Design and development of smart beehive solution

Petar Lukovac
Department for e-business
University of Belgrade – Faculty of Organizational Sciences
Belgrade, Serbia
petar.lukovac@fon.bg.ac.rs
[0000-0003-4561-8886]

Hesam Jafarzadeh
Department for e-business
University of Belgrade – Faculty of Organizational Sciences
Belgrade, Serbia
hj20235089@student.fon.bg.ac.rs
[0009-0007-1812-5740]

Milica Simić
Department for e-business
University of Belgrade – Faculty of Organizational Sciences
Belgrade, Serbia
milica.simic@fon.bg.ac.rs
[0000-0002-6870-2303]

Radovan Marković
Department for e-business
University of Belgrade – Faculty of Organizational Sciences
Belgrade, Serbia
rm20210005@student.fon.bg.ac.rs

Svetlana Jovanović
Center for cyber forensics
University of Belgrade – Faculty of Organizational Sciences
Belgrade, Serbia
svetlana.jovanovic@fon.bg.ac.rs
[0000-0003-4561-8886]

***Abstract*—This paper focuses on the problem of efficiency in agriculture and its' environmental impact. The agricultural areas are increasing to follow the growing human demand and ensure food security. These actions have negative impact on natural habitats and biodiversity. Introduction of smart solutions in beekeeping can reduce this negative impact by improving efficiency of this agricultural field, while supporting the growth of pollinators population and preserving natural habitats. The paper presents architecture and a prototype of a smart beehive solution. The paper offers clear segmentation of layers in smart beehive architecture. This allows for clear understanding of the role of each part of the solution and quick error detection and handling. The paper showcases development of an edge device with all hardware components as well as key segments of implemented software. The smart beehive offers additional sensory data to beekeepers supporting data-based decisions and thus improving efficiency in beekeeping.**

Keywords—smart beehive, prototype development, IoT

I. Introduction

Pollination is an process vital for natural ecosystems and food security on land [1]. Pollinators and the environments where they live are experiencing increasing human impacts primarily leading to declines in species population [2]. There are many problems causing the decline in bee populations including land cover, climate change, disease, pesticides and pollutants [2]. Increasing deforestation and agricultural practices have major impact on biodiversity, species interactions and ecosystem functioning [1]. On the other hand, global demand for food resources is only increasing and requires expansion of agricultural areas and improvement in agricultural processes [3].

Digitalization and integration of information technologies and growing automation capabilities are drivers of precision agriculture [4]. Introduction of technologies such as Internet of Things (IoT), big data, artificial intelligence (AI), robotics, and blockchain technology greatly contributed to the development of smart agriculture extending the number of tasks devices can perform. This approach optimizes quantity and quality of crops thus promoting adoption and reducing environmental impacts [4], [5], [6]. Smart agriculture can increase efficiency of agricultural processes and thus reduce negative impacts in natural habitats.

The role of pollinators, especially bees, has critical influence on natural ecosystem and biodiversity [7]. Increased bee mortality and reduced colonies present challenges in its self. They demand new solutions for improving health of bee colonies and their numbers. Smart devices can offer solutions for improved monitoring and management of apiaries, by recording environmental data, internal hive data and other data that has impact on bee health [8], [9].

Objective of this paper is to present a functional prototype for a smart beehive device, focusing on sensor analysis and data collection.

The following section analyses existing smart solutions in apiculture, factors they focus on, sensors used in the process and key differences among them. Next, the paper focuses on smart apiculture architecture including all hardware and software components. Section 4 shows the developed solution including hardware components and key software segments. The paper then offers insight into collected data through visualizations on Grafana platform. Lastly, in conclusion section, main findings are summarized along with future direction on development of the smart beehive solution.

II. Related work

Traditional beekeeping methods rely on manual data collection making it labor and time intensive. Sensors and smart devices can monitor changes in environment, along with bees' health and activity. Most existing smart beehive solutions focus on collecting environmental data such as humidity, weight, and temperature, highlighting the importance of these parameters for assessing the basic condition of bee colonies. Some systems improve this approach by capturing in-hive acoustics to analyze bee behavior, while others use imaging or radar to count the number of bees entering and leaving the hive [10], [11], [12].

Many solutions are focused on data collection and visualization, typically presented through web platforms or mobile applications. Only a small number offer advanced features such as using artificial intelligence for tasks such as pest detection, issuing warnings, and classifying bees [10], [13]. These capabilities enhance colony management and enable faster, data-driven decision-making.

Connectivity also differs across platforms, with systems relying on technologies such as Wi-Fi, mobile networks, Bluetooth, or satellite connections. For instance, using a satellite hub to transmit data several times daily can be a valuable feature in regions with limited infrastructure.

Through utilization of modern communication protocols and real-time visualization tools, smart beekeeping enables continuous monitoring, timely interventions, and substantially enhances traditional beekeeping practices.

Despite these advancements, the majority of available devices remain focused on simple monitoring without incorporating predictive models or more sophisticated analytics. This gap presents clear opportunities for improvement particularly in prediction and in-depth data analysis which could significantly enhance the effectiveness and efficiency of modern beekeeping.

III. Smart Apiculture Architecture

Smart apiculture architecture follows specific pipeline for collecting, transporting, recording and visualizing data. A detailed architecture for smart beehive solution allows us to better understand the place and role of each component including their internal communication. The proposed system is presented in Figure 1.

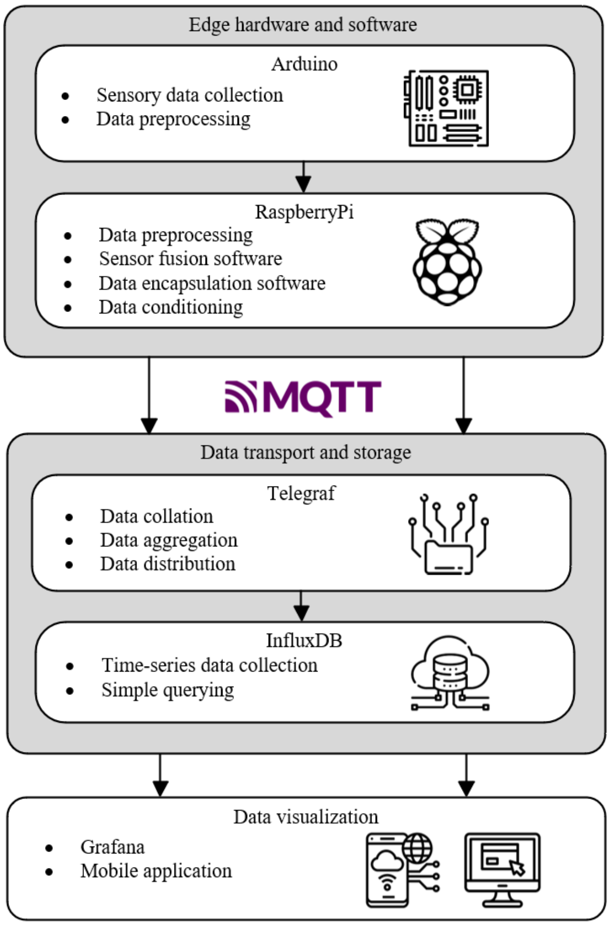

Fig 1. Smart apiary model

The architecture is divided into three main segments

- Edge hardware and software
- Data transport and storage
- Data visualization

The proposed pipeline enables scalability, flexibility and resilience. If there is a need for more sensors, data only needs to be published using MQTT, while Telegraf can automate rest of the process [14]. The proposed system publishes collected data enabling development of additional services without the need for any changes in current setup. New applications or AI models only need to subscribe to topics of interest. If there are problems in the system, for example if Grafana was down, data will still be recorded in the database. Additionally, if InfluxDB database is down, MQTT will buffer messages until Telegraf reconnects and flag them or send data upon connection.

Dividing specific tasks to separate parts of the system enables clear understanding of tasks where sensors are used to measure environmental conditions, MQTT transports collected data, Telegraf processes published data and automates communication with database. InfluxDB stores collected data and Grafana visualizes it through personalized dashboard. This approach enhances identification of possible errors and speeds up their corrections. The user can then follow the data in real time and get notified if any of parameters are out of defined range.

IV. Design and Implementation of a Prototype

Edge hardware and software include Arduino, Raspberry Pi connected to sensors such as DHT11, MQ135, presented in Figure 2. They measure temperature, humidity, air quality and other parameters.

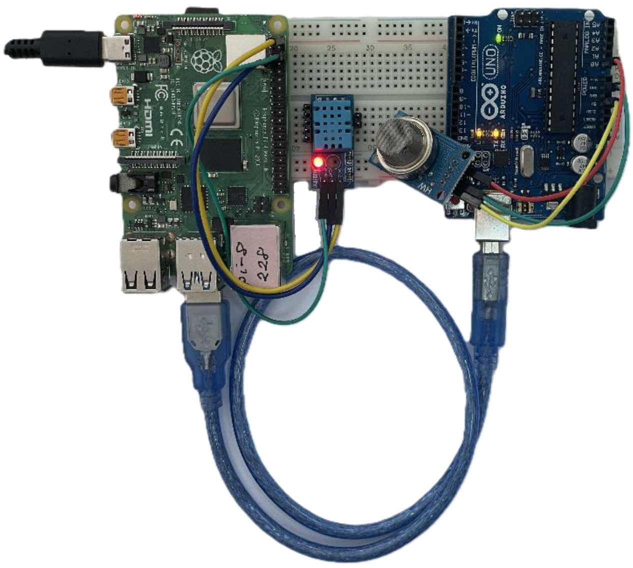

Fig 2. System prototype - edge devices

Instead of directly storing the data, the device publishes values to MQTT topics.

```
temp, hum = read_dht()
if temp is not None and hum is not None:
  publish.single(topic=topic_temp, payload=str(temp),
hostname=mqtt_host)
```

```
  publish.single(topic=topic_hum, payload=str(hum),
hostname=mqtt_host)
  print(f"Published -> {topic_temp}: {temp}°C,
{topic_hum}: {hum}%")
else:
  print("No valid data, skipping...")
# MQ-135 readings
mq_data = mq135.read()
if mq_data:
  raw_value, voltage = mq_data
  publish.single(topic=topic_mq135_raw,
payload=str(raw_value), hostname=mqtt_host)
  publish.single(topic=topic_mq135_voltage,
payload=str(voltage), hostname=mqtt_host)
  print(f"Published -> {topic_mq135_raw}: {raw_value},
{topic_mq135_voltage}: {voltage:.2f} V")
else:
  print("No valid MQ-135 data, skipping...")
  time.sleep(2)
```

MQTT Broker is used as part of the message transport layer. MQTT protocol is chosen for its' lightweight publish/subscribe communication. The broker, in this case Mosquitto, is there to collect all sensor data from edge devices. Each type of data has a topic:

- sensors/hive1/temperature
- sensors/hive1/humidity
- sensors/hive1/air_quality

By using publish/subscribe protocol, we enable multiple subscribers to use the same data, allowing simple expansion of the system. It can be used directly to trigger alerts, visualize data or store it in a database for later use.

In our case, Telegraf is used as a subscriber to listen to these topics. Telegraf is an agent that subscribes to MQTT topics and transforms data into a format that databases understand. It connects to MQTT, reads messages, and then writes them into InfluxDB. The advantages of using Telegraf are its' quick configuration through telegraf.conf file, elimination of the need to write custom scripts to save data to a database, and automatic reconnection to input and output plugins, in our case MQTT consumer and InfluxDB [14].

Using InfluxDB is a well-established choice for time-series database, optimized for time-series data generated by smart systems [15]. Example InfluxDB record:

```
environment,hive=hive1 temperature=28.5, humidity=62.0,
air_quality=110.0 1693658917000000000
```

Stores sensor values with:

- Measurement name: environment
- Fields: temperature, humidity, air_quality
- Tags including metadata: hive=hive1
- Timestamp: when it was measured

Benefits of using InfluxDB include easy querying through Flux query language [14].

All data stored in database is passed to Grafana which makes up visualization layer. It is used to create real-time dashboards, review data history or automate alerts in case values are out of range. Grafana queries InfluxDB using InfluxQL query language. Example query for temperature trend:

```
SELECT mean("value")
FROM "mqtt_consumer"
WHERE "topic" = 'sensors/hive1/temperature'
AND $timeFilter
GROUP BY time(5m) fill(null)
```

Telegraf, InfluxDB and Grafana form a stack known as TIG stack of open-source tools for monitoring and visualizing time-series data [16].

V. Data Collection and Visualization

The prototype presented in the previous section was tested in laboratory conditions. Samples were collected every two seconds, ensuring enough data to catch every anomaly that might happen. Sensors such as MQ135 were connected to Arduino to enable analog reading. Data is then transferred to Raspberry Pi, where MQTT publisher is implemented.

Preliminary results are presented through Grafana platform, Sample graphs showcase data collected from sensors.

Fig 3. Grafana graph for air quality

The graph on Figure 3. shows data from MQ135 sensor. It collects data about gases including CO_2, ammonia, smoke, nitrogen oxides etc. The green line presents raw sensor data ranging from 83 to 97. These values are considered low in the 0-1023 range, meaning the air around the hive is relatively clean. Measured values can be transferred to voltage using simple formula:

$$V_{out} = Raw\ value\ x\ V_{ref} / 1023 \qquad (1)$$

Where V_{ref} in (1) is circuit voltage of 5V for Arduino Uno used in the prototype and V_{out} is the result. This value can then be used to calculate the gases concentration in ppm and overall air quality. Since MQ135 sensor responds to multiple gases, we cannot be sure which gas it is responding to. Additional sensors are needed to be able to triangulate specific gases ant their concentration.

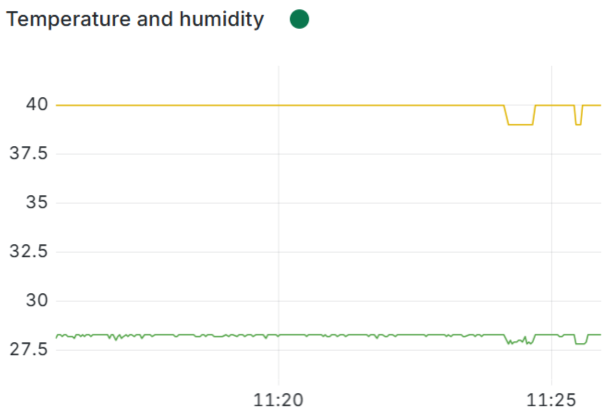

Fig 4. Grafana graph for temperature and humidity

The graph shows temperature and humidity around beehive. The green line presents a temperature of approximately 28 degrees which is within normal range. During the time period, temperature is very stable with small fluctuations. The humidity is presented by yellow line and is around 40% with slight drops that may indicate small environmental changes.

VI. Conclusion

This paper explains the need for smart apiary solutions, showing that introduction of smart devices can bring substantial benefits in apiculture. Improving efficiency is essential for long-term pollinator health and food security. Smart apiaries have the potential to redefine beekeeping by introducing data-driven decision-making processes.

This paper presented the development of a prototype smart apiary solution. The chosen technologies provide clear approach to data collection, transfer, storage and visualization. Dividing system into clear parts enables fast error detection and correction for more reliable solution.

The developed solution provides promising decision-support tool for beekeepers. It allows for more efficient processes, reducing manual work that is needed. The user can then follow the data in real time and get notified if any of parameters are out of defined range.

The developed solution demonstrates how relatively simple hardware can provide valuable insights into both internal conditions of the hive and environmental changes.

A. Future work

Addition of additional sensors such as weight sensor can offer early indicators of activities inside the bee colony, its health and honey production. Additional air quality sensors are needed to gather data about specific gases and their quantity instead of general air quality information.

The developed solution will be tested in the field, outside laboratory conditions. This can provide valuable information about the durability of sensors to weather conditions.

Setting the solution in real-world environment also includes testing different power solutions and comparison longevity and reliability of battery and solar power solutions.

References

[1] H. Liang, Y. D. He, P. Theodorou, and C. F. Yang, “The effects of urbanization on pollinators and pollination: A meta-analysis,” *Ecol Lett*, vol. 26, no. 9, pp. 1629–1642, Sep. 2023, doi: 10.1111/ELE.14277;WGROUP:STRING:PUBLICATION.

[2] G. LeBuhn and J. Vargas Luna, “Pollinator decline: what do we know about the drivers of solitary bee declines?,” *Curr Opin Insect Sci*, vol. 46, pp. 106–111, Aug. 2021, doi: 10.1016/J.COIS.2021.05.004.

[3] D. J. Langley, J. van Doorn, I. C. L. Ng, S. Stieglitz, A. Lazovik, and A. Boonstra, “The Internet of Everything: Smart things and their impact on business models,” *J Bus Res*, vol. 122, pp. 853–863, Jan. 2021, doi: 10.1016/J.JBUSRES.2019.12.035.

[4] F. Marinello *et al.*, “The Path to Smart Farming: Innovations and Opportunities in Precision Agriculture,” *Agriculture 2023, Vol. 13, Page 1593*, vol. 13, no. 8, p. 1593, Aug. 2023, doi: 10.3390/AGRICULTURE13081593.

[5] Y. Liu, X. Ma, L. Shu, G. P. Hancke, and A. M. Abu-Mahfouz, “From Industry 4.0 to Agriculture 4.0: Current Status, Enabling Technologies, and Research Challenges,” *IEEE Trans Industr Inform*, vol. 17, no. 6, pp. 4322–4334, Jun. 2021, doi: 10.1109/TII.2020.3003910.

[6] B. Radenković, M. Despotović-Zrakić, Z. Bogdanović, D. Barać, A. Labus, and Ž. Bojović, *Internet inteligentnih uređaja*. Univerzitet u Beogradu, Fakultet organizacionih nauka, 2017.

[7] T. A. Shaikh, W. A. Mir, T. Rasool, and S. Sofi, “Machine Learning for Smart Agriculture and Precision Farming: Towards Making the Fields Talk,” *Archives of Computational Methods in Engineering 2022 29:7*, vol. 29, no. 7, pp. 4557–4597, Jul. 2022, doi: 10.1007/S11831-022-09761-4.

[8] T. Haaker, P. T. M. Ly, N. Nguyen-Thanh, and H. T. H. Nguyen, “Business model innovation through the application of the Internet-of-Things: A comparative analysis,” *J Bus Res*, vol. 126, pp. 126–136, Mar. 2021, doi: 10.1016/J.JBUSRES.2020.12.034.

[9] S. Gierej, “The Framework of Business Model in the Context of Industrial Internet of Things,” *Procedia Eng*, vol. 182, pp. 206–212, Jan. 2017, doi: 10.1016/J.PROENG.2017.03.166.

[10] “AmoHive – Smart Hives, Network and IoT Solutions.” Accessed: Dec. 20, 2024. [Online]. Available: https://amohive.com/

[11] “BroodMinder - Every Hive Counts.” Accessed: Dec. 18, 2024. [Online]. Available: https://broodminder.com/

[12] A. Zaman and A. Dorin, “A framework for better sensor-based beehive health monitoring,” *Comput Electron Agric*, vol. 210, p. 107906, Jul. 2023, doi: 10.1016/J.COMPAG.2023.107906.

[13] “BeeMate .” Accessed: Dec. 10, 2024. [Online]. Available: https://beemate.buzz/

[14] A. Skalkin and A. Romanov, “Automation Methods of Data Collection and Analysis for Managing Complex IT Infrastructures,” *Proceedings - 2025 International Russian Smart Industry Conference, SmartIndustryCon 2025*, pp. 263–268, 2025, doi: 10.1109/SMARTINDUSTRYCON65166.2025.10986101.

[15] P. Tripathi, M. H. Miraz, and S. Joshi, “Comparative Analysis of MongoDB and InfluxDB for Time Series Data Management in IoT Environments: A Study on Performance, Scalability, and Concurrency,” *Proceedings - 2023 International Conference on Computing, Networking, Telecommunications and Engineering Sciences Applications, CoNTESA 2023*, pp. 39–42, 2023, doi: 10.1109/CONTESA61248.2023.10384962.

[16] M. Chakraborty and A. P. Kundan, “TICK Stack,” *ENERGIE*, vol. 37, no. 6, pp. 133–186, 2021, doi: 10.1007/978-1-4842-6888-9_5.