

Some more terminology is needed

- Function(al requirement) – WHAT main function the system (product) shall have
- User function(al requirement) – WHAT interaction with our system (product) we shall have (input/output)
- Non-functional requirements – qualities about our system; cybersecurity, modularity, premium quality, serviceability, response time, etc.
- Context¹⁾²⁾³⁾ – where we have different levels of knowledge, i.e., the solvable contexts, and insolvability, where we need to come back to a solvable context, to solve the built-in root cause
- Uncertainty context – of knowledge of customer need, WHAT (e.g., include modularity as a requirement or not)
- Complex context – where we need more knowledge⁴⁾, planning is not possible, unplannable HOW
- Complicated context – where we need some more knowledge, planning is possible, plannable HOW
- Clear context – where we have the knowledge we need, planning is possible, plannable HOW
- Variability – of the unknown future, which needs to be considered with margins in the plannable HOW⁵⁾, due to; mistakes, carelessness⁶⁾, inaccuracies, Murphy's law, process variation, environmental factors, etc.
- Domain – a field of specific knowledge; trucks, retirement salary, telecom, bank, retail, etc.⁷⁾
- Indefiniteness – includes uncertainty, complexity, variability, but not insolvability

1) A context is domain independent, incl. independent of number of people 2) Think about context as "arv" with human natural science to follow, and domain more as "miljö", i.e., what we learn

3) Note! The focus is on context, especially the complex context. Think requirements, architecture, integration, test, etc.; domain independent 4) until we know if there is a solution or not

5) In the complicated or clear context 6) e.g., careless time planning that is not handling variability is a special one;-) 7) SW and HW are only sub-domains