

Group Report Template

Stefan-Daniel Horvath*, Hi IM Micahel*, Student 3*, Student 4*, Student 5*,
University of Southern Denmark, SDU Software Engineering, Odense, Denmark
Email: * {Stefan-Daniel Horvath,student2,student3,student4,student5}@student.sdu.dk

Abstract—

Index Terms—Keyword1, Keyword2, Keyword3, Keyword4, Keyword5

I. Introduction and Motivation

The advancement of technology is at an all-time high and the possibilities using technology are expanding rapidly. Several industries are benefitting using assembly robots, self-driving forklifts, etc. People, machines, and products are directly related to each other. The term "Industry 4.0" describes the intelligent networking of machines and processes for industry with the help of information and communication technology [?]. The principles of Industry 4.0 result in several benefits, including the possibility of flexibility within production, ability to quickly change a production line for a different task, and many more [?].

This concept introduces the need for many different systems to be able to communicate in one way or another, which is one of the big challenges Industry 4.0 is faced with [?]. Therefore, for a system to successfully adhere to Industry 4.0 principles it must be capable of communicating between subsystems and various types of hardware within the system.

Industry 4.0 might originally have been coined for the manufacturing industry, but the potential of the concept is not limited to this industry alone. There is great potential for farming to adapt to the principles also. By incorporating the power of real-time data, IoT (internet of things) and analytics, farmers can respond to more nuanced changes in the environment and the health of the animals. This not only has great benefits for the productivity of the farm, but also has benefits for a more sustainable farming industry.

In this paper the intent of the project is to create a system that can handle various aspects of production, including monitoring biometrics, managing feed, and handling orders in a fully automated livestock farming system.

Therefore this paper designs and implements an architecture based on a smart livestock farming system. The system is evaluated by its ability to adhere to stated quality attributes. The motivation behind designing such a system is based on the need for interoperability between different systems, which in its current state in farming is one of the major challenges. The lack of communication between these technologies hampers their synergistic use

by farmers [?]. Through our work, we aim to address these challenges and contribute to the seamless integration of diverse technologies.

The structure of the paper is as follows. Section II outlines the research question and the research approach. Section III describes similar work in the field and how our contribution fits the field. Section IV-A presents a use case of the system, which gives us a better representation of what the system is able to do. The use case serves as input to specify QA requirements for the system IV-B. Section V introduces the proposed software architecture design for the system. Section VI evaluates the proposed architecture on tests conducted on the system and analyzes the results against the stated QA requirement.

II. Problem and Approach

Problem. In livestock farming, the production process is highly dependent on the environment and the health of the animals. To address this the system that has been developed in this project aims to look into how different architectural approaches can bridge different services and servers in a way that the production process can be reconfigured to adapt to the changes in the environment and the health of the animals. The problem of catching decease early and also trying to automating aspects of the farming production line could lead to more optimized environment both for the livestock but also for the farmer. It is believed that by developing such a interoperable system it would lead to better health for the animals and more time for the farmer to focus on other things where needed. To approach this problem the following research questions are formulated:

Research questions:

- 1) How can different architectures support the stated production system requirements?
- 2) Which architectural tradeoffs must be taken due to the technology choices?

Approach. The following steps are taken to answer this paper's research questions:

- 1) Develop the overall architecture of the system.
 - a) Identify the services and servers that are needed to support the production system.
 - b) Identify the usecases that the system should support.

- c) Identify the Quality Attributes that the system should support and how they are prioritized.
 - d) Identify the non-functional requirements that the system should support.
 - 2) Research the technologies that could be used to support the system and the tradeoffs that are made by using them.
 - 3) Develop a prototype of the system.
 - 4) Evaluate the prototype based on the Quality Attributes that are identified in step 1.
 - 5) Analyze the results and answer the research questions.
- A. Experiment design
 - B. Measurements
 - C. Pilot test
 - D. Analysis

VII. Future work

VIII. Conclusion

III. Related work

This Section addresses existing contributions by examining xxx in the I4.0 domain. In total, x papers are investigated.

In [?], experiences are elaborated on a three-layer architecture of a reconfigurable smart factory for drug packing in healthcare I4.0.

The paper [?] proposes an ontology agent-based architecture for inferring new configurations to adapt to changes in manufacturing requirements and/or environment.

In

In [?], [?] an architecture for a reconfigurable production system is specified. Two objectives for reconfiguration and how they can be reached are described.

Several papers [?], [?], [?] describe reconfigurable manufacturing systems that are cost-effective and responsive to market changes.

All contributions provide valuable knowledge about reconfiguration but lack a study of the software architecture perspective that specifies a quantifiable reconfigurability architectural requirement, a software architecture that adopts the architectural requirements, and evaluates the architectural requirement.

IV. Use Case and Quality Attribute Scenario

This Section introduces the use case and the specified x QASes. The QASes are developed based on the use case.

A. Use case

B. Quality attribute scenarios

V. The solution

This section will describe a proposed design of that aims to achieve the stated QASes stated in the previous section.

VI. Evaluation

This Section describes the evaluation of the proposed design. Section VI-A introduces the design of the experiment to evaluate the system. Section VI-B identifies the measurements in the system for the experiment. Section VI-C describes the pilot test used to compute the number of replication in the actual evaluation. Section VI-D presents the analysis of the results from the experiment.