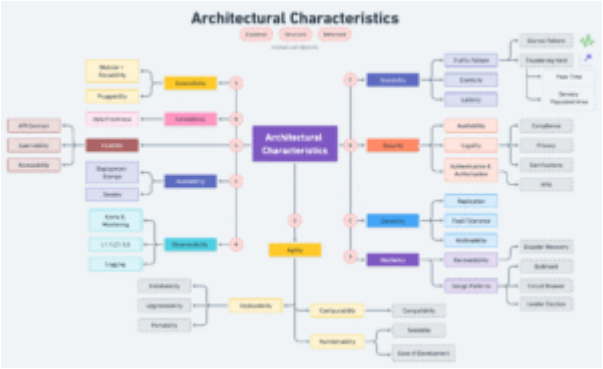
Aqui um artigo bem interessante da ByteByteGo sobre requisitos não funcionais:

https://medium.com/bytebytego-system-design-alliance/top-10-architecture-characteristics-non-functional-requirements-with-cheatsheat-7ad14bbb0a9b



Imediatamente me lembrei dos anos em consultoria e os diversos projetos de assessment da qualidade dos sistemas em vários bancos (na época parte da base teórica era a ISO 9126).

Hoje em dia, com a expansão (muitas vezes não suficientemente bem arquitetada) do uso do modelo ágil, tenho a percepção de que é bem maior o risco de se esquecer desses conceitos e acabar por ir se degradando a qualidade dos sistemas.

Especialmente se a maturidade do uso do ágil se limitar simplesmente ao “vamos colocar cada dia mais squads para rodar, é só colar uns post-its no quadro e não temos por que se preocupar além do escopo dessa sprint”!

Isso mostra a importância de se perseguir uma visão de médio e longo prazo sobre a saúde e a qualidade dos sistemas ao longo de todo o seu lifecycle.

Não dá para se limitar apenas no que será feito nas próximas duas semanas, tem que se ponderar o que se deseja para a arquitetura geral.

Caso contrário, tudo cobra o seu preço e quando menos se espera você tem nas mãos um legado impossível de manter e se vê ocupando boa parte do seu tempo (e da vida do seu time) apenas resolvendo incidentes!

Creio ser fundamental, quando se trata de sistemas para os quais se espera uma vida útil de vários anos, se preocupar em escalar o ágil de uma forma bem organizada e sem perder de vista a execução dos papéis que garantirão a visão geral da qualidade arquitetônica das aplicações.