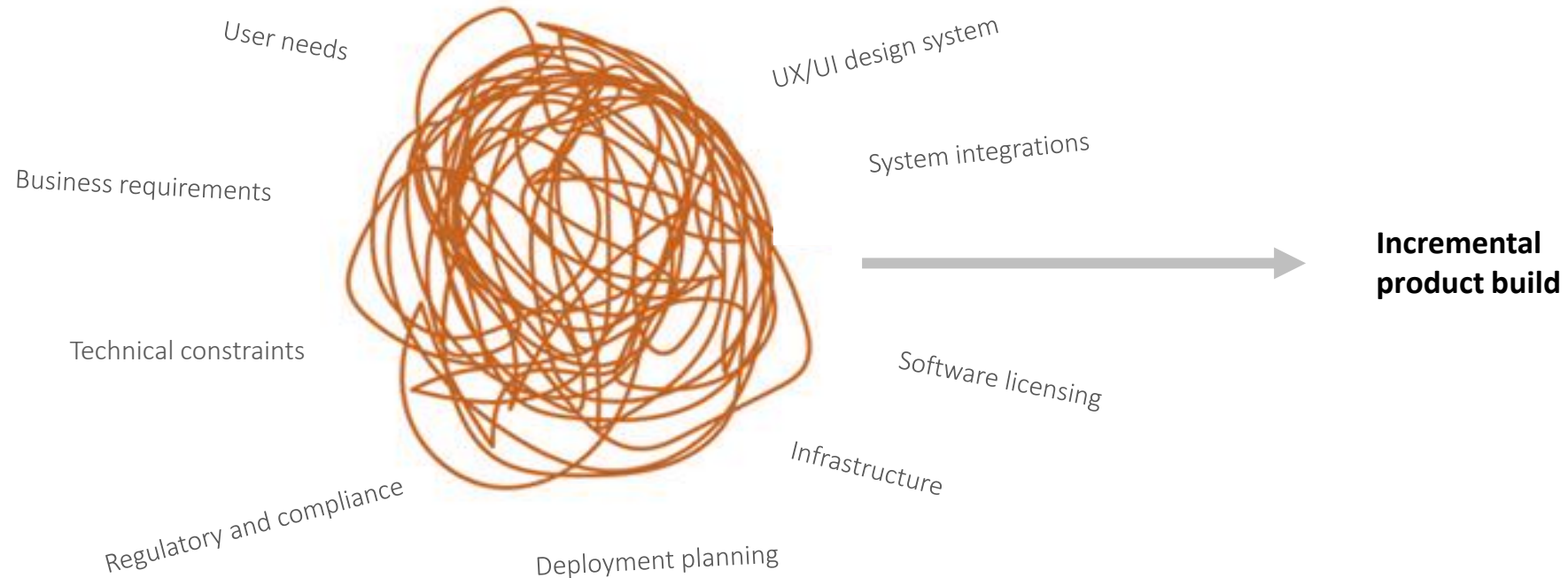


Software Requirements for Agile Development Teams

Building software is complex, messy and time consuming

And even experienced teams can have trouble knowing where to start

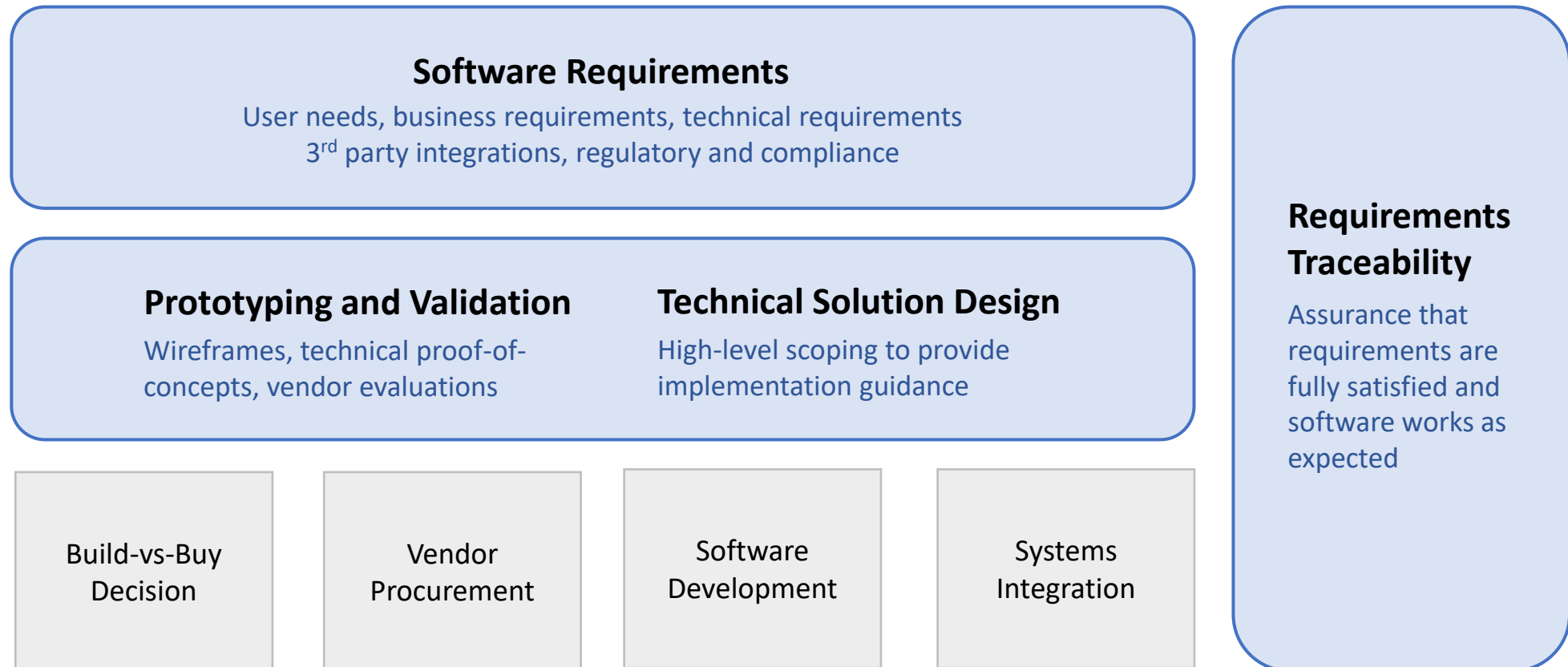


Our carefully developed product backlog will ensure all your team members know exactly what to do

And our retained support will keep it that way

Service Offering

We provide everything a software development team needs to thrive



Core service proposition



Supported activities

(nb. an agile approach is assumed by default)

Best practices for software requirements

What the research says

Requirement issues are common place [1]

- Incomplete or hidden requirements
- Inconsistent requirements
- Unclear non-functional requirements
- Communication flaws between project and customer
- Communication flaws with the project team
- Difficulties separating requirements from previous solutions

Software development is highly tailored [2]

- Software engineering is highly applied across a very large range of situational contexts
- No one software development process or set of practices is universally suited to all software development settings
- Most project disciplines show a clear trend towards an agile implementation

What to do about it

Software requirements

- Clear and unambiguous
- Validated with the right stakeholders
- Provided at the right time
- Provided in the right format
- Provided at the right level of detail

Software requirements process

- Responsive to the customer needs
- Responsive to the business stakeholders
- Responsive to the technical team
- Founded on agile practices
- Embedded as the way of working

[1] *Naming the Pain in Requirements Engineering*, a globally distributed family of surveys on Requirements Engineering practices and problems. [source](#)

[2] *What Makes Agile Software Development Agile?*, a large-scale international survey to investigate software development methods. [source](#)



Frank Ray

I'm really passionate about developing excellent software

As a former software developer myself, I know what it's like to receive rubbish, incorrect or simply no requirements at all

Agile gives us everything we need to thrive in an industry dogged by uncertainty, rapid change and last minute requests

Meaningful conversations, early prioritisation and deferral of detail are examples of valuable practices to adopt

The simplicity of the user story format helps bring them to life