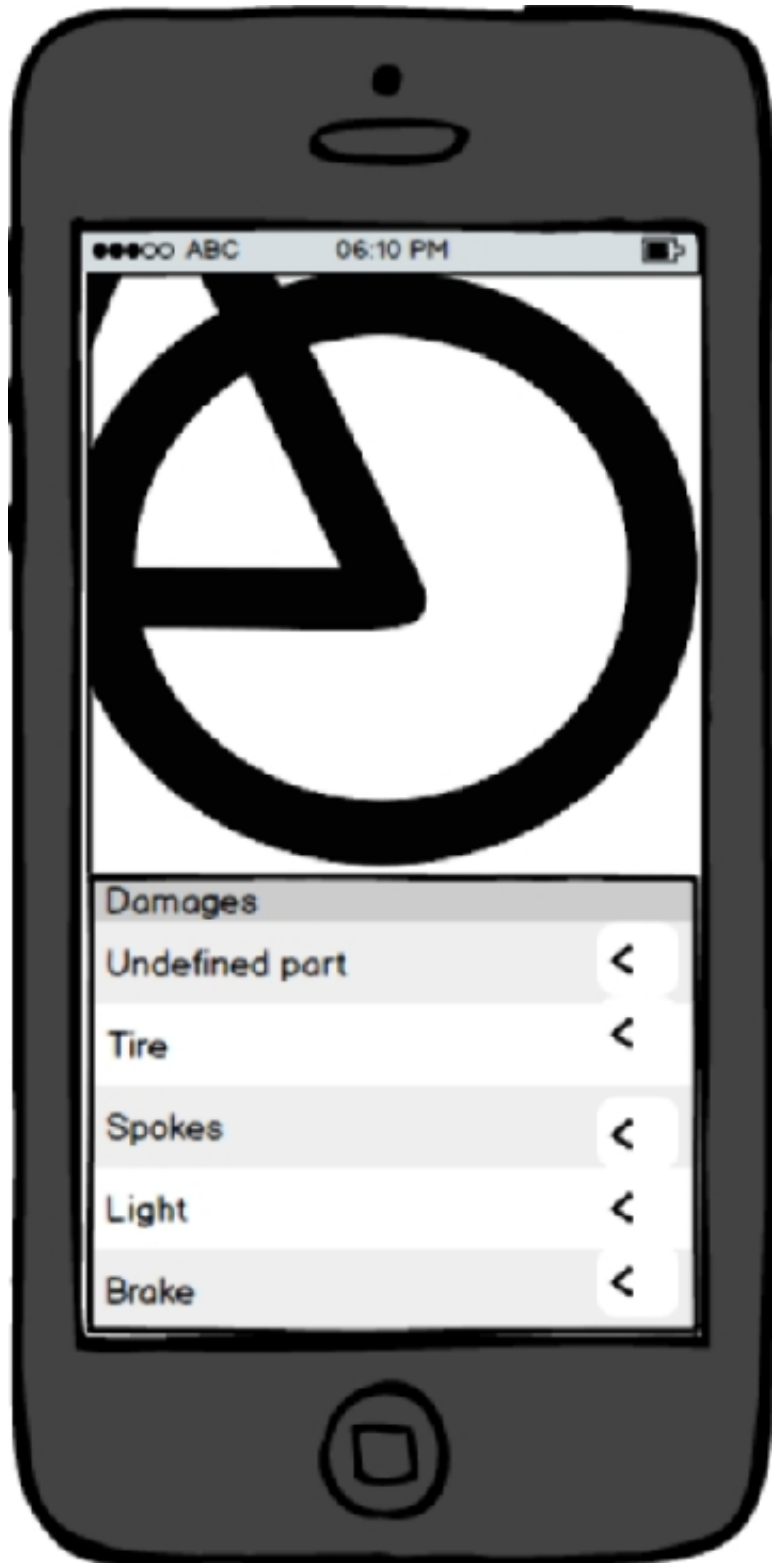




From low to high fidelity prototyping



Low fidelity



Damages

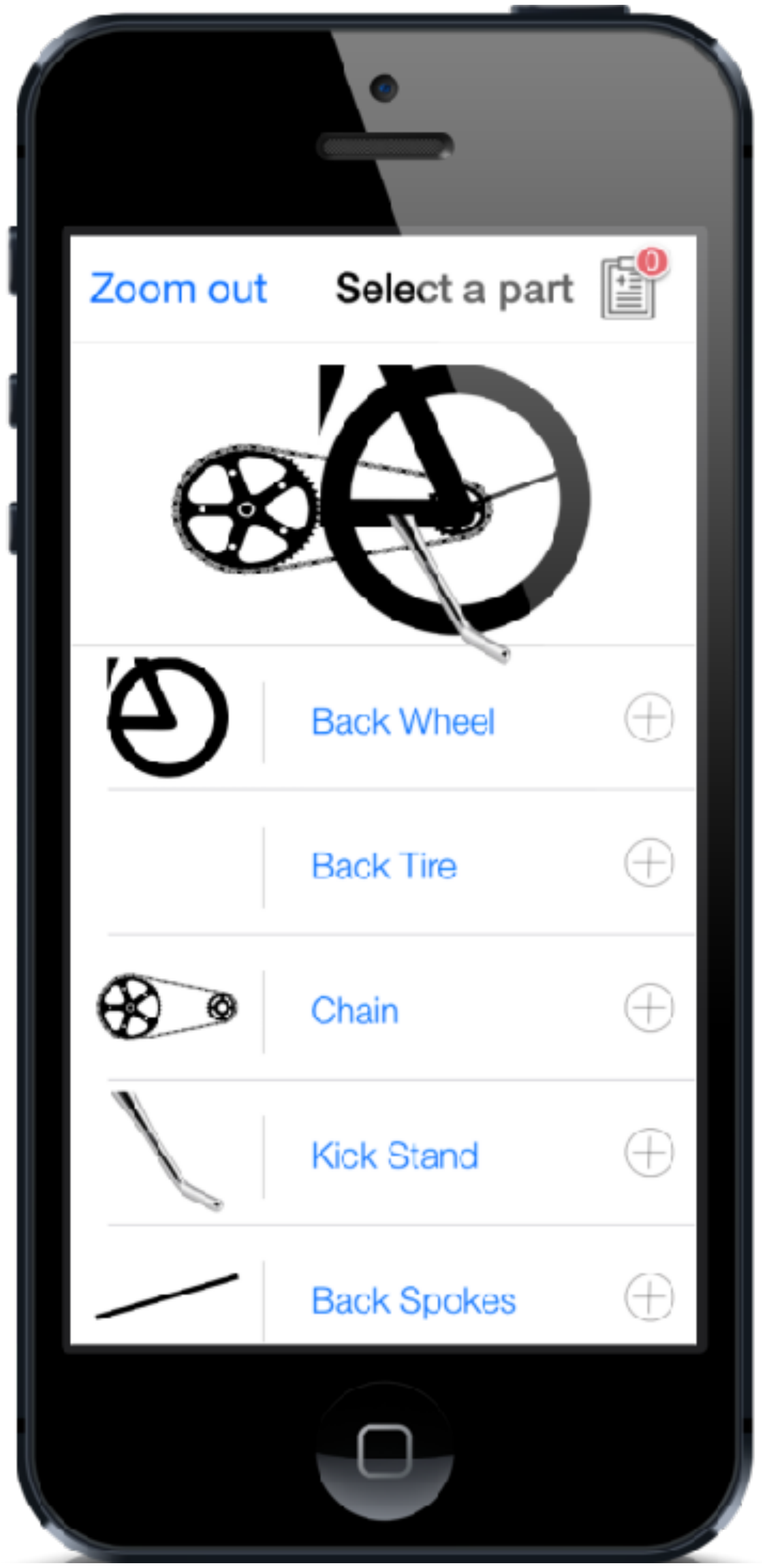
Undefined part <

Tire <

Spokes <

Light <

Brake <



Back Wheel ⊕

Back Tire ⊕

Chain ⊕

Kick Stand ⊕

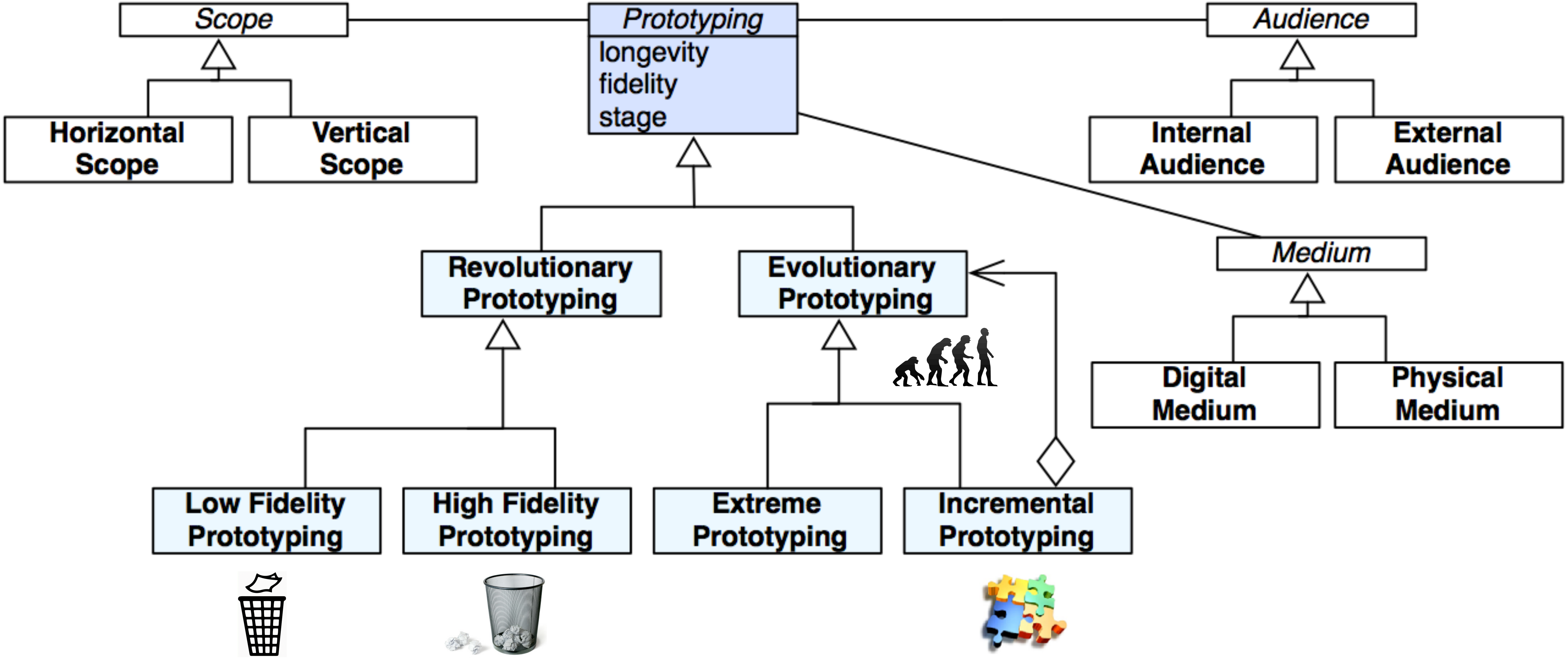
Back Spokes ⊕

High fidelity

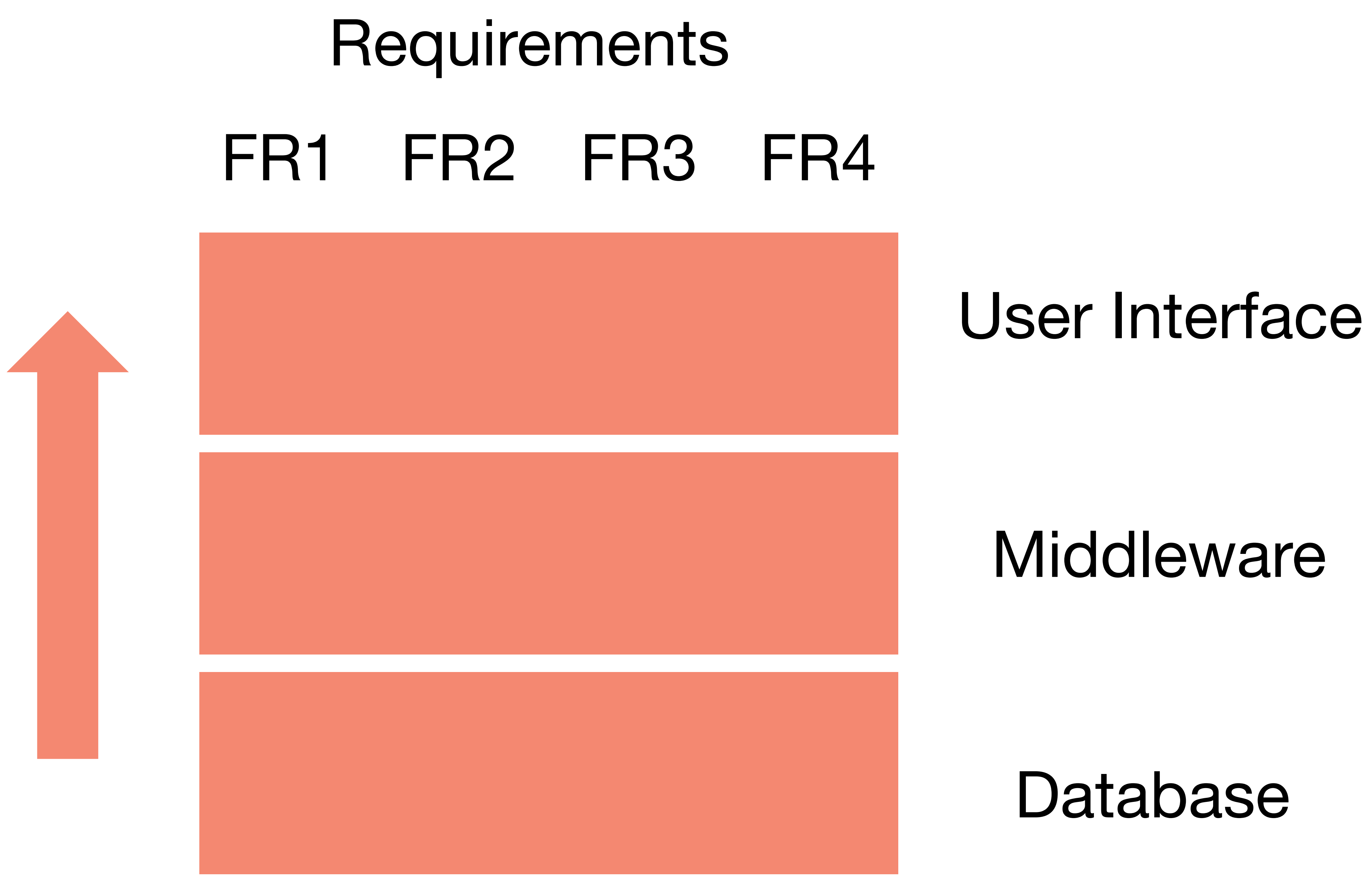
Low vs. high fidelity prototyping

	Advantages	Disadvantages
Low Fidelity	• Easy to produce • More feedback • No design decisions	• Mostly not reused • Important details are ignored
High Fidelity	• More realistic • More detailed problems can be identified • More impressive	• May cause much effort • Less feedback • High expectations (especially with interactive prototypes)

Different types of prototypes



Horizontal prototypes

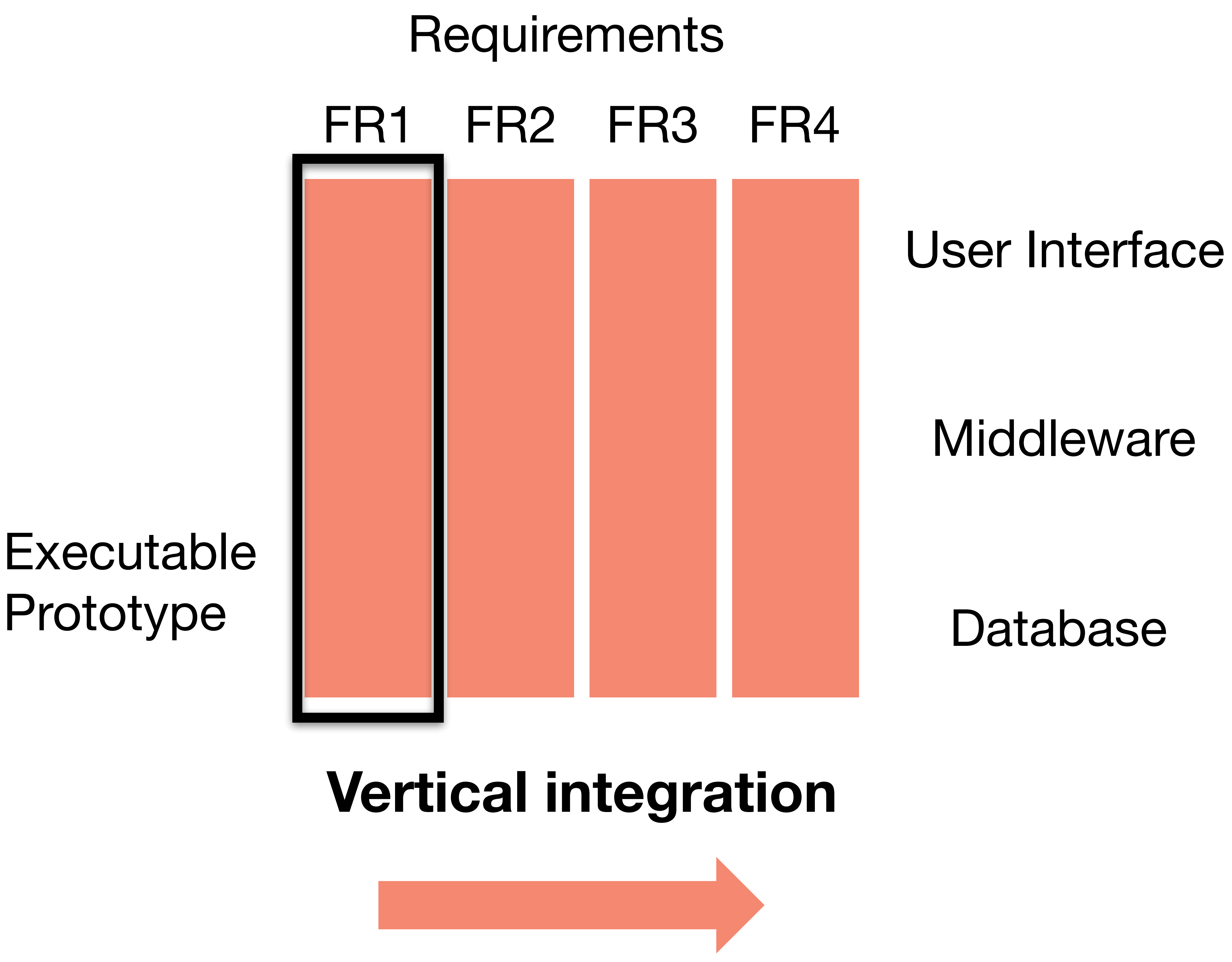


Horizontal integration: Bottom-up

Horizontal Prototypes

- Show wide range of requirements
- Horizontal integration
  - Bottom up or top down
- Used in traditional, linear processes:
  - No full implementation of the requirements until the end

Vertical prototypes



Vertical Prototypes

- Show small range of requirements
- Full implementation of these requirements
- Vertical Integration
- Used in agile processes

Boyle’s law

“Never go to a meeting without a prototype”

— Dennis Boyle, IDEO

Software Engineering Essentials

Prototyping

Bernd Bruegge, Stephan Krusche, Andreas Seitz, Jan Knobloch
Chair for Applied Software Engineering — Faculty of Informatics