1. **Introduction**

The Internet of Things (IoT) enables billions of interconnected devices to sense, communicate, and exchange data over a network [1]. These connected entities collaborate to trigger actions based on continuously gathered and analyzed information, facilitating informed decision-making [2]. IoT has emerged as a transformative technology, enhancing human life by enabling seamless interaction between objects, machines, and individuals. It allows real-world objects to evolve into intelligent, remotely controllable decision-making systems [3]. The IoT ecosystem comprises a diverse range of physical entities, including smartphones, household appliances, surveillance cameras, medical devices, vehicles, people, and buildings. These interconnected components share data based on specific communication protocols, with the primary goal of enhancing smart living, ensuring secure real-time monitoring, optimizing processes, and improving management systems [4, 5].

The concept of a System of Systems (SoS) [6] has gained prominence in distributed software systems. An SoS consists of multiple independent software systems that operate autonomously while interacting with one another. This interconnected nature results in a larger and more complex system. An SoS emerges through the interaction of distinct individual constituent systems (ICs) to deliver functionalities that none of the components could achieve independently. Unlike conventional large-scale systems, an SoS exhibits unique characteristics [7], all of which must be met for a system to qualify as an SoS. A smart city is a notable example of an IoT-based SoS [8], where IoT and SoS principles integrate to establish a strong technological foundation for smart urban environments. IoT-based SoS systems exhibit critical attributes such as operational and managerial independence, geographical distribution of components, evolutionary development, device heterogeneity, and emergent behavior.

These characteristics contribute to their inherent complexity, necessitating a well-structured design approach. However, a systematic methodology for designing IoT-based SoS remains largely unexplored. Existing studies [9-11] lack comprehensive guidance for modelling the software architecture of IoT-based SoS, offering only generic insights with minimal focus on software design. Moreover, most existing research separately addresses IoT and SoS architectures without integrating them. To develop an effective IoT-based SoS, it is crucial to establish a robust architecture that meets essential requirements such as distributed sensing, integrated management, and flexibility [12]. While some efforts have been made in this direction, they lack a specific focus on software design. Consequently, there is a pressing need for a systematic software modelling approach to simplify the development of IoT-based SoS and manage its complexity.

To bridge this gap, this article introduces a software modelling framework tailored for the development of IoT-based SoS. The research aims to address two fundamental questions:

How can software for IoT-based SoS be designed to be less complex and more reusable?

How can the complexity and reusability of the proposed approach be effectively validated?

Following the Unified Modelling Language (UML) and model-driven development (MDD) practices, the proposed framework consists of a meta-model and an associated UML profile. A UML profile is a domain-specific extension of UML that reuses existing UML elements by introducing customized stereotypes, tagged values, and constraints [13]. The meta-model and UML profile facilitate the creation of platform-independent software models for IoT-based SoS by incorporating key concepts from both IoT and SoS domains. Additionally, the framework includes a process model to guide systematic software design.

The proposed framework is based on microservices architecture, a widely adopted approach that enhances scalability, reusability, and interoperability—all critical requirements for IoT-based systems [6, 14]. To demonstrate its applicability, the proposed modelling approach is applied to a smart city case study, where design models are developed and evaluated using service-oriented software metrics. Comparative analysis shows that the proposed framework results in software models that are less complex and more reusable than existing approaches.

The remainder of this article is structured as follows: The next section provides the necessary background knowledge, followed by a review of related work on IoT-based SoS. Section 4 outlines the methodology used for developing the meta-model, while Section 5 details the proposed approach. Section 6 presents its application to a smart city case study. The evaluation of the proposed approach through service-oriented design metrics is discussed in Section 7, with a comparative analysis and results provided in Section 8. Finally, Section 9 concludes the article and outlines future research directions.

1. **Background**

The rapid evolution of digital technologies has led to the emergence of innovative architectures and paradigms that enhance the efficiency and scalability of modern systems. Microservices and the Internet of Things (IoT) are two such technologies that have significantly transformed software development and smart system implementations. Microservices facilitate the development of modular, loosely coupled applications, while IoT enables seamless connectivity among physical devices to create intelligent environments. Their integration has become crucial in domains such as smart cities, where real-time data collection and processing play a vital role in enhancing urban living.

**2.1 Microservices**

Microservices are lightweight, self-contained units of logic designed to support agile software development by enabling smaller and more frequent release cycles [15]. This architectural approach promotes the creation of highly modular and independent services, allowing for greater flexibility and scalability. In end-to-end applications, microservices typically communicate using HTTP REST, exchanging data in JSON format [16], which ensures independence from underlying architectures and programming languages. Furthermore, microservices are often coordinated through decentralized service choreography rather than centralized orchestration, fostering autonomy and reducing system complexity. Choreographed services interact in a loosely coupled manner, enhancing system cohesion while maintaining modularity.

**2.2 Traditional IoT Systems**

The Internet of Things (IoT) refers to a network of interconnected physical devices—ranging from smartphones to household appliances—that communicate with each other to exchange data and create an intelligent ecosystem. By leveraging IoT technologies, everyday objects can collect and transmit environmental data to remote servers, often in the cloud, enabling automated decision-making and enhancing overall efficiency. This connectivity paves the way for smart environments, allowing devices to operate seamlessly and improve daily life through automation and real-time monitoring.

**2.3 IoT for Smart Cities**

The concept of smart cities aims to leverage advanced technologies to enhance urban living by improving infrastructure, public safety, and overall efficiency. IoT plays a fundamental role in smart city initiatives by enabling real-time data collection through connected sensors and devices. These devices can be strategically deployed across various locations to monitor environmental conditions, manage traffic flow, enhance security, and respond to emergencies. By integrating IoT with software and communication networks, cities can automate processes and make data-driven decisions, ultimately improving residents' quality of life.