Summary

Requirements engineering (RE) activities involve capturing both functional and non-functional requirements of the software system to be developed. This dissertation addresses a critical aspect of the RE process, namely, the engineering of Quality Requirements (QRs), also known as non-functional requirements, in agile. Agile development methods emphasize a Just-In-Time (JIT) approach to requirements, where functional requirements emerge incrementally throughout the development process. However, this approach presents challenges when applied to QRs.

Unlike functional requirements, which can evolve over time, QRs are long-term objectives that must often be identified early in the Software Development Life Cycle (SDLC). Early recognition of these requirements is essential for software architects, as it allows them to design an architecture that supports desired quality attributes, such as performance, scalability, security, and maintainability. While functional requirements can be addressed incrementally, QRs need to be integrated from the start to ensure the architecture meets these goals.

Agile development methods were originally designed for small, co-located teams as a response to the rigid, heavily documented approaches like the sequential waterfall model. The success of agile frameworks, such as Scrum, in these environments led large organisations to adopt agile methods, despite the differences between small, co-located teams and large, distributed ones. Implementing agile in large, distributed settings has not only stretched the benefits of agile but also amplified the challenges that were present in its original context—particularly the neglect of QRs. Several empirical studies have highlighted the failure to adequately address QRs in agile, a problem that is even more pronounced in large, distributed environments due to the increased complexity of requirements and risks.

Motivated by this issue, this paper-based dissertation proposes an approach to improve the engineering of QRs in Agile Large-Scale Distributed (ALSD) contexts. Through empirical research, we first examined the challenges faced by agile teams when engineering QRs in ALSD environments. This research identifies fifteen key challenges that impede effective QR engineering in ALSD projects. We then explored the underlying causes of these challenges and analysed the strategies agile teams use to mitigate their impact. Additionally, we assessed existing Agile Scaled Frameworks (ASFs) to determine whether their practices could address the identified QRs challenges. If so, we examined which practices they employ and which specific quality requirements challenges they aim to resolve. The study found that these ASFs partially address QRs but fall short in areas such as cross-team communication for QR management.

To bridge this gap, the dissertation proposes the Agile Quality Requirements Elaboration (AQRE) approach, designed to assist Agile teams in managing QRs effectively. AQRE is a Goal-Oriented Requirements Engineering (GORE) approach that helps structure QR discussions early in the project, ensuring that they are maintained throughout the Agile development process. The approach was validated by applying AQRE to a real-world agile project and collecting experts' opinions on the results of its application using Focus Group (FG) sessions with agile practitioners, who found it to be practical and potentially cost-effective.