



Contents lists available at ScienceDirect

Journal of Open Innovation: Technology, Market, and Complexity

journal homepage: www.sciencedirect.com/journal/journal-of-open-innovation-technology-market-and-complexity

BHiveSense: An integrated information system architecture for sustainable remote monitoring and management of apiaries based on IoT and microservices

Duarte Cota^{a,b}, José Martins^{c,d,e,*}, Henrique Mamede^{a,d}, Frederico Branco^{b,d}^a CEG-UAb, Universidade Aberta, Rua da Escola Politécnica, 147, 1269-001 Lisboa, Portugal^b Universidade de Trás-os-Montes e Alto Douro, Vila Real, Portugal^c AquaValor – Centro de Valorização e Transferência de Tecnologia da Água, Rua Dr. Júlio Martins, N°1, 5400-342 Chaves, Portugal^d INESC TEC – Institute for Systems and Computer Engineering, Technology and Science, 4200-465 Porto, Portugal^e Instituto Politécnico de Bragança, Campus de Santa Apolónia, 5300-253 Bragança, Portugal

ARTICLE INFO

Keywords:

Precision beekeeping
Sustainable remote monitoring
Low-cost
Internet-of-Things
Information systems architecture
Design science research

ABSTRACT

Precision Beekeeping, a field of Precision Agriculture, is an apiary management strategy based on monitoring honeybee colonies to promote more sustainable resource usage and maximise productivity. The approach related to Precision Beekeeping is based on methodologies to mitigate the stress associated with human intervention in the colonies and the waste of resources. These goals are achieved by supporting the intervention and managing the beekeeper's timely and appropriate action at the colony's level. In recent years, the growth of IoT (Internet-of-Things) in Precision Agriculture has spurred several proposals to contribute to the paradigm of Precision Beekeeping, built on different technical concepts and with different production costs. This work proposes and describes an information systems architecture concept named BHiveSense, based on IoT and microservices, and different artefacts to demonstrate its concept: (1) a low-cost COTS (Commercial Off-The-Shelf) hive sensing prototype, (2) a REST backend API, (3) a Web application, and (4) a Mobile application. This project delivers a solution for a more integrated and sustainable beekeeping activity. Our approach stresses that by adopting microservices and a REST architecture, it is possible to deal with long-standing problems concerning interoperability, scalability, agility, and maintenance issues, delivering an efficient beehive monitoring system.

1. Introduction

The current digital transition is a remarkable and transformative phenomenon reshaping societies and revolutionising businesses worldwide. With rapid advancements in technology and connectivity, the world is witnessing a remarkable shift towards a digital-centric era. This transition is marked by the widespread adoption of digital technologies and the increasing reliance on digital platforms and services for various aspects of life (Guandalini, 2022).

Agriculture, a vital economic activity for human survival as a species, has evolved throughout time, adapting - through the constant incorporation of innovation, to new challenges and realities where environmental balance and sustainability are keys (de Boon et al., 2022; Wu and Li, 2023). According to ISPA - *International Society of Precision Agriculture*, "Precision Agriculture is a management strategy that gathers,

processes and analyses temporal, spatial and individual data and combines it with other information to support management decisions according to estimated variability for improved resource use efficiency, productivity, quality, profitability and sustainability of agricultural production" (ISPA, 2021).

Precision Beekeeping is a branch of Precision Agriculture which extends these concepts to the specific area of apiculture in a systematic approach of beekeeping practices supported by the consistent integration of all the information necessary for beekeeper's decision-making, efficient use of resources and sustainably maximising production (Hadjur et al., 2022).

The world has been slowly but relentlessly witnessing the degradation of honeybee colonies. The role of these insects in the balance of ecosystems, environmental health and preservation of life is crucial. Further than the production of honey, beeswax, royal jelly and propolis,

* Corresponding author at: AquaValor – Centro de Valorização e Transferência de Tecnologia da Água, Rua Dr. Júlio Martins, N°1, 5400-342 Chaves, Portugal.
E-mail address: jose.martins@aquavalor.pt (J. Martins).

¹ Present address: Rua Dr. Júlio Martins, N°1, 5400-342 Chaves, Portugal

<https://doi.org/10.1016/j.joitmc.2023.100110>

Received 22 June 2023; Received in revised form 31 July 2023; Accepted 1 August 2023

Available online 4 August 2023

2199-8531/© 2023 The Authors. Published by Elsevier Ltd on behalf of Prof JinHyio Joseph Yun. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

bees' pollination activity is essential to the proliferation of a wide variety of plant species (Kasiotis et al., 2021; Ullah et al., 2021).

This phenomenon of decreased activity in apiaries, known as CCD (Colony Collapse Disorder), is characterised by a drastic and sudden reduction in the bee population. The causes of this problem are not yet fully identified, but scientists tend to converge towards multiple factors that can act alone or in combination, contributing to the decline of bee populations (e.g., mites and viruses, malnutrition, pesticides, beekeeping practices, genetically modified crops, etc.) (Cecchi et al., 2020; Hong et al., 2020).

The introduction of systems that allow monitoring of apiary production and environmental conditions, inside and outside the hives, has been proposed through several solutions aiming to mitigate this phenomenon. These devices generate data to understand better which factors most affect productivity, honeybee activity, and hives' health. Besides data collecting and scientific analyses to better understand and mitigate CCD, these systems are also focused on directly supporting beekeepers in their activity by signalling situations that require human intervention in apiaries, either by triggering alarms and sending notifications or signalling events on monitoring panels. These features lead to fast and focused intervention while diminishing negative impacts in apiaries, fitting these systems within the paradigm of Precision Agriculture and Precision Beekeeping.

The expansion of IoT systems in recent years has been paralleled by generalised access to low-cost COTS (Commercial Off-The-Shelf) components. Within these conditions, some remote monitoring systems have been proposed to reduce apiary maintenance costs and increase colony production and data collecting for scientific and investigation purposes. These contribute to CCD mitigation and a better understanding of the factors affecting beekeeping activity.

Our proposed solution aligns to contribute to more sustainable apiary management, both in terms of productivity maximisation, colony health and well-being, by conceptualising an architecture for precision beekeeping based on microservices, focused on flexibility, modularity, scalability, replicability and adaptability, granting interoperability, a significant low-cost production, and low-complexity maintenance.

To demonstrate and evaluate the adequacy of the proposed conceptual artefact, a remote sensing hardware prototype, a Web App, and a Mobile App, both served by a REST-based (Representational State Transfer) backend API were developed, and these will be described later in this article.

Hence, in summary, our research highlights are as follows:

- There is a clear need to establish innovative, scalable and agile beehive and apiary monitoring solutions based on IoT to not only provide beekeepers with an additional layer of support for his decision-making process but also to ensure he is alerted when the beehives need *in-loco* attention;
- Monitoring solutions based on microservices architectures tend to be more agile, scalable, maintainable and performant when compared to solutions based on other types of architectures;
- A novel IoT-based beehive monitoring solution supported by a microservices architecture that can easily encompass additional sensing nodes and functional modules is proposed;
- The proposed monitoring solution has been tested using a controlled case study, and it was possible to establish that it not only delivered valid and accurate results, but also presented adequate performance and stability.

This paper is organised as follows: in Section 2, we present the related work, while in Section 3, the selected methodology and the process model approach are described in detail. Section 4 describes the conceptual architecture, and the validation artefacts are presented and evaluated. In Section 5 and 6, we offer the conclusions and suggest topics for future work.

2. Related work

2.1. Remote monitoring

One of the main challenges associated with Agriculture 4.0 is the development of remote monitoring systems and solutions that can aid farmers in improving the control they have on their farms and continuously fostering production efficiency and sustainability (Dayioglu and Turker, 2021). As established by Abbasi et al. (2022), when applied to the agriculture sector, the remote sensing field of the study refers to the development and deployment of sensor networks – composed of multi-parameter sensing nodes – that can ensure connectivity between all the elements of the network and a central connectivity node, system, service or API. Furthermore, drawing on Khujamatov and Toshtemirov (2020), despite the referred sensing network's potential impact towards the development of the agri-food sector, they are somewhat dependent on the degree of complexity and ability to accurately monitor - and report the collected data - the set of predefined parameters of each of the sensing nodes.

Remote monitoring systems for hives integrated into apiaries is an area of research where a relevant effort has been made and with a significant set of literature, mainly focused on hardware, its complexity, and how data generated by these devices contribute to Precision Beekeeping. This branch of Precision Agriculture is defined as a strategy for apiary management based on monitoring individual colonies (hives) with the goal of not only minimising resource consumption and maximising production sustainably but also optimising beekeeping activity and CCD mitigation (Zacepins et al., 2015).

Precision beekeeping's focus is to supply beekeepers with access to intelligent and automated tools that reduce their workload, maintain long colony life cycles, and increase productivity. Alongside this, it is also important to establish this technological scope and focus on protecting the colonies and their continuous sustainability (Uthoff et al., 2023).

Also, the use of remote monitoring systems and solutions will aid beekeepers as – due to the wireless technologies inherent to the sensor networks and nodes, data on apiaries can be accessed at any moment, in near real-time, on any devices (e.g., mobile devices), thus avoiding unnecessary travel and expenses (Hong et al., 2020). In summary, remote sensing solutions are one of the most promising technological solutions for precision beekeeping (J. Navarro et al., 2022).

2.2. IoT-based remote monitoring

Drawing on authors such as Symeonaki et al. (2022), IoT is increasingly assuming a central role as a trigger for the development of innovative and disruptive (remote) monitoring systems based on sensing nodes that are continuously interconnected, that are both scalable and agile in the incorporating of updates when facing new needs, and that are globally interoperable. Furthermore, existing literature also holds multiple examples of IoT-based applications, frameworks, ontologies and methodologies (Chukkapalli et al., 2020; Farooq et al., 2020; Symeonaki et al., 2022).

IoT-based remote monitoring in Agriculture 4.0 refers to using Internet of Things (IoT) technology to monitor and manage agricultural operations remotely (Abbasi et al., 2022). It involves the integration of sensors, devices, and connectivity solutions to collect and transmit data from various agricultural assets, such as crops, livestock, soil, and equipment. From a more practical perspective, when applied to the Agriculture 4.0 paradigm, IoT-based remote monitoring is perceived as an enabler of real-time data acquisition and analysis, offering farmers, livestock breeders, poultry farmers, pig farmers, goat farmers, beekeepers, and many other industry players valuable insights for decision-making, optimising resource usage, and improving overall efficiency. The key components of this system include IoT sensors, network connectivity, data processing, and visualisation tools (Araújo

et al., 2021; Maffezzoli et al., 2022).

IoT sensors are deployed in agricultural environments to capture and collect data (Roy and De, 2022). These sensors can measure various parameters, including temperature, humidity, soil moisture, pH levels, nutrient content, rainfall, air quality, and livestock behaviour. They can be placed in fields, greenhouses, animal housing, and machinery to monitor critical factors affecting agricultural productivity continuously (Swain et al., 2021). The collected data from IoT sensors is transmitted via wireless networks, such as Wi-Fi, cellular networks, or Low-Power Wide-Area Networks (LPWAN), to a centralised platform or cloud infrastructure for storage and analysis (Miranda et al., 2019).

The data obtained from IoT-based remote monitoring systems can be analysed using advanced analytics techniques and algorithms. Machine learning and data modelling approaches can be employed to detect patterns, identify anomalies, and derive actionable insights (Pyngkodi et al., 2022). For example, predictive analytics can help anticipate weather conditions, disease outbreaks, or crop growth stages, facilitating proactive interventions and resource planning (Xu et al., 2022). Visualisation tools, such as dashboards and data visualisations, enable users to understand the collected data intuitively. These tools can present real-time information, historical trends, and alerts in a user-friendly format, allowing farmers to monitor the status of their agricultural operations at a glance and respond promptly to emerging challenges or opportunities. Thus, IoT-based monitoring can trigger the arise of decision support systems that, drawing on real-time data collection, analysis, and visualisation, can aid farmers in their daily decision-making tasks and their strategic planning (Zhai et al., 2020).

The benefits of IoT-based remote monitoring in Agriculture 4.0 include improved resource management, enhanced productivity, reduced costs, and increased sustainability. Farmers can optimise irrigation, fertilisation, and pest control practices by continuously monitoring key parameters, leading to better resource allocation and minimising environmental impact. Additionally, early detection of anomalies or deviations in crop health or livestock behaviour can help prevent losses and improve overall yield and animal welfare (Maffezzoli et al., 2022).

Hence, drawing on the arguments above, IoT-based remote monitoring in Agriculture 4.0 leverages sensor technology, connectivity, data analysis, and visualisation to enable real-time monitoring and decision-making in agricultural operations. This approach empowers farmers to manage their assets more efficiently, respond to changing conditions promptly, and adopt precision agriculture practices for sustainable and productive farming.

2.3. Beehive monitoring using IoT-based solution

Bees are vital in pollination, supporting global food production and maintaining ecosystem biodiversity. However, in recent years, bee populations have faced significant challenges due to climate change, habitat loss, and pests (Panziera et al., 2022; Patel et al., 2021). To address these concerns and ensure the well-being of bee colonies, beekeepers and researchers are turning to IoT-based solutions for beehive monitoring (Hong et al., 2020).

To this point, multiple authors aimed to deliver IoT-based solutions to monitor beehives and, consequently, foster bee sustainability. As easily perceived by analysing existing literature, the majority of proposed solutions have a similar backbone structure that is composed of multiple sensors connected to a microcontroller and to a communication hub, all of these powered by a battery system that serves the purpose of relaying the collected data with the remote servers (Andrijević et al., 2022; Aydin and Nafiz Aydin, 2022; Cecchi et al., 2020; Hong et al., 2020; Imoize et al., 2020; Kontogiannis, 2019; Tashakkori et al., 2021; Zacepins et al., 2020). Hence, we can extrapolate that, from a conceptual perspective, the main modules of a beehive monitoring solution are as follows:

- A MCU - Microcontroller Unit (also known as the brain of the entire solution) that collects data from the sensors and communicates these readings to a remote server at regular intervals;
- A set of sensors physically connected to the microcontroller or communicating with it via wireless protocols;
- A source of electrical energy consisting of a battery connected or not to a photovoltaic solar panel through a charge regulator (i.e., a solution mainly adopted for remote installations).

The sensing module, i.e., the set of sensors used to collect data, is typically located inside the hive, although there are cases of sensors placed on the outside of that hive or in its proximity. Concerning the MCU, these are generally placed outside, on the roof of the beehive (Khairul Anuar et al., 2019), in COTS dedicated boxes or 3D printed solutions similar to the one proposed by Tashakkori et al. (2021). The isolation of the MCU allows protection against the defensive and natural behaviour of the honeybees. The design of a remotely monitored hive must also be carefully planned, trying to anticipate the obstacles that may arise in its development and assembly (Ntawuzumunsi et al., 2021).

Based on Bellino et al. (2022) and Voudiotis et al. (2021) arguments, the main module of an apiary remote monitoring system is the microcontroller unit. This element is responsible for iteratively executing a firmware that updates all sensor readings, prepares the data in a specific data format, and sends it to a Web server. Several COTS microcontroller units have been tested and integrated into remote monitoring systems. Despite the relevant amount of cases using the "NODE ESP32" model as the MCU (Andrijević et al., 2022), "NODE ESP8266" model is more widely used (Imoize et al., 2020; Jegan et al., 2021; Zacepins et al., 2020) as the results of its higher performance, easier connections to external components and easier programming. This last MCU also has an onboard Wi-Fi module – an added value to those developing the monitoring solutions – and is considered low-cost. In some more recent research, "Arduino" family units are also used. Examples of this are both the Catania and Vallone (2020) and Ntawuzumunsi et al. (2021) works, where the authors used an "Arduino ATmega2560", and also Kontogiannis (2019) used an "Arduino ATmega328" as the MCU of the proposed monitoring solution. Other platforms, such as "Raspberry Pi", have also been used by researchers when offering their remote monitoring solutions (Cecchi et al., 2020; Tashakkori et al., 2021).

2.4. Beehive monitoring parameters

In recent years, bee populations have faced significant declines due to various environmental factors and the threat of colony collapse disorder. To address this issue and support beekeepers in their efforts to protect these essential pollinators, beehive monitoring has emerged as a promising solution (Anwar et al., 2023).

Beehive monitoring involves using advanced technology to gather real-time data and insights from beehives. By deploying various sensors within the hive, beekeepers can monitor critical parameters such as temperature, humidity, weight, and sound. These sensors provide valuable information on the overall health and well-being of the colony, allowing beekeepers to make informed decisions and intervene promptly if necessary (Abdollahi et al., 2022; Bellino et al., 2022).

One of the primary benefits of beehive monitoring is the early detection of potential issues. For example, sudden changes in temperature or humidity levels may indicate that the hive is under stress, and swift action can be taken to identify and address the underlying problem, such as disease or pests (Zaman and Dorin, 2023). Tracking the hive's weight can offer insights into honey production and help beekeepers determine the optimal time for honey harvesting, ensuring maximum yield while leaving enough honey for the bees' sustenance (Robustillo et al., 2022).

As established by authors such as Ntawuzumunsi et al. (2021) and Szczurek et al. (2023), ensuring efficient and effective multiparameter monitoring of a given beehive is essential to assess its overall health and

production (perspectives). The related literature also unveils a set of indicators that are commonly used for both beehives and apiaries monitoring systems that serve the purpose of delivering insights on the occurrence of essential changes that can potentially indicate the need for human intervention to preserve the colony:

- **Weight:** monitoring the weight allows beekeepers to have necessary inputs concerning the state of a hive (Meikle et al., 2016). Providing this data in almost real-time lets for the arise of valid perceptions of the state of the hive. The decrease in the hive's weight tends to represent the occurrence of events for which it is critical to unravel the origin and totality of the impact generated. Examples of these events are the increase or decrease of the colony population size, the existence of adverse environmental conditions that might have triggered the need to supply food, the occurrence of swarming, or the honey harvesting itself (Cecchi et al., 2020; Fitzgerald et al., 2015; Hong et al., 2020). Thus, monitoring weight allows beekeepers to have precise data to analyse the condition of the hives and indicates the need for harvesting or human intervention, all without disturbing the colony.
- **Temperature:** honeybee reproduction demands that the air temperature inside the hive remains stable and floats, at the very worst, between + 15 and + 35°C. To ensure the referred temperature stability, bees generate metabolic heat by grouping themselves in a cluster in case of lower temperature values or taking water to the hive's interior in case of higher temperature values, thus increasing humidity. Temperature stability is an important indicator of the colony's health status (Edwards-Murphy et al., 2016). According to Zacepins et al. (2016), the increase in temperature is also an important indicator of swarming, a natural phenomenon which, if not controlled by beekeepers, can result in crucial negative profitability impacts.
- **Humidity:** monitoring the relative humidity inside the hives is a relevant variable for determining the state of the colonies and has a direct impact on productivity (Mohamed and Mansor, 2023). As established by existing literature, temperature and humidity vary according to the bee's subspecies (Gil-Lebrero et al., 2020). For instance, according to Hong et al. (2020), the relative humidity inside the hive must be between 90% and 95% to enhance the hatching of the eggs. On the other hand, authors such as Andrijević et al. (2022) pose that when maintaining a temperature of 35°C and a humidity percentage of around 75%, beehives have better chances to ensure sustainable growth throughout. Hence, drawing on the above, it is very relevant to provide long-term monitoring of relative humidity inside the hive as it also contributes, alongside temperature, to prompt the beekeeper for immediate intervention.
- **Sound:** The sound produced by bees can indicate the hive's overall health and vitality. A strong and healthy hive will produce distinctive sounds, typically associated with the worker bees' labour. In contrast, weak or stressed colonies might exhibit different sound patterns, reflecting potential issues that require attention, such as diseases, foraging activities, swarming or other colony events (Abdollahi et al., 2022). Hence, monitoring the sound of a hive will thus allow the collection of data that the beekeeper can use to, in parallel with the definition of alarm mechanisms, be able to react in anticipation of the occurrence of events or situations that put the health and sustainability of the hive at risk (Kulyukin et al., 2018; Murphy et al., 2015).
- **Population Size:** Several works also refer to the population size as an indicator of the hive health and production level. Honeybee colonies can be likened to intricate systems, where their survival hinges on individual bees' quality, ability to adapt, and resilience to various pressures. Multiple stress factors interacting with one another are likely responsible for the mortality of colonies (Requier, 2019). Thus by ensuring efficient monitoring of the population size, beekeepers can actively anticipate serious issues that might endanger the

beehive. The existing literature poses multiple proposals to count/estimate the number of bees that exist in a given beehive at a given time from an array of photoelectric or infrared sensors on both sides of the individual holes in the entrance of the hive that, by detect the sequence of the interruption of the radiation beams, can give information about the movement of the bees (i.e., leaving or entering the hive) that is finally used to estimate the population size (Struye et al., 1994), to systems that use RFID tags attached to the back of the bees to allow detection at the entrance of the hive (Schneider et al., 2012), and to real-time imaging systems that by a series of analysis to a set of collected images can detect each individual inside the hive (Ngo et al., 2019).

2.5. Power systems

Despite representing a groundbreaking paradigm and playing a decisive role in remote monitoring and operational control for a multitude of contexts, as established by Ramson et al. (2020), one of its most critical aspects is the continuous assurance of power supply. If this is true for all situations where IoT-based remote monitoring is used, when focusing on this issue from the agriculture sector perspective this is even critical as the referred solutions must be deployed in remote areas where most of the times there isn't a technical supply grid that supplies power, energy or both (Liu et al., 2022).

According to Sari et al. (2020), the choice for the optimal location for deployment of an apiary is made by considering a set of parameters that are intrinsically related to the beehive sustainability and production maximisation and not on the availability of internet connection or even local energy supply. Hence, and drawing on authors such as Zacepins et al. (2015), we can easily establish that these are not the ideal condition to implement a precision beekeeping approach or solution. Remote apiary monitoring systems demand the existence of a stable and permanent electric power source, allowing a long-term and low-maintenance system operation (Fitzgerald et al., 2015). The typical location of apiaries, away from public power grids, represents a contingency that can be overridden by using autonomous power systems (Gil-Lebrero et al., 2016).

In order to address the abovementioned issues, authors such as Andrijević et al. (2022), Fitzgerald et al. (2015), He et al. (2020), Kontogiannis (2019), and Murphy et al. (2015), have proposed a self-sustained power system using solar photovoltaic cells and batteries to compensate the intermittency of the solar energy availability, associated with low-power components.

2.6. Proposed architectures

Proposing architectures that support scalable Internet of Things (IoT) monitoring solutions is of paramount importance in today's rapidly advancing technological landscape. As the number of connected devices and sensors continues to grow exponentially, scalable architectures are crucial for efficiently managing and processing the massive volume of data generated by these interconnected devices (Calvo et al., 2022). Scalability ensures that IoT monitoring systems can accommodate the increasing number of devices and users without compromising performance or reliability (Sharma et al., 2017).

According to existing literature, using microservices is a key criterion for adopting a contemporary software architecture approach, where features such as scalability and flexibility, decentralised governance and data management and use, reusability, resiliency, technology stack, replaceability, and maintainability (Atitallah et al., 2022; Butzin et al., 2016; Cebeci and Korçak, 2020; Lyu et al., 2020; Siddiqui et al., 2023).

Several benefits are associated with the usage of microservices. Having a single responsibility, each microservice provides modularity for quick and easy deployment (Gan and Delimitrou, 2018) and enables agility and fast development (Marquez et al., 2021). A microservices architecture also delivers technology heterogeneity. Not only in

software (coding language, frameworks, or database management systems) but also in hardware (Gan and Delimitrou, 2018).

For communications between the sensing devices and the data and storage subsystems, the IoT-based remote sensing concept typically uses the Transmission Control Protocol/Internet Protocol (TCP/IP) technology. In these conditions, having an access point is necessary and depends on the availability of: (i) a local Wi-Fi network using cable connections (ethernet); (ii) a local Wi-Fi connection based on mobile data networks. In the related work applied to beehive monitoring systems, Wi-Fi modules are the most frequent devices for data transfer (Andrijević et al., 2022; Ntawuzumunsi et al., 2021; Tashakkori et al., 2021). In several studies, it is also possible to determine the options for data storage. Several authors use MySQL (Hong et al., 2020; Zabasta et al., 2019) while others use NoSQL databases – MongoDB – (Komasilovs et al., 2019).

From the analysis of previous research it is possible to identify authors who proposed concepts of simple, yet disruptive, solutions for beehive monitoring that are based on the abovementioned micro-services architecture (Alifieris et al., 2023). Despite this, and as stated by the referred authors, at this point in time there is still a gap in what refers to the practical development and deployment of those conceptual proposal.

3. Methodology

3.1. Design science research

We follow a Design Science Research (DSR) approach in this project. DSR is a proven methodology oriented towards problem-solving. DSR aims to improve human knowledge by creating innovative artefacts through disruptive solutions for real-world problems (Hevner et al., 2004). According to these authors, it is possible to identify a set of principles to apply DSR in information systems design, to produce artefacts whose utility, quality and efficiency must be rigorously verified, and the solution efficiently disclosed to the relevant stakeholders. This project is based on the Design Science Research Model (DSRM) proposed by Peffers et al. (2007) for information systems research. This model encloses a set of six nominal sequential activities (Fig. 1). However, the authors say that not always researchers would proceed in this sequential order. In practice, it is possible to start at almost any point and move in both directions.

The six sequential activities proposed by Peffers et al. (2007) are the following:

Activity 1: Identify problem and motivation – In this activity, the specific research problem should be defined to atomise the problem conceptually so that the solution can capture its complexity. By justifying the value of a solution, researchers and the general stakeholders will be motivated to research, find the solution and accept the results. It also helps to understand the rationale linked to the researcher's understanding of the problem. For this activity, it is important to have knowledge of the state of the problem and the impacts of its solution.

Activity 2: Define the objectives for a solution – conclude about the objectives of a solution starting from the problem definition and awareness of what is possible and practical. The objectives can be quantitative or qualitative and should be derived from rationality from the problem specifications. For this activity, knowledge about the state of problems and current solutions, if any, and their efficacy is required.

Activity 3: Design and development – at this point, it is expected to create the artefact. Such prototypes are potential models, methods, or instantiations. Artefacts can also be new properties of technical, social, and/or informational resources. Conceptually, an artefact delivered in the context of a design research methodology can be any designed object in which a research contribution is embedded in the design. This activity includes also setting the artefact's desired functionality, its architecture and then developing the actual artefact itself. Resources required for going from objectives to design and development include knowledge of theory that can be brought to deliver a solution.

Activity 4: Demonstration – this activity aims to demonstrate the use of the artefact to solve one or more instances of the identified problem. It can involve the use of the artefact in experimentation, simulation, case study, proof, or other activities. To demonstrate the artefact, an effective knowledge of how to use the artefact to solve the problem is needed.

Activity 5: Evaluation – aims to observe and measure how the artefact supports the solution to the problem. The evaluation aims to compare the results from the demonstration of the artefact to those defined in the proposed solution. This activity demands mastery of metric and analytical tools and, depending on the nature of the artefact, can take many forms. Evaluation could include items such as a comparison of the artefact's functionality with the solution goals from Activity 2, objective quantitative performance measures such as budgets or items produced, the results of satisfaction surveys, client feedback, or simulations. It could include quantifiable measures of system performance, such as response time or availability. Conceptually, could include any relevant empirical evidence or logical proof. At the end of this activity, the researchers can decide whether to iterate back to Activity 3 to try to improve the effectiveness of the artefact or to continue

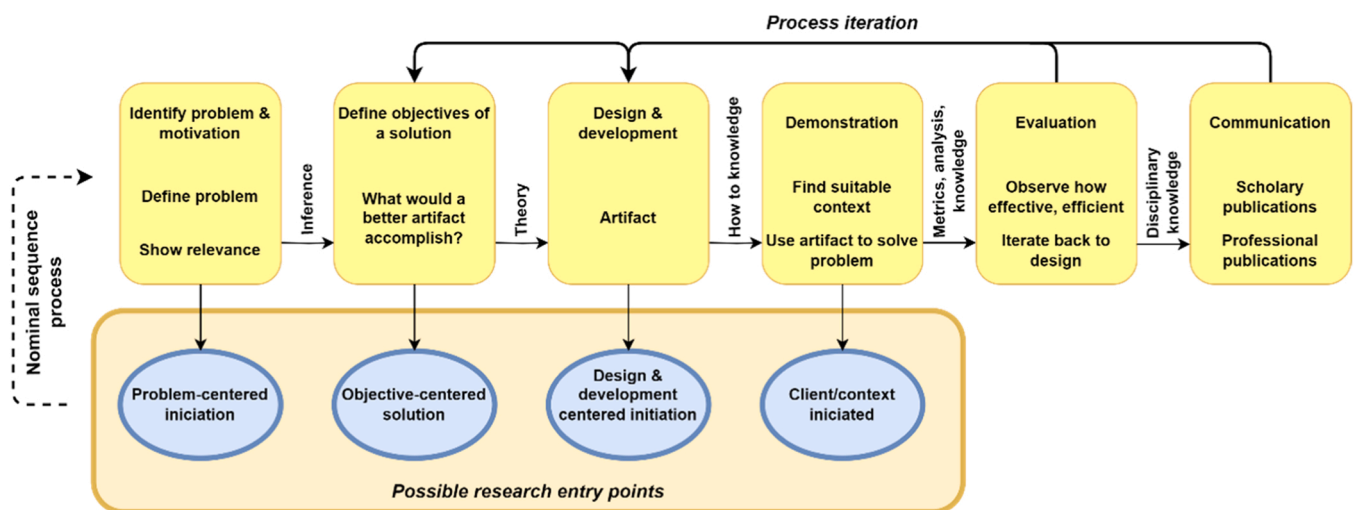


Fig. 1. DSRM process model adapted from Peffers et al. (2007).

to communicate and leave future improvements to subsequent projects.

Activity 6: Communication – in this activity the problem, its relevance, the artefact, the degree of innovation and functionality, the rigour of the design and its efficacy must be communicated to other researchers and other audiences. Researchers may use the process model to structure their papers.

Considering the context of this project in the field of Precision Beekeeping, its goals and its focus on the process model proposed by Peffers et al. (2007), we have structured the work along a six-step iterative process detailed in Fig. 2. Each step is directly related to the project development.

4. Proposed architecture

In this chapter, the proposed architecture is described in detail. First, we identified, contextualised, and defined the problem, based on our goals and the related work. Then, we propose a solution based on the previously identified problem and the goal of implementing the necessary features. After this step, we designed and developed the artefact corresponding to the proposed architecture, and created four pilot artefacts for demonstration proposes. In the evaluation step, several tests were performed using these pilot artefacts. Finally, in the communication step, we present the knowledge disclosure, results, and future strategies.

4.1. Problem identification and motivation

The problem identification and motivation start by analysing the main research question: what are the functional, technological, and architectural visions necessary for the deployment of a low-cost, COTS-based monitoring system, for apiary remote monitoring based on IoT? The answer to this question can be divided into four sub-questions: (1) What is the technical and functional framework for an apiary monitoring system?; (2) Which low-cost components should be selected and applied to produce an artefact allowing remote self-sustained apiary monitoring?; (3) Which technological architecture will best support the execution of an IoT-based system to apiary remote monitoring?; and, (4) What contextual limitations will have a bigger impact in the success of an apiary remote monitoring system?

Despite the significant amount of work already proposed in this area

of Precision Beekeeping, the literature review helped to identify a lack of focus on the global information system architectural concept for the delivered systems. It is clear that previous studies are well-oriented to describe prototypes that supply huge amounts of data to validate hypotheses concerning, e.g., the identification of the environmental variables that can provide important information about the state of the honeybee's colonies and their health, relating these variables with hive production and bee colony evolution. However, proposing a high-level conceptual architecture, and solid demonstration artefacts, based on more recent architectural and technological paradigms can contribute significantly to the field of Precision Beekeeping, also in the scope of the emerging paradigm of Artificial Intelligence, more specifically the sub-fields of Machine Learning and Deep Learning, well known to be intensively data dependent.

The problem characterisation was an iterative process which considered the extant literature and the current state-of-the-art of remote monitoring systems for Precision Beekeeping. This characterisation identified the main problem that motivates this project: the lack of an integrated solution describing an information system architecture for remote apiary monitoring, in terms of hardware, communications, data processing and storage, end-user application features, and data specifications.

4.2. Artefact features definition

Considering the problem definition and the related context (Precision Beekeeping), the main artefact features were listed. As stated in the literature review, the main goal of Precision Beekeeping is to allow having almost real-time data collected from the bee colonies (hives). This data can provide the necessary information to assess the state of the colony in terms of health, the presence of predators or growing diseases, and natural events like swarming.

For this project, to generate real and contextualised data, the choice was to monitor temperature, humidity (inside and outside the hive), the weight of the hive, and sound inside the hive, since these are the variables most mentioned as the most important to assess the state of the honeybee colonies. An additional mechanical sensor to detect if the hive lid is opened, by an animal attack, for example, was also included. By monitoring these variables, several important events can be detected generating alarms and notifications to the beekeeper.

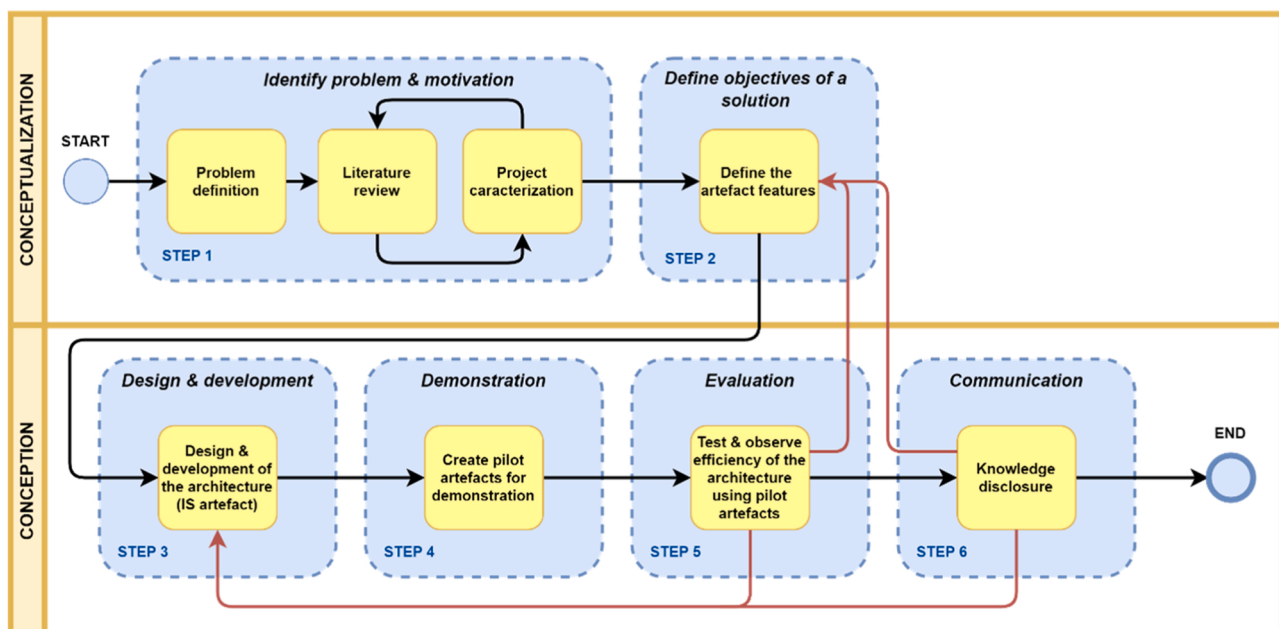


Fig. 2. Schematic representation of the project process model.

After being collected and processed by a physical device (sensing device), the data must be sent to a data processing unit (Web server) using a specific data format. The data is then processed before being stored for future analytics. During this process, several instant analyses can be performed to assess the actual state of the hive. This intermediate step will trigger and register events that will automatically generate notifications to the beekeeper using different approaches (e.g., push notifications, update monitoring dashboards, or sending e-mails). These notification events are triggered based on high levels of temperature or humidity inside the hive, swarming based on the sound level inside the hive and the evolution of the weight, and simple harvesting information by using a setpoint defined by the beekeeper.

Also, in the scope of this work, management of the apiaries is an important feature. So, the system must also manage the scheduling of interventions (e.g., regular maintenance or cleaning), associated with apiaries, notifying the beekeeper when those calendar interventions are approaching, in a single and integrated interface.

Several other features were also considered, like organisational aspects of the apiaries, registration of devices (hives), commissioning and decommissioning of devices in apiaries, analytical aspects like monitoring honey production (hive and apiary level), and event logs, among

others.

4.3. Design and development of the architecture artefact

The IoT can be defined as a worldwide network of unique inter-connected devices, with sensing abilities, using communication protocols, leveraging computational capacities, and delivering services/capacity to provide data analytics (Gupta and Quamara, 2020).

The most common architecture in IoT is the three-layer and four-layer (Ramachandran et al., 2022). The three-layer architecture, the most common, includes (i) a physical or perception layer, consisting of sensors and actuators that capture data or control acting devices, (ii) a network layer responsible for connecting the physical layer to, for example, servers, and (iii) an application layer that delivers applications to the users.

The four-layer architecture is based on the three-layer approach, adding a new layer called the service layer. This layer acts as a middleware between the application layer and the network layer, managing access to different services and APIs.

In this project, a proposal for a beehive remote monitoring system has been developed, based on a multilayer microservice-based

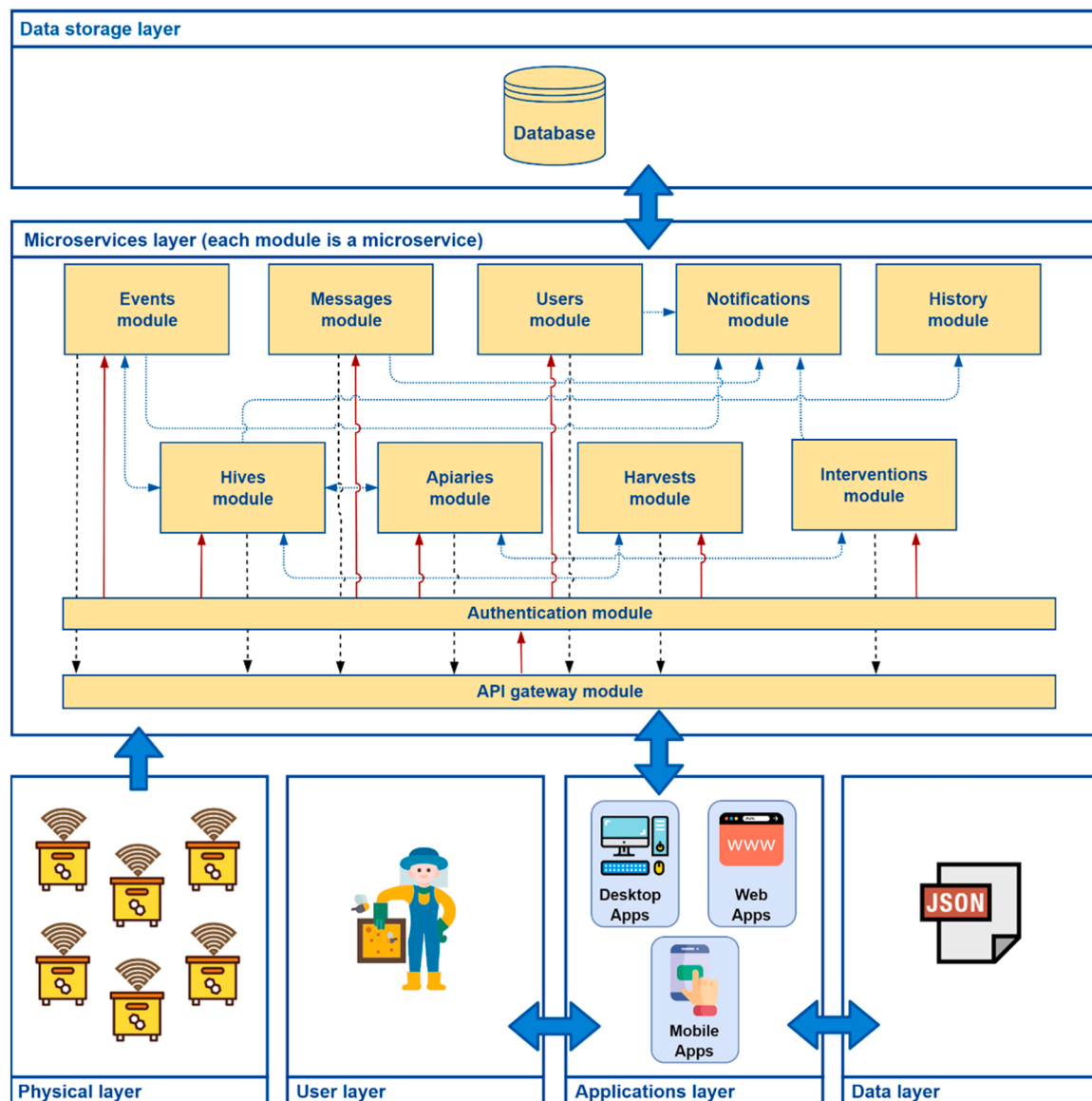


Fig. 3. BhivSense high-level system architecture.

architecture represented in Fig. 3. This architecture consists of six different layers: the physical layer, the user layer, the applications layer, the data layer, the microservices layer, and the data storage layer. Layers are interconnected with each other depending on the flow of data between them. Data flows from the physical layer to the microservices layer and then to the user layer thru the applications layer. Other services can be provided to the users (beekeepers) by several microservices. These services can or not depend on the data sent by the physical layer.

Physical layer: the physical layer corresponds to the hive sensing nodes network (each node defined by a set of subsystems including sensors, communications, processing, and power) and an Internet gateway. Several different types of sensors can be placed inside and outside the hive, depending on the variables to monitor. As for the Internet gateway, a mobile data router is proposed for TCP/IP communications with the web server via HTTP requests (network layer). Yet, there is no dependency on the chosen technology since the same data specification format is used.

User layer: the user layer is mainly composed by the beekeepers as the main stakeholders of the system. These stakeholders are the final users of the system in terms of the Precision Beekeeping context. In this work, and concerning the context of the proposal, the main stakeholders are the beekeepers. However, others could be identified, such as the system administrators or researchers in this field. Even though typical IoT architectures don't include a user layer and given that this work proposes a solution squarely contextualised in the field of Precision Beekeeping, we believe that underlining the set of stakeholders for this context in a dedicated layer is relevant.

Application layer: different categories of software applications can be developed to use the collected data in the scope of an apiary monitoring and management system. Typical desktop, Web and mobile applications are proposed as possible solutions to deliver the entire features of the system, in the application layer. Also, this layer represents the user interfaces (front end) with which the stakeholders interact with the system. Independently from the type of application (desktop, Web, mobile or hybrid), the proposed REST-based architecture allows the integration of monitoring, management, and events notification preconised in this approach.

Data layer: the data layer represents the data format specification for the proposed architecture. Different data formats can be provided by the data layer (e.g., XML, CSV, JSON, etc.) since REST-based APIs can handle messages in practically all data formats. We propose JSON due to its popularity, smaller size, and being more easily readable by humans. All messages flowing thru the proposed architecture use the JSON data format.

Microservices layer: the microservices layer consists of a set of microservices to manage every data transaction and can have independent databases and/or data models. However, the more important feature of a microservice approach is that each one should have a single responsibility. For example, in the proposed solution, the HIVE module is only responsible for: (1) handling data between the physical layer and the client applications (applications layer), in a context of interoperability between components; (2) providing analytics regarding hive data upon request from the users. This layer is identical to the service layer described in the four-layer architecture approach.

Data storage layer: the data storage layer is the layer responsible for persistent data storage, not only to provide real-time data to the user layer but also to deliver long-term data for analytics and apiary management purposes requested by other microservices. As per the user layer, we propose this layer to better underline the importance of data storage in the overall solution.

4.4. Demonstration of the proposed architecture

The main data source for the proposed apiary monitoring system architecture is a physical device installed in each apiary hive. This device can gather different types of data, as much as is needed, not having

any impact on the system architecture, since the same data format is followed.

According to the DSR methodology adopted for this work, the demonstration step is crucial to assess the validity of any proposed artefact, by creating a suitable context and developing test/pilot artefacts for evaluation purposes.

To demonstrate the proposed architecture, three pilot instances were developed: a node hive device (physical layer), a RESTful microservices-based API (microservices layer), and a Web and a mobile application (applications layer). These pilot artefacts are described below.

4.4.1. Hive node sensing device: hardware and firmware

In the context of providing a hive node prototype to demonstrate the proposed architecture, a popular NODE MCU ESP8266 low-cost microcontroller was used. This unit is an open-source software and hardware development environment built around an inexpensive System-on-a-Chip (SoC) named ESP8266. The ESP8266 contains the crucial elements of a computer: CPU (32 bit-80Mhz), RAM (4MB), and networking (Wi-Fi). That makes it an excellent choice for IoT projects of all kinds. In terms of collected data, we selected temperature (inside and outside the hive), sound (inside the hive) and the weight of the hive. These variables are commonly referred to in related works as the most important for detecting the hive health, production levels and other dynamic honey-bee behaviour like swarming. Isolated or combined, these variables provide crucial information for the context of this project.

To monitor temperature and humidity inside and outside the hive, we used two low-cost and commonly used DHT22 temperature and humidity sensors. Concerning sound measurement, an Adafruit electret microphone (20–20KHz), with a MAX4466 amplifier and adjustable gain is used. As for weight, four strain gauges connected in a Wheatstone Bridge configuration were used with an HX711, a precision 24-bit analogue-to-digital converter (ADC) designed for weigh scales and industrial control applications, to interface directly with the bridge sensor. An additional GNSS (Global Navigation Satellite System) module is included to provide the geolocation of the hive and spot its position in a digital map (e.g., Google Maps), thus allowing the beekeeper to best find the location of a hive for intervention. The addition of this element can also enable future integrations in other research projects, for example, studies requiring spatial data and cross-information between colony data and the geographic position of the hives. The artefact also includes several mechanical sensors: two press buttons, one for resetting the GNSS module to provide new geographical coordinates (in case the hive switches apiary), one to calibrate the weight sensor, and a microswitch to detect if the hive lid has been opened to trigger an event alarm to the beekeeper.

The proposed demonstration prototype is powered by a 7 V photovoltaic cell with a 3.3 V-1500mAh lithium-polymer (Li-Po) battery, controlled by a solar power manager for energy autonomy. The overall system schematic is shown in Fig. 4. Fig. 5 shows the installation in situ of the BHiveSense node hive prototype.

The hive node prototype firmware is coded in C/C++. However, any programming language could be used if another MCU or development board is used. The only important aspect is the data format in the scope of this solution. Before commissioning the hive node, the Internet gateway credentials (node ID, network SSID, and password) must be saved to the device while uploading the firmware. In the first iteration of the loop function, the system waits for GNSS fixing to get and save the latitude and longitude of the device in the Electrically Erasable Programmable Read-Only Memory (EEPROM). In case the beekeeper decides to install the same node in a different hive, located in a different apiary, the EEPROM address containing the coordinates is cleared and the procedure to get the new coordinates is repeated for the new location. After this, the process reiterates: get updated readings from the sensors, turn the Wi-Fi connection ON and connect to the gateway, send HTTP POST request with the updated JSON body containing the last sensor readings, plus the device credentials, until receiving an HTTP

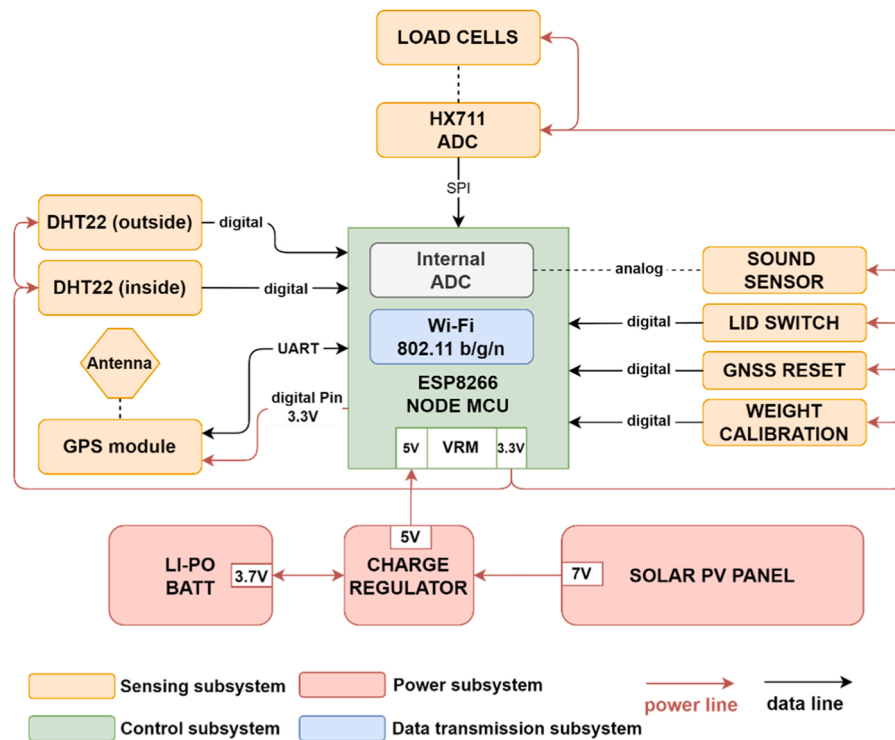


Fig. 4. BhiveSense hive node prototype – schematic.



Fig. 5. BhiveSense hive node prototype – hardware.

response status of 200. After a successful POST request, the system idles for 10 min until it wakes up and a new iteration of the process starts. Fig. 6 shows the flow diagram of the hive node firmware.

4.4.2. Data model

In the proposed architecture the data storage layer stores persistent long-term data to support apiary monitoring and management features, compatible with the Precision Beekeeping paradigm. Despite the ability to have different databases for different microservices, in this solution, we use a single MongoDB NoSQL database with several JSON data collections, each one linked with a separate microservice. Fig. 7 shows the data model schema for this project. The connection between each entity and the corresponding microservice is obvious. Several collections are proposed for normalisation features. The attribute data in the HIVE entity is an array of JSON instances. One instance is automatically created for each day, and the dayData array saves each JSON instance sent from the hive node device. This approach makes the analytics process faster and more efficient, for example, getting weekly or monthly data for the hive.

4.4.3. RESTful API

The proposed backend architecture follows a modular, flexible, and robust architecture built under the HTTP protocol and REST, an architectural pattern known to ease Web service development (Cebeci and Korçak, 2020). Due to the nature of HTTP, request/response communication are carried on synchronously. HTTP delivers this type of communication through blocking and awake approach. The use of an API (Application Programming Interface) with the REST architecture provides efficiency, procedure automation, and interoperability as the main advantages. REST also allows independent implementations from the server and client sides. The client only needs to know the endpoints to reach the resources delivered by the API and a known message format. In this case, we use JSON.

Implementing the API gateway in a RESTful Web approach eases the use of microservices methods providing an interface, keeping it as simple as possible in our solution.

The demonstration backend is implemented in NodeJS following the MVC protocol and the CRUD paradigm (Create, Read, Update, Delete) which corresponds to the methods POST, GET, PUT and DELETE in an

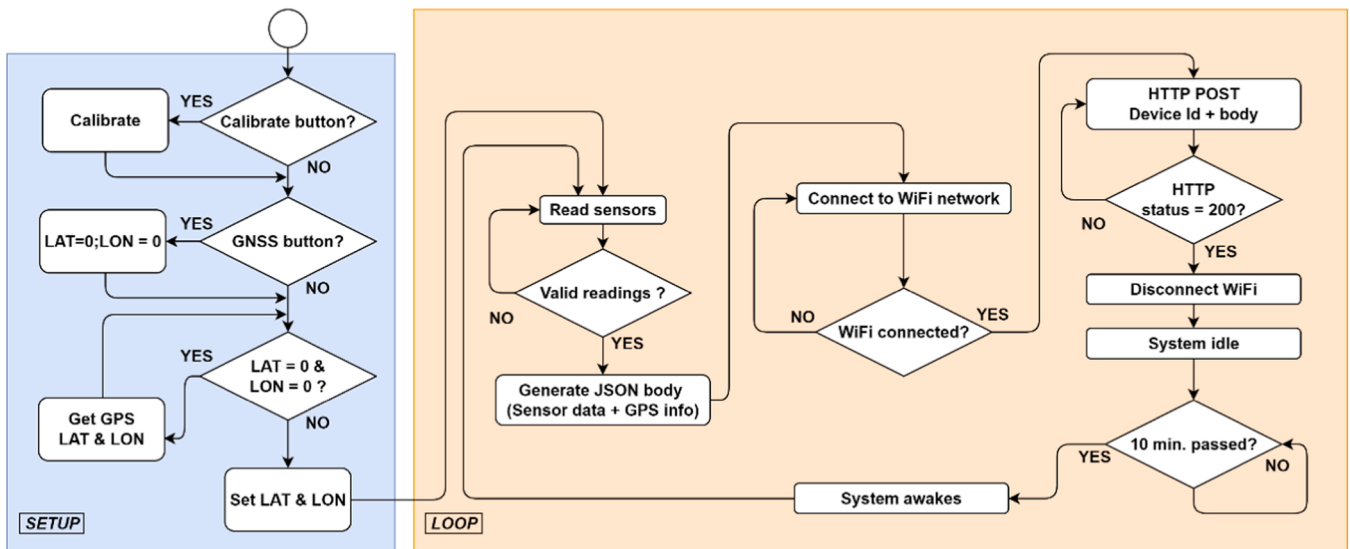


Fig. 6. BhiveSense hive node prototype firmware - flow diagram.

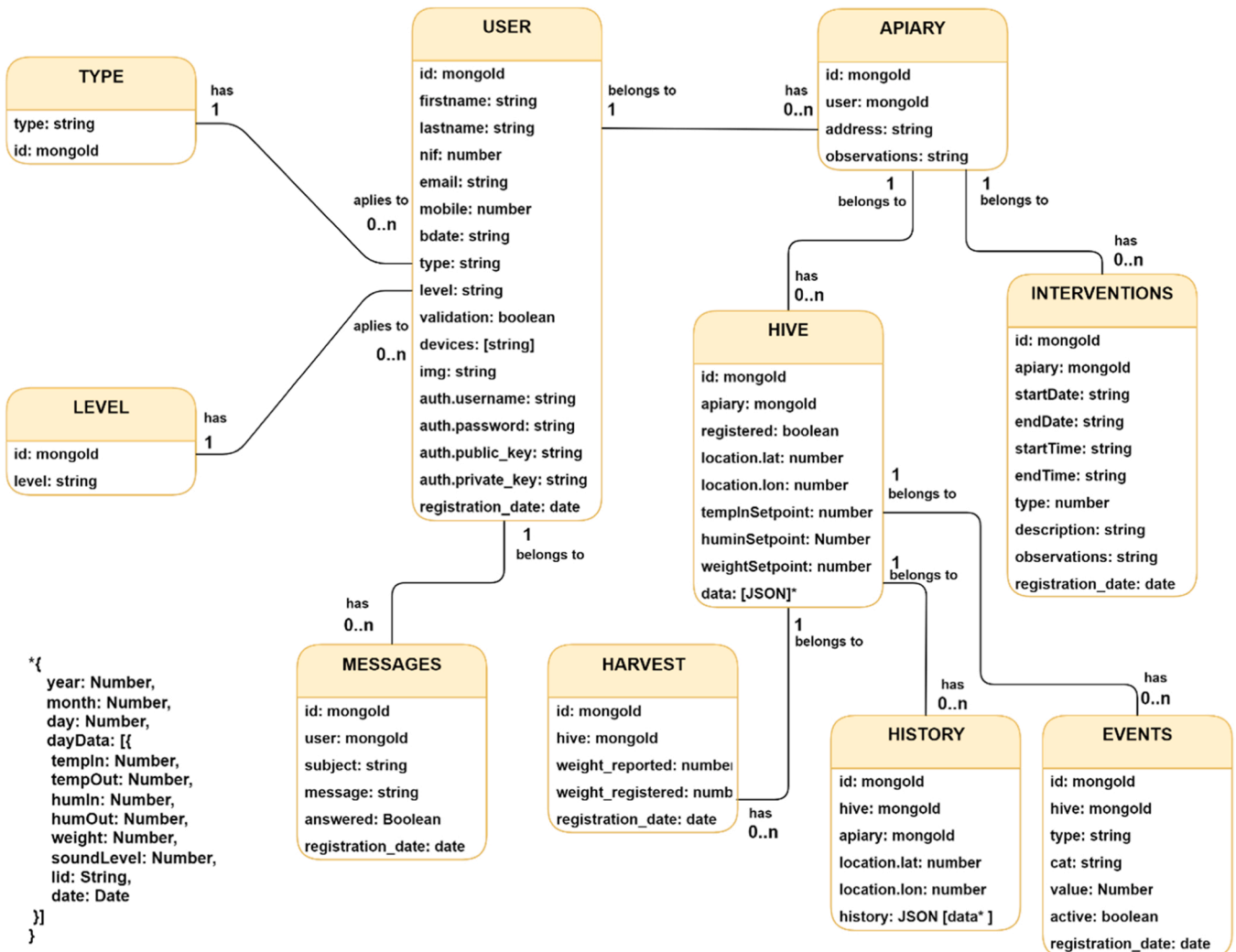


Fig. 7. BhiveSense data model.

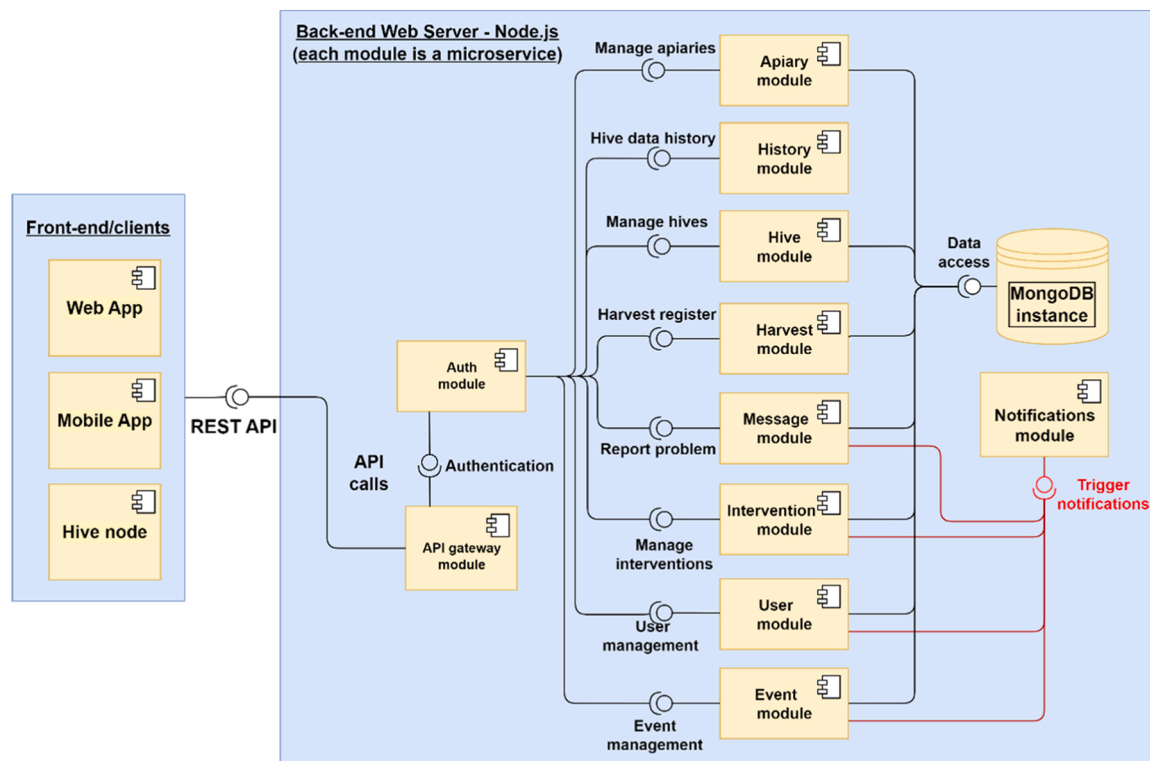


Fig. 8. BhiveSense component diagram (API).

HTTP environment. Fig. 8 shows the component diagram for the API and backend.

4.4.4. API gateway module

the API gateway module is the entry point for the backend application. This module handles the HTTP requests from the clients and routes them to the relevant module. The routing feature is a fundamental technique in a REST API. This module also runs a daily routine (CRON JOB) to detect if there are any near interventions scheduled for the next days ($0 \geq \text{numDays} \leq 5$) to notify the beekeeper.

4.4.5. Auth module

this module contains all the login and authentication methods to secure the application. All request handling is subject to authentication control (token) before routing to the queue correspondent microservice.

4.4.6. User module

is responsible for managing the user login and authentication, providing the tokens that are delivered in the response header and the necessary user data to the client. It also provides methods for user registration and account management.

Apiary module: the Apiary module manages requests to perform all the operations concerning apiaries (e.g., create an apiary, update apiary data, delete the apiary, get apiaries for a specific user, etc.).

Hive module: this module delivers methods to assign/unassign hives (node devices) to a specific user, register the hives in an apiary, and return the hive data to the users. It also has a method to handle the requests sent from the node devices to register a new data package.

Interventions module: the Interventions module can register scheduled interventions for an apiary and handle requests to update, delete, or change its state to conclude. These microservices are related to apiary management, one of this solution's goals in the Precision Beekeeping paradigm.

Harvest module: this module is responsible for registering harvest actions for each hive/apiary and delivers methods for production

analytics by hive, apiary, or user.

Message module: this is the module that groups methods to handle messages from the beekeepers (users) to the platform administration in case of the need to report any issue or problem with the system.

Event module: this module handles requests related to events. It delivers the methods to create events and to change their status. This is the crucial microservice to trigger alarms to the beekeeper. An event is a particular situation that arises from hive data analysis in real-time. In this project, events are triggered in the following conditions: temperature or relative humidity inside the hive above the setpoint; the hive lid is open; the harvest setpoint has been reached; or a possible swarm situation has been detected.

Notifications module: this module supplies the method and services to generate and send notifications to the users in case of new events or near future interventions scheduled. It delivers services to send e-mails or push notifications to the application layer instances. In this solution, we use an external microservice from OneSignal that eases sending a web and mobile push notifications.

History module: in case the beekeeper decides to move a hive from one apiary to another, there are important procedures to carry out. First, the new location will be a different geographical point. Second, the beekeeper should have a way to decide about the accumulated data in the previous location. The history module provides the possibility to save the collected data when a hive is unassigned from an apiary, for future analytical proposes.

4.4.7. User applications

4.4.7.1. Web application. To demonstrate the proposed system architecture, a web application was developed. This artefact was developed in a modern SPA (Single Page Application) approach, using the Vue.JS framework and several specific packages. The application offers distinct features to the beekeeper, according to project goals and identified problems in the scope of Precision Beekeeping. In the application, there are two main user levels: administrator and beekeeper.

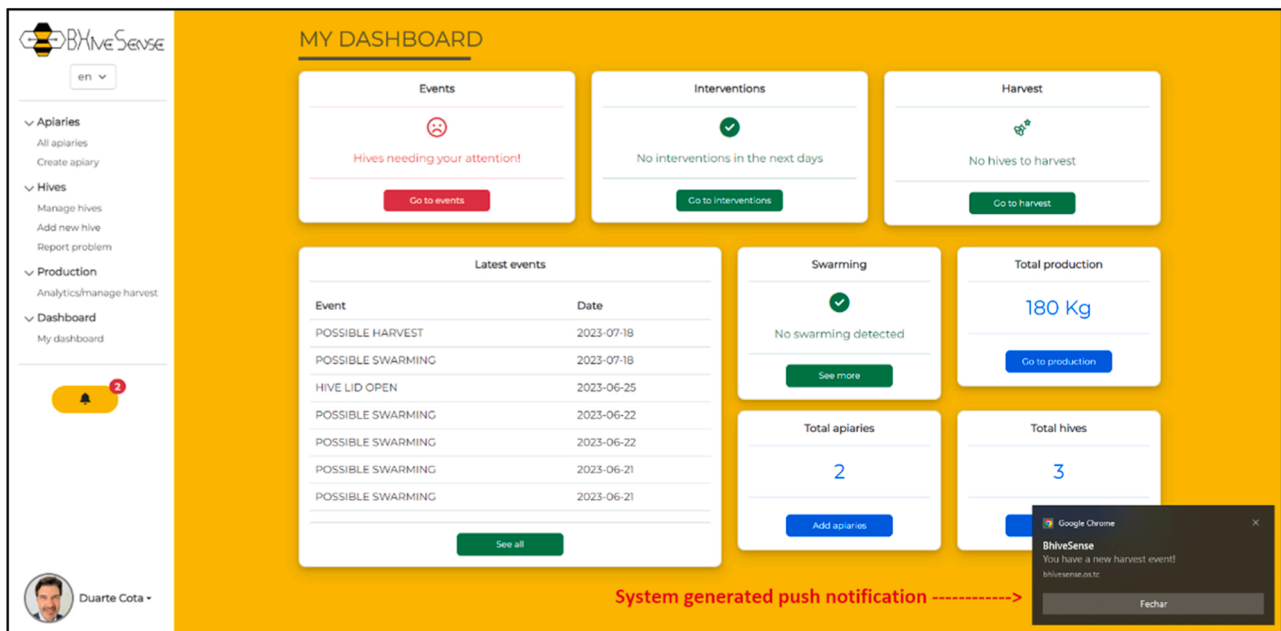


Fig. 9. BhiveSense Web app dashboard showing a notification (Web).

The first has access to different tools such as user administration, registering or managing devices in the system, or replying to beekeepers' messages. The beekeepers have access to different options, for example, to visualise aggregated information in a simple dashboard (Fig. 9), perform CRUD operations over apiaries or devices (hives), monitor detailed hive data (Fig. 10), view and manage event logs (Fig. 11), and several others feature like register harvest activities, monitor production by apiary or hive, and receive push notifications for events and alarms.

This platform also includes an intervention management service that allows the beekeeper to schedule interventions in each apiary, being notified in the dashboard and via e-mail when the intervention date is coming (less or equal to five days). The focus is on the information about important events that occurred in the hives such as abnormal values of

temperature or humidity inside the hives, the hive lid opening, possible swarm events or near future interventions to come.

The dashboard also shows the need for harvesting using the weight parameter and gives access to a complete log of events. The beekeeper also has the correct tools to register each harvest to control the production by apiary or hive.

4.4.7.2. Mobile application. In addition to the web application, an extra demonstration artefact was developed, consisting of a mobile application (Android version). This tool was also implemented using a modern concept, with the framework Flutter, developed by Google. The application allows the beekeeper to monitor the state of the apiaries in a simple dashboard (Fig. 12a). The apiaries can be monitored individually (Fig. 12b), accessing their associated hives (Fig. 12c). Information about

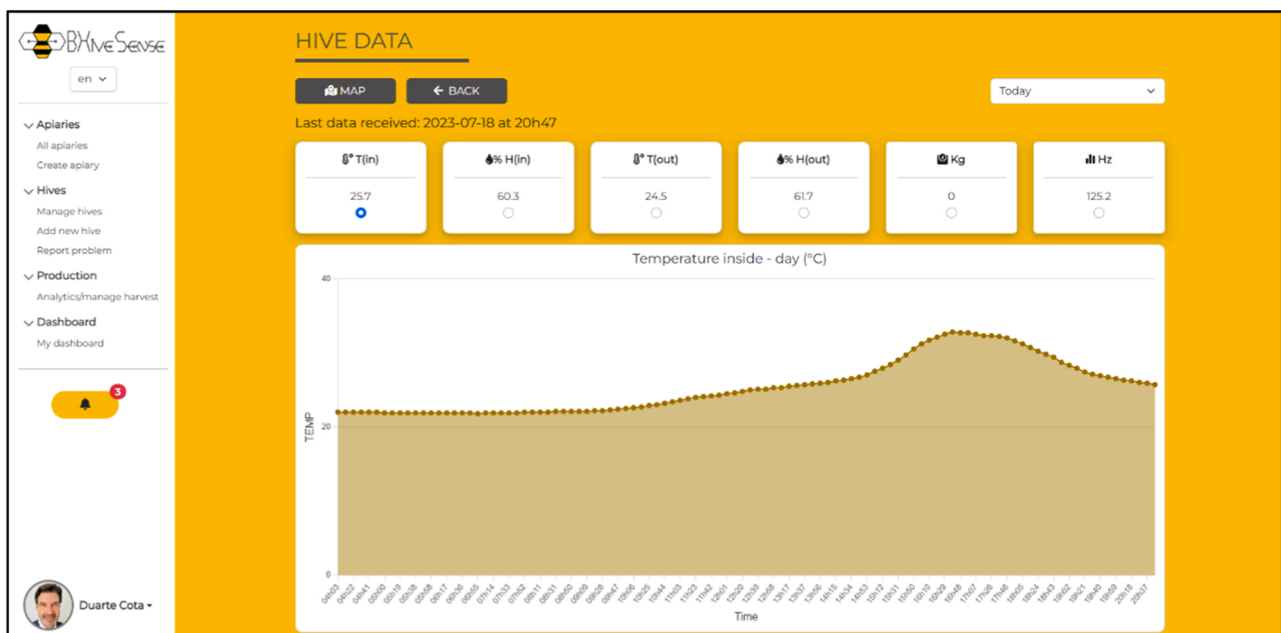


Fig. 10. BhiveSense detailed hive data example (Web).

Notification	Hive	Date	Apiary
POSSIBLE HARVEST ACTIVE	63d14e22a290886b1524564d	2023-07-18	
POSSIBLE HARVEST SOLVED	63d14e22a290886b1524564d	2023-07-18	
POSSIBLE SWARMING SOLVED	63d14e22a290886b1524564d	2023-07-18	
HIVE LID OPEN SOLVED	645aad9081943e7023cee294	2023-06-25	
POSSIBLE SWARMING SOLVED	63d14e22a290886b1524564d	2023-06-22	
POSSIBLE SWARMING SOLVED	63d14e22a290886b1524564d	2023-06-22	
POSSIBLE SWARMING SOLVED	63d14e22a290886b1524564d	2023-06-21	
POSSIBLE SWARMING SOLVED	63d14e22a290886b1524564d	2023-06-21	
POSSIBLE HARVEST SOLVED	63d14e22a290886b1524564d	2023-06-20	
INTERIOR TEMPERATURE HIGH ACTIVE	63d14e22a290886b1524564d	2023-03-27	
POSSIBLE HARVEST SOLVED	63d14e22a290886b1524564d	2023-03-15	
HIVE LID OPEN ACTIVE	63d14e22a290886b1524564d	2023-03-12	

Fig. 11. BhiveSense event log for user devices (Web).

each hive parameter and the general hive state can be accessed on a dedicated screen (Fig. 12d). Both applications (web and mobile) also use the GNSS feature of the device prototype to give a geolocation dimension to the system (Fig. 12e). Like the web version, this application receives push notifications in case of the occurrence of important events that can indicate anomalies demanding the beekeeper intervention: hive inside temperature our air humidity higher than the setpoint, hive lid open, harvest setpoint reached or possible swarm situation (Fig. 12a), allowing a fast reaction from the beekeeper to prevent situations that could impact in the apiary production, or in the colony's health and behaviour.

4.4.7.3. Evaluation. Conducting practical tests on an IoT-based monitoring solution for beehives presents several notable challenges that require thoughtful consideration (Wrysinski et al., 2023). Firstly, the ever-changing and unpredictable natural environment of beehives, with its diverse weather conditions, temperature fluctuations, and varied bee behaviour patterns, poses uncontrollable factors that can impact data collection and overall system performance. Ensuring consistent and reliable results becomes a demanding task. Secondly, the introduction of new technology like IoT devices into beehives might disturb or alter the behaviour of bee colonies. As bees are highly sensitive creatures, any disruptions to their habitat could cause stress and affect their natural activities, potentially influencing the accuracy of the collected data and the effectiveness of the monitoring solution. Moreover, the dispersion of beehives across different geographical locations creates logistical challenges for deploying and maintaining the IoT devices. Overcoming obstacles such as accessing remote areas, ensuring stable network connectivity, and powering the devices in isolated locations demands careful planning and resource allocation. Another crucial concern is the durability and safety of the IoT devices used in beehives. They must withstand harsh environmental conditions, including extreme temperatures and moisture. Additionally, ensuring the quality, reliability, security and privacy of the collected data is of utmost importance. Any data bias or breaches could have serious implications for both the beekeepers and the bee colonies (Hadjur et al., 2022).

Hence, real-life testing requires prolonged observation periods to understand the long-term performance and reliability of the IoT-based monitoring solutions. This extended testing demands significant time and resources, potentially delaying the evaluation of the system's

effectiveness and hindering the pace of improvements or modifications (Abdollahi et al., 2022).

Therefore, considering the aforementioned factors and taking inspiration from similar research (Heaton and Parlikad, 2019; Paganelli et al., 2022; Tamburis et al., 2020) where conceptual proposals underwent only initial validation but emphasized the need for real-life testing to assess all contextual aspects and biases, a series of initial evaluation procedures were undertaken to validate the adequacy of the proposed components, their compatibility, and interoperability. This step was critical to assess the proposed architecture and we used demonstration artefacts for this validation.

As established by existing literature (Jacob and Mani, 2018; Popere-shnyak et al., 2018), in order to validate the proposed artefact, an initial set of different types of tests was performed: unit tests, integration tests, and acceptance tests.

Unit tests were created and performed during the development of the demonstration artefacts to test each service provided by the REST API, individually, and ran often, as payloads and logic involved. These tests, performed using the Insomnia REST Client application (Insomnia, 2023), indicated that all the used sensors were able to individually perform according to the manufacturer specifications, thus allowing to collect data within expected sets of values.

To test and evaluate the integration of the different services, several integration tests were performed in a top-down approach depending on the coverage of the components. These tests allowed us to evaluate if the several components and services work together in an orchestrated way. In order to perform the referred integration tests all sensing nodes were activated in parallel and were set in a "data collection" mode that enabled each one of them to effectively and simultaneously collect data and save it in the data storage module. After analysing the data collection and communication log it was possible to perceive that all activated sensing modules performed correctly and within the expected accuracy and performance levels.

Finally, acceptance tests validate the user requirements (beekeepers), specifically if the several operations and monitoring features related to the Precision Beekeeping paradigm are implemented in the solution. These validations were based on a set of dynamic tests, using the entire system architecture. The REST API was deployed in a shared cloud server service, while the web and mobile applications were running in a local environment. The monitoring hardware was

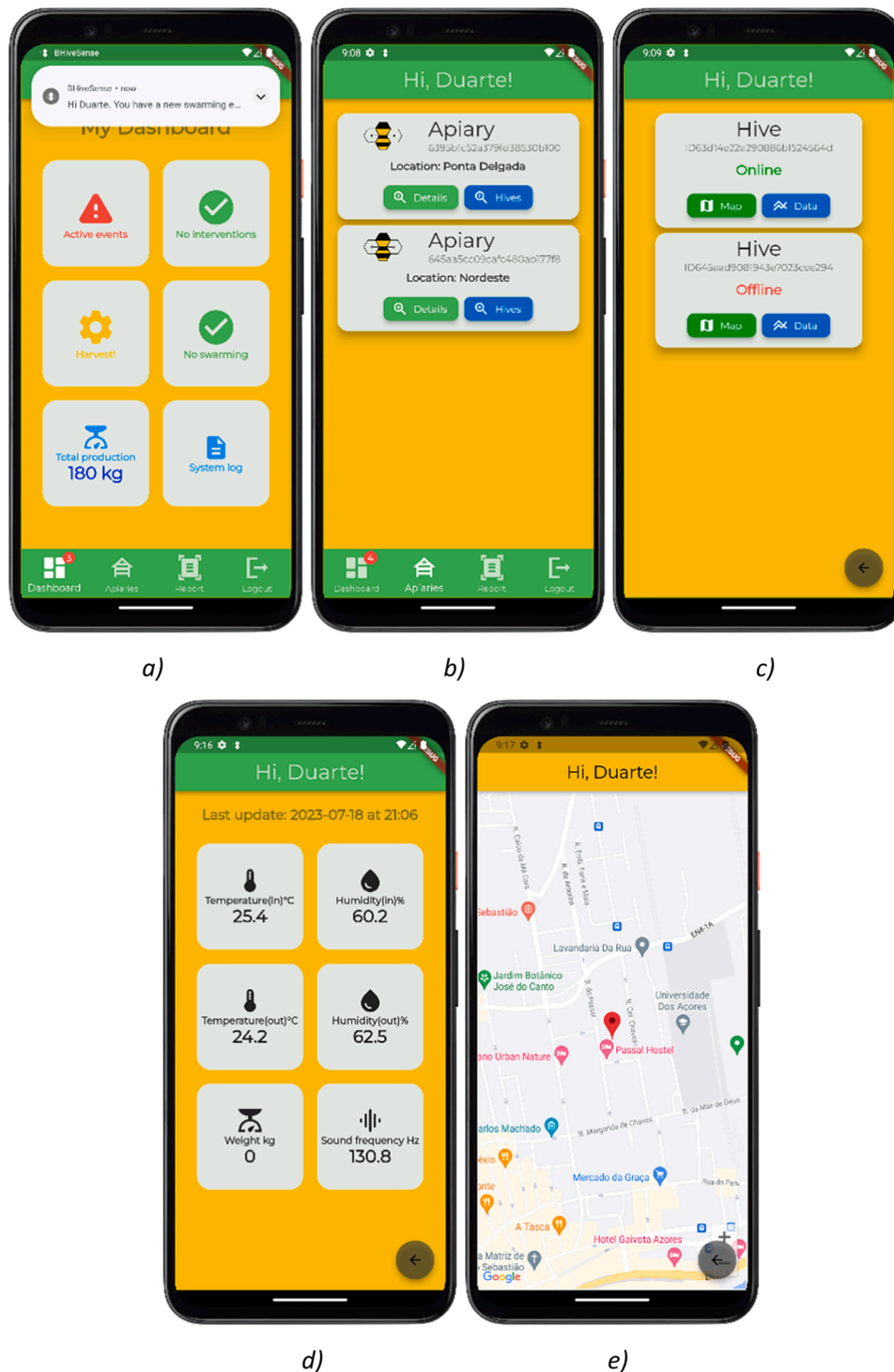


Fig. 12. (a) Mobile app dashboard showing notification; (b) List of apiaries; (c) List of hives in one apiary; (d) Hive detailed data example; (e) Location of the hive in Google Maps.

commissioned for five continuous months in a controlled environment composed of a beehive installed in similar conditions as it would be in a real apiary but without a bee colony installed inside. During this period, several events were induced to test the response of the system and validation of the architecture. For example, the hive lid was open, the temperature and humidity were intentionally raised, or sounds with frequencies compatible with swarming were simulated to test the response of the sensors and the action of the microservice responsible for triggering the events and notifying the beekeeper.

4.5. Communication

The final step of DSR-based research is the communication of the entire project development and results, including the relevance of the research question, the proposed artefact, its design, innovation, and functionality must be delivered, disclosing the achievements to other researchers and related relevant stakeholders.

This paper itself is integrated into this step. Considering the research and development stage of this work, future actions to communicate this

work will be addressed, with planned dissemination sessions and workshops in the apiculture section of the Agricultural Development Services of the Azores (São Miguel Island), to engage beekeepers with the proposed solution.

5. Conclusions and future work

Monitoring beehives and apiaries is critical to preserving both honeybee numbers and the delicate ecological balance that they support. Honeybees contribute greatly to agriculture as pollinators by supporting the reproduction of numerous fruit, vegetable, and nut crops. Beekeepers can monitor the health of colonies, detect and manage possible hazards such as diseases, pests, or environmental stressors, and implement appropriate remedies by frequently observing beehives. Timely monitoring also helps to reduce colony losses, which promotes food security and economic stability. Furthermore, knowing honeybee behaviour through monitoring allows researchers to acquire insights into their sophisticated social structure and communication patterns, allowing for the development of science-based conservation policies that are both sustainable and long-term. We can protect these important pollinators by investing in careful monitoring procedures.

Using IoT-based monitoring systems and solutions provide vital aid in the abovementioned monitoring and administration of beehives and apiaries, thus offering an innovative approach to beekeeping. These smart systems collect real-time data on critical parameters such as temperature, humidity, hive weight, and even sound and movement patterns via a network of interconnected sensors, cameras, and gadgets installed within the beehives. This constant flow of data enables beekeepers to remotely monitor and assess hive conditions, allowing them to respond to any irregularities or signals of stress inside the colony as soon as possible. By configuring alerts and notifications, beekeepers can receive instant indications if conditions deviate from ideal, preventing potential problems. Furthermore, IoT monitoring solutions provide data insights and analytics that enable beekeepers to make more informed decisions.

Microservices architectures have emerged as a powerful approach to building and managing complex systems, including IoT (Internet of Things) applications. In a microservices architecture, a large application is broken down into small, independent, and loosely coupled services, each responsible for specific functionalities. This modular design allows for greater flexibility and scalability, especially in IoT systems where the number of connected devices and data volume can vary significantly. By adopting microservices, IoT applications can easily scale up or down based on demand, as each service can be individually deployed, upgraded, or replicated as needed. This elasticity ensures that the system can efficiently handle varying workloads, making it well-suited to the dynamic and evolving nature of IoT environments. Additionally, microservices promote rapid development and deployment, allowing IoT applications to adapt quickly to changing requirements and technological advancements, while also enabling easier maintenance and fault isolation.

With the above in mind the present research proposes a novel low-cost IoT-based monitoring solution supported by a microservices architecture and aimed at continuously monitor beehives and apiaries. From a formal perspective, the proposed solution well-framed and interoperable six-layer architecture ensures the needed agility and scalability that beekeepers require.

In order to ensure the necessary validity to the proposed artifact, we have developed a prototype of the conceptual solution that has been tested using a controlled simulation environment. The achieved results allowed to attain that the proposed solution fulfils all the initially established functional and performance requirements, thus allowing us to consider it as a valuable contribute to both science and practice.

5.1. Limitations and future work

This work has some primary limitations. Firstly, only one prototype for the hive node device was developed and tested. Although the tests were conducted in a natural laboratory with varying weather conditions, there were no tests conducted with an actual bee colony installed inside the hive. In the future, to fully validate the node sensing prototype's response in a broader and long-term context, it is essential to produce and install multiple identical hive nodes in different apiaries located in various places. This validation will also involve testing the hardware prototype with bee colonies installed in the hives. The plan is to collaborate with the local Agricultural Development Services of the Azores (São Miguel Island, Azores), who have agreed to deploy the proposed system in several of their apiaries for these tests. However, securing funds for replicating the devices is necessary for carrying out these tests. Despite these limitations, the tests conducted with the developed demonstration artifacts are considered sufficient to validate the functionality, scalability, and flexibility of the proposed architectural solution.

Supplementary materials

Not applicable.

Informed consent statement

Not applicable.

Funding

This work was supported by national funds through the Portuguese Funding Agency, the Fundação para a Ciência e a Tecnologia (FCT) within project LA/P/0063/2020.

Ethical statement

Not applicable as no human subjects were included as participants in the research activities that were conducted.

CRediT authorship contribution statement

Conceptualization, D.C., F.B. and J.M.; Methodology, J.M., D.C., and F.B.; Investigation: D.C., J.M., F.B., and H.M.; Writing – original draft, D. C., J.M., F.B., and H.M.; Writing – review & editing, D.C., J.M., F.B., and H.M.; Supervision, F.B. and J.M.; Funding acquisition, H.M. and J.M. All authors have read and agreed to the published version of the manuscript.

Declaration of Competing Interest

none.

Acknowledgements

JM is grateful to the FCT for the projects titled "AquaValor—Centro de Valorização e Transferência de Tecnologia da Água" (NORTE-01-0246-FEDER-000053) and "Emprego altamente qualificado nas empresas ou em COLABS—Contratação de Recursos Humanos Altamente Qualificados (PME ou CoLAB)" (NORTE-06-3559-FSE-000095), supported by the North Portugal Regional Operational Programme (NORTE 2020) under the PORTUGAL 2020 Partnership Agreement through the European Regional Development Fund (ERDF).

References

- Abbasi, R., Martinez, P., Ahmad, R., 2022. The digitization of agricultural industry – a systematic literature review on agriculture 4.0. *Smart Agric. Technol.* 2, 100042 <https://doi.org/10.1016/j.atech.2022.100042>.
- Abdollahi, M., Giovenazzo, P., Falk, T.H., 2022. Automated beehive acoustics monitoring: a comprehensive review of the literature and recommendations for future work. *Appl. Sci.* 12 (8), 3920.
- Aliferis, C., Chamaidi, T., Malisova, K., Mamalis, D., Nomikos, E., Rigakis, C., Vlachogiannis, E., & Stavrakis, M. (2023). IOHIVE: Architecture and Infrastructure of an IOT System for Beehive Monitoring and an Interactive Journaling Wearable Device for Beekeepers. 133–149.
- Andrijević, N., Urošević, V., Arsić, B., Herceg, D., Savić, B., 2022. IoT monitoring and prediction modeling of honeybee activity with alarm. *Electronics* 11 (5), 783. <https://doi.org/10.3390/electronics11050783>.
- Anwar, O., Keating, A., Cardell-Oliver, R., Datta, A., Putrino, G., 2023. Apis-Prime: a deep learning model to optimize beehive monitoring system for the task of daily weight estimation. *Appl. Soft Comput.* 144, 110546 <https://doi.org/10.1016/j.asoc.2023.110546>.
- Araújo, S.O., Peres, R.S., Barata, J., Lidon, F., Ramalho, J.C., 2021. Characterising the agriculture 4.0 landscape—Emerging trends, challenges and opportunities. *Agronomy* 11 (4), 667.
- Atitallah, S.B., Driss, M., Ghzela, H.B., 2022. Microservices for data analytics in IoT applications: current solutions, open challenges, and future research directions. *Procedia Comput. Sci.* 207, 3938–3947.
- Aydin, S., Nafiz Aydin, M., 2022. Design and implementation of a smart beehive and its monitoring system using microservices in the context of IoT and open data. *Comput. Electron. Agric.* 196, 106897 <https://doi.org/10.1016/j.compag.2022.106897>.
- Bellini, F., Turvani, G., Garlando, U., Riente, F., 2022. An integrated multi-sensor system for remote bee. *Health Monit.* 334–338.
- de Boon, A., Sandström, C., Rose, D.C., 2022. Governing agricultural innovation: a comprehensive framework to underpin sustainable transitions. *J. Rural Stud.* 89, 407–422.
- Butzin, B., Golatowski, F., & Timmermann, D. (2016). Microservices approach for the internet of things. 1–6.
- Calvo, I., Espin, A., Gil-García, J.M., Fernández Bustamante, P., Barambones, O., Apinániz, E., 2022. Scalable IoT architecture for monitoring IEQ conditions in public and private buildings. *Energies* 15 (6), 2270.
- Catania, P., Vallone, M., 2020. Application of A precision apiculture system to monitor honey daily production. *Sensors* 20 (7), 2012. <https://doi.org/10.3390/s20072012>.
- Cebeci, K., Korçak, Ö., 2020. Design of an enterprise level architecture based on microservices. *Bilişim Teknol. Derg.* 13 (4), 357–371. <https://doi.org/10.17671/gazibtd.558392>.
- Cecchi, S., Spinsante, S., Terenzi, A., Orcioni, S., 2020. A smart sensor-based measurement system for advanced bee hive monitoring. *Sensors* 20 (9), 2726. <https://doi.org/10.3390/s20092726>.
- Chukkapalli, S., Piplai, A., Mittal, S., Gupta, M., & Joshi, A. (2020). A Smart-Farming Ontology for Attribute Based Access Control. 2020 IEEE 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS), 29–34. <https://doi.org/10.1109/BigDataSecurity-HPSC-IDS49724.2020.00017>.
- Dayioglu, M.A., Turker, U., 2021. Digital transformation for sustainable future-agriculture 4.0: a review. *J. Agric. Sci.* 27 (4), 373–399.
- Edwards-Murphy, F., Magno, M., Whelan, P.M., O'Halloran, J., Popovici, E.M., 2016. B+WSN: smart beehive with preliminary decision tree analysis for agriculture and honey bee health monitoring. *Comput. Electron. Agric.* 124, 211–219.
- Farooq, M.S., Riaz, S., Abid, A., Umer, T., Zikria, Y.B., 2020. Role of IoT technology in agriculture: a systematic literature review. *Electronics* 9 (2), 319. <https://doi.org/10.3390/electronics9020319>.
- Fitzgerald, D.W., Murphy, F.E., Wright, W.M. D., Whelan, P.M., & Popovici, E.M. (2015). Design and development of a smart weighing scale for beehive monitoring. 2015 26th Irish Signals and Systems Conference (ISSC), 1–6. <https://doi.org/10.1109/ISSC.2015.7163763>.
- Gan, Y., Delimitrou, C., 2018. The architectural implications of cloud microservices. *IEEE Comput. Archit. Lett.* 7 (2), 155–158. <https://doi.org/10.1109/LCA.2018.2839189>.
- Gil-Lebrero, S., Quiles-Latorre, F.J., Ortiz-López, M., Sánchez-Ruiz, V., Gámiz-López, V., Luna-Rodríguez, J.J., 2016. Honey bee colonies remote monitoring system. *Sensors* 17 (1), 55.
- Gil-Lebrero, S., Navas González, F.J., Gámiz López, V., Quiles Latorre, F.J., Flores Serrano, J.M., 2020. Regulation of microclimatic conditions inside native beehives and its relationship with climate in Southern Spain. *Sustainability* 12 (16). <https://doi.org/10.3390/su12166431>.
- Guandalini, I., 2022. Sustainability through digital transformation: a systematic literature review for research guidance. *J. Bus. Res.* 148, 456–471.
- Gupta, B.B., Quamara, M., 2020. An overview of internet of things (IoT): architectural aspects, challenges, and protocols. *Concurr. Comput.: Pract. Exp.* 32 (21) <https://doi.org/10.1002/cpe.4946>.
- Hadjur, H., Ammar, D., Lefèvre, L., 2022. Toward an intelligent and efficient beehive: a survey of precision beekeeping systems and services. *Comput. Electron. Agric.* 192, 106604 <https://doi.org/10.1016/j.compag.2021.106604>.
- He, W., Zhang, S., Hu, Z., Zhang, J., Liu, X., Yu, C., Yu, H., 2020. Field experimental study on a novel beehive integrated with solar thermal/photovoltaic system. *Sol. Energy* 201, 682–692. <https://doi.org/10.1016/j.solener.2020.03.054>.
- Heaton, J., Parlikad, A.K., 2019. A conceptual framework for the alignment of infrastructure assets to citizen requirements within a smart cities framework. *Cities* 90, 32–41. <https://doi.org/10.1016/j.cities.2019.01.041>.
- Hevner, et al., 2004. Design science in information systems research. *Hevner, A., March, S., Park, J., & Ram, S. (2004) MIS Q.* 28 (1), 75–105.
- Hong, W., Xu, B., Chi, X., Cui, X., Yan, Y., Li, T., 2020. Long-term and extensive monitoring for bee colonies based on internet of things. *IEEE Internet Things J.* 7 (8), 7148–7155. <https://doi.org/10.1109/JIOT.2020.2981681>.
- Imoize, A.L., Odeyemi, S.D., Adebisi, J.A., 2020. Development of a low-cost wireless beehive temperature and sound monitoring system. *IJEEI* 8 (3), 476–485. <https://doi.org/10.11591/ijeei.v8i3.2268>.
- Insomnia. (2023). Insomnia REST Client application (2023 v4.0) [Computer software]. (<https://insomnia.rest/>).
- ISPA. (2021). Precision Ag Definition | International Society of Precision Agriculture. (<https://www.ispag.org/about/definition>).
- Jacob, P.M., Mani, P., 2018. A reference model for testing internet of things based applications. *J. Eng. Sci. Technol.* 13 (8), 2504–2519.
- Jegan, S., Padmapriya, L., Manjari, P.V., & Shanmathi, R. (2021). Honey hive monitoring using IOT. 140023. <https://doi.org/10.1063/5.0070127>.
- Kasiotis, K.M., Zafeiraki, E., Kapaxidi, E., Manea-Karga, E., Antonatos, S., Anastasiadou, P., Milonas, P., Macher, K., 2021. Pesticides residues and metabolites in honeybees: a Greek overview exploring Varroa and Nosema potential synergies. *Sci. Total Environ.* 769, 145213 <https://doi.org/10.1016/j.scitotenv.2021.145213>.
- Khairul Anuar, N.H., Amri Md Yunus, M., Baharuddin, M.A., Sahlan, S., Abid, A., Ramli, M.M., Razzi Abu Amin, M., & Mohd Lotpi, Z.F. (2019). IoT Platform for Precision Stingless Bee Farming. 2019 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS), 225–229. <https://doi.org/10.1109/I2CACIS.2019.8825089>.
- Khujiatov, K., & Toshemirov, T. (2020). Wireless sensor networks based Agriculture 4.0: Challenges and apportions. 2020 International Conference on Information Science and Communications Technologies (ICISCT), 1–5. <https://doi.org/10.1109/ICISCT50599.2020.9351411>.
- Komasilovs, V., Zacepins, A., Kviesis, A., Fiedler, S., & Kirchner, S. (2019). Modular sensory hardware and data processing solution for implementation of the precision beekeeping [PDF]. 523.5Kb. <https://doi.org/10.15159/AR.19.038>.
- Kontogiannis, S., 2019. An internet of things-based low-power integrated beekeeping safety and conditions monitoring system. *Inventions* 4 (3), 52. <https://doi.org/10.3390/inventions4030052>.
- Kulyukin, V., Mukherjee, S., Amlathe, P., 2018. Toward audio beehive monitoring: deep learning vs. Standard machine learning in classifying beehive audio samples. *Appl. Sci.* 8 (9), 1573.
- Liu, Y., Li, D., Du, B., Shu, L., Han, G., 2022. Rethinking sustainable sensing in agricultural internet of things: from power supply perspective. *IEEE Wirel. Commun.* 29 (4), 102–109. <https://doi.org/10.1109/MWC.004.2100426>.
- Lyu, Z., Wei, H., Bai, X., Lian, C., 2020. Microservice-based architecture for an energy management system. *IEEE Syst. J.* 14 (4), 5061–5072. <https://doi.org/10.1109/JSYST.2020.2981095>.
- Maffezzoli, F., Ardolino, M., Bacchetti, A., Perona, M., Renga, F., 2022. Agriculture 4.0: a systematic literature review on the paradigm, technologies and benefits. *Futures* 102998.
- Marquez, G., Taramasco, C., Astudillo, H., Zalc, V., Istrate, D., 2021. Involving stakeholders in the implementation of microservice-based systems: a case study in an ambient-assisted living system. *IEEE Access* 9, 9411–9428. <https://doi.org/10.1109/ACCESS.2021.3049444>.
- Meikle, W., Weiss, M., Stilwell, A., 2016. Monitoring colony phenology using within-day variability in continuous weight and temperature of honey bee hives. *Apidologie* 47, 1–14.
- Miranda, J., Ponce, P., Molina, A., Wright, P., 2019. Sensing, smart and sustainable technologies for Agri-Food 4.0. *Comput. Ind.* 108, 21–36.
- Mohamed, M.Y., & Mansor, M.S. (2023). Measuring the Variability and Humidity of Apis mellifera Honeybee Hives by the Effect of some Ventilation and Indoor Fires During the Autumn Season Using a New System. 1158(3), 032015.
- Murphy, F.E., Magno, M., O'Leary, L., Troy, K., Whelan, P., & Popovici, E.M. (2015). Big brother for bees (3B) — Energy neutral platform for remote monitoring of beehive imagery and sound. 2015 6th International Workshop on Advances in Sensors and Interfaces (IWASI), 106–111. <https://doi.org/10.1109/IWASI.2015.7184943>.
- Navarro, J., Lima, F., Porto, M., Steinfeld, L., 2022. BeeSmart: A real-time remote monitoring and control system for beekeeping Sypm. . *Internet Things (SIoT)* 2022 14 doi: 10.1109/SIoT56383.2022.10070330.
- Ngo, T.N., Wu, K.-C., Yang, E.-C., Lin, T.-T., 2019. A real-time imaging system for multiple honey bee tracking and activity monitoring. *Comput. Electron. Agric.* 163, 104841.
- Ntawuzumuni, E., Kumaran, S., Sibomana, L., 2021. Self-Powered Smart Beehive Monitoring and Control System (SBMacS). *Sensors* 21 (10), 3522. <https://doi.org/10.3390/s21103522>.
- Paganelli, A.I., Velmovitsky, P.E., Miranda, P., Branco, A., Alencar, P., Cowan, D., Endler, M., Morita, P.P., 2022. A conceptual IoT-based early-warning architecture for remote monitoring of COVID-19 patients in wards and at home. *Internet Things* 18, 100399. <https://doi.org/10.1016/j.iot.2021.100399>.
- Panziera, D., Requier, F., Chantawannakul, P., Pirk, C.W., Blacquièrre, T., 2022. The diversity decline in wild and managed honey bee populations urges for an integrated conservation approach. *Front. Ecol. Evol.* 10, 767950.
- Patel, V., Pauli, N., Biggs, E., Barbour, L., Boruff, B., 2021. Why bees are critical for achieving sustainable development. *Ambio* 50, 49–59.

- Peffer, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S., 2007. A design science research methodology for information systems research. *J. Manag. Inf. Syst.* 24 (3), 45–77. <https://doi.org/10.2753/MIS0742-1222240302>.
- Poperezhnyak, S., Suprun, O., Suprun, O., & Wieckowski, T. (2018). IoT application testing features based on the modelling network. 127–131.
- Pyngkodi, M., Thenmozhi, K., Nanthini, K., Karthikeyan, M., Palarimath, S., Erajavignesh, V., & Kumar, G.B. A. (2022). Sensor based smart agriculture with IoT technologies: A review. 1–7.
- Ramachandran, V., Ramalakshmi, R., Kavin, B., Hussain, I., Almaliki, A., Almaliki, A., Elnaggar, A., Hussein, E., 2022. Exploiting IoT and Its enabled technologies for irrigation needs in agriculture. *Water* 14 (5), 719. <https://doi.org/10.3390/w14050719>.
- Ramson, S.R. J., Vishnu, S., & Shanmugam, M. (2020). Applications of Internet of Things (IoT) – An Overview. 2020 5th International Conference on Devices, Circuits and Systems (ICDCS), 92–95. <https://doi.org/10.1109/ICDCS48716.2020.243556>.
- Requier, F., 2019. Bee colony health indicators: Synthesis and future directions. *CABI Rev.* 2019, 1–12.
- Robustillo, M.C., Pérez, C.J., Parra, M.I., 2022. Predicting internal conditions of beehives using precision beekeeping. *Biosyst. Eng.* 221, 19–29.
- Roy, S.K., De, D., 2022. Genetic algorithm based internet of precision agricultural things (IopaT) for agriculture 4.0. *Internet Things* 18, 100201.
- Sari, F., Kandemir, İ., Ceylan, D.A., Gül, A., 2020. Using AHP and PROMETHEE multi-criteria decision making methods to define suitable apiary locations. *J. Apic. Res.* 59 (4), 546–557.
- Schneider, C.W., Tautz, J., Grünwald, B., Fuchs, S., 2012. RFID tracking of sublethal effects of two neonicotinoid insecticides on the foraging behavior of *Apis mellifera*. *PLoS One* 7 (1), e30023. <https://doi.org/10.1371/journal.pone.0030023>.
- Sharma, P.K., Chen, M.-Y., Park, J.H., 2017. A software defined fog node based distributed blockchain cloud architecture for IoT. *IEEE Access* 6, 115–124.
- Siddiqui, H., Khendek, F., Toeroe, M., 2023. Microservices based architectures for IoT systems-State-of-the-art review. *Internet Things*, 100854.
- Struye, M.H., Mortier, H.J., Arnold, G., Miniggio, C., Borneck, R., 1994. Microprocessor-controlled monitoring of honeybee flight activity at the hive entrance. *Apidologie* 25 (4), 384–395. <https://doi.org/10.1051/apido:19940405>.
- Swain, M., Zimon, D., Singh, R., Hashmi, M.F., Rashid, M., Hakak, S., 2021. LoRa-LBO: an experimental analysis of LoRa link budget optimization in custom build IoT test bed for agriculture 4.0. *Agronomy* 11 (5), 820.
- Symeonaki, E., Arvanitis, K.G., Piromalis, D., Tseles, D., Balafoutis, A.T., 2022. Ontology-based IoT middleware approach for smart livestock farming toward agriculture 4.0: a case study for controlling thermal environment in a pig facility. *Agronomy* 12 (3). <https://doi.org/10.3390/agronomy12030750>.
- Szczurek, A., Maciejewska, M., Batog, P., 2023. Monitoring system enhancing the potential of urban beekeeping. *Appl. Sci.* 13 (1) <https://doi.org/10.3390/app13010597>.
- Tamburis, O., Giannino, F., D'Arco, M., Tocchi, A., Esposito, C., Di Fiore, G., Piscopo, N., Esposito, L., 2020. A night at the OPERA: a conceptual framework for an integrated distributed sensor network-based system to figure out safety protocols for animals under risk of fire. *Sensors* 20 (9). <https://doi.org/10.3390/s20092538>.
- Tashakkori, R., Hamza, A.S., Crawford, M.B., 2021. Beemon: an IoT-based beehive monitoring system. *Comput. Electron. Agric.* 190, 106427 <https://doi.org/10.1016/j.compag.2021.106427>.
- Ullah, A., Tlak Gajger, I., Majoros, A., Dar, S.A., Khan, S., Kalimullah, Haleem Shah, A., Nasir Khair, M., Hussain, R., Khan, H.U., Hameed, M., Anjum, S.I., 2021. Viral impacts on honey bee populations: a review. *Saudi J. Biol. Sci.* 28 (1), 523–530. <https://doi.org/10.1016/j.sjbs.2020.10.037>.
- Uthoff, C., Homs, M.N., von Bergen, M., 2023. Acoustic and vibration monitoring of honeybee colonies for beekeeping-relevant aspects of presence of queen bee and swarming. *Comput. Electron. Agric.* 205, 107589.
- Voudiotis, G., Kontogiannis, S., Pikridas, C., 2021. Proposed smart monitoring system for the detection of bee swarming. *Inventions* 6 (4), 87.
- Wrynski, E., Stock, J., Paiz, C., & Blanc, D. (2023). A Hardware Platform for Wireless Beehive Monitoring [Bachelor Of Science In Computer Science And Engineering, Santa Clara University]. https://scholarcommons.scu.edu/cseng_senior/245/.
- Wu, H., Li, J., 2023. Risk preference, interlinked credit and insurance contract and agricultural innovative technology adoption. *J. Innov. Knowl.* 8 (1), 100282.
- Xu, J., Gu, B., Tian, G., 2022. Review of agricultural IoT technology. *Artif. Intell. Agric.* 6, 10–22.
- Zabasta, A., Kunicina, N., Kondratjevs, K., & Ribickis, L. (2019). IoT Approach Application for Development of Autonomous Beekeeping System. 2019 International Conference in Engineering Applications (ICEA), 1–6. <https://doi.org/10.1109/CEAP.2019.8883460>.
- Zacepins, A., Brusbardis, V., Meitalovs, J., Stalidzans, E., 2015. Challenges in the development of precision beekeeping. *Biosyst. Eng.* 130, 60–71. <https://doi.org/10.1016/j.biosystemseng.2014.12.001>.
- Zacepins, A., Kviesis, A., Stalidzans, E., Liepniece, M., Meitalovs, J., 2016. Remote detection of the swarming of honey bee colonies by single-point temperature monitoring. *Biosyst. Eng.* 148, 76–80. <https://doi.org/10.1016/j.biosystemseng.2016.05.012>.
- Zacepins, A., Kviesis, A., Komasilovs, V., Rido Muhammad, F., 2020. Monitoring system for remote bee colony state detection. *Balt. J. Mod. Comput.* 8 (3) <https://doi.org/10.22364/bjmc.2020.8.3.05>.
- Zaman, A., Dorin, A., 2023. A framework for better sensor-based beehive health monitoring. *Comput. Electron. Agric.* 210, 107906.
- Zhai, Z., Martínez, J.F., Beltran, V., Martínez, N.L., 2020. Decision support systems for agriculture 4.0: survey and challenges. *Comput. Electron. Agric.* 170, 105256.