INTELLIGENT LONELINESS MONITORING IN THE ELDERLY: A SERVERLESS ARCHITECTURE WITH REAL-TIME COMMUNICATION API

Data de aceite: 01/07/2024

Ainhoa Osa Sanchez
eVida Research Group, University of Deusto, Bilbao, Spain

Oscar Jossa-Bastidas
eVida Research Group, University of Deusto, Bilbao, Spain

Amaia Mendez-Zorrilla
eVida Research Group, University of Deusto, Bilbao, Spain

Ibon Oleagordia-Ruiz
eVida Research Group, University of Deusto, Bilbao, Spain

Begonya Garcia-Zapirain
eVida Research Group, University of Deusto, Bilbao, Spain

ABSTRACT: Loneliness and social isolation pose critical public health challenges, particularly among older adults grappling with factors such as living alone, loss of social connections, chronic illness, and hearing impairment. These conditions significantly elevate the risk of premature death from various causes, including dementia, heart disease, and stroke. To combat these issues, the integration of technological platforms and commercial monitoring devices is increasingly prevalent in healthcare and elderly care. In this study, we leverage serverless architectures due to their manifold benefits, such as simplified application deployment, enhanced security-performance combination, cost and time savings, scalability without constraints, and streamlining of internal processes for continuous improvement. Our objective is to design and develop a loneliness monitor serverless architecture aimed at obtaining real-time data from commercial activity wristbands through an Application Programming Interface (API). Leveraging the Amazon Web Services (AWS) platform, we employ the Fitbit Charge 5 bracelet for loneliness monitoring. Data acquisition and storage are conducted within AWS services using the web API provided by the AWS Lambda service, with automated frequency facilitated by the event bridge. Additionally, a serverless IoT device is integrated to augment the architecture's effectiveness where applicable. In the pilot phase, the system demonstrates promising capabilities in simplifying data collection and programming sampling frequencies. Upon request, the collected data is automatically analyzed to monitor loneliness. The

proposed architecture holds significant potential for streamlined data collection, analysis, security, personalization, real-time inference, and scalability of sensors and actuators. It offers valuable benefits for application in the healthcare sector, mitigating cases of depression and loneliness among vulnerable populations.
KEYWORDS: Loneliness, Serverless, AWS, wearables, Monitoring, IoT

INTRODUCTION

Loneliness is characterized by negative subjective feelings stemming from low-quality or lack of social connections [1]. The elderly are particularly vulnerable to loneliness and social isolation due to factors such as living alone, bereavement, chronic illness, and sensory impairments like hearing loss [2]. Recognizing the severity of these issues, the Centers for Disease Control and Prevention identifies loneliness and social isolation among the elderly as significant public health concerns [3]. Studies on prevalence reveal that in certain European countries, between 18.7% and 24.2% of middle-aged individuals experience loneliness [1]. In the United States, approximately 33% of middle-aged adults report feelings of loneliness, while 25% of those aged 65 and above are socially isolated [3]. Importantly, loneliness heightens the risk of premature mortality from various causes, including dementia, heart disease, and stroke [4].

It is mostly believed that loneliness only affects older people as they age but is experienced across all age groups as O'Sullivan et all conclude by reviewing the systematic review and meta-analysis of 113 countries during the covid 19 pandemic [5]. Variations in loneliness by age and region require further exploration and data collection because not everyone is at the same risk of feeling lonely. The state of loneliness is affected by situations of poverty, poor physical or mental health, few community connections, and living alone. These increase the risk of loneliness. A better understanding of the situation and impact of the loneliness experience is required. Also, in different cultures and geographical variations.

It is important to see the increase in loneliness in the general and older population since 1970. As indicated by Crowe et al. [6]. The incidence has increased from 11% to 50% in people over 80 years of age. They suggest a multisystem approach strategy that can be developed from the health and care systems at a global level.

Loneliness and the sense of community are intricately linked components of social well-being. Loneliness often stems from a dearth of meaningful connections and a pervasive sense of isolation, while a robust sense of community can serve as a protective shield against its effects by offering support, fostering belonging, and facilitating social interaction [7]. Yet, it's essential to acknowledge that not all forms of social engagement can effectively address loneliness, particularly when individuals face challenges such as societal rejection or feelings of alienation. Nevertheless, nurturing inclusive and supportive communities remains paramount in the battle against loneliness and the promotion of holistic mental and emotional health. These communities provide invaluable spaces where individuals feel embraced, valued, and deeply connected.

The use of innovative technological tools in homes and nursing homes is giving positive results. As Waycott et al. [8] analyzed with 11 interviews immersive virtual reality (VR) is being used as an enriching experience for people living in care homes, with healthcare staff playing a key role in helping clients use virtual reality. It is an opportunity where users have fun and allows them to travel virtually where it generates opportunities to make memories while exercising among all residents. Although they conclude that the possibility of customizing the technology to be used for each user is important.

Passive sensing and Wearables-Based Solutions have been used to monitor and manage loneliness. For instance, Site et al., [9] studied a combination of the domains of gerontology, social psychology, architecture, and portable wireless technology to monitor and attempt to predict loneliness. Elmer et al. [10] use passive sensing collecting data from smartphone data to analyze time-stamped sensor data of social interactions. With that, they generate multistate survival models.

This study's primary contribution lies in the creation and implementation of a portable smart real-time loneliness monitor. A pivotal aspect to highlight is the utilization of cutting-edge tool technologies to integrate these systems seamlessly into homes and residences. Cloud computing, particularly through the adoption of serverless architectures, is emerging as a key enabler for such innovations. These architectures, operating under the concept of Function as a Service (FaaS) and Backend as a Service (BaaS), facilitate the development and execution of applications with unprecedented efficiency [11].

Serverless architectures have demonstrated versatility in various applications, including the establishment of a blockchain-enabled consortium architecture using Hyperledger Fabric. This implementation ensures security, integrity, transparency, and provenance in health-related transactions and the exchange of sensitive clinical information [12]. Additionally, serverless architectures have been instrumental in developing AI-powered smart healthcare frameworks aimed at reducing heart disease-related fatalities and minimizing financial losses by improving diagnostic accuracy [13]. The inherent advantages of serverless architectures, such as simplified application deployment without the need for dedicated servers, enhanced security-performance synergy, cost reduction, rapid development cycles, unlimited scalability, process optimization, and continuous improvement facilitation, further underscore their significance [14].

Ultimately, the primary aim of this study is to devise a customizable loneliness monitoring system capable of seamlessly integrating with Amazon Web Services (AWS). This system incorporates a bespoke IoT device and leverages a commercially available activity bracelet, specifically the Fitbit Charge 5 model, to achieve its objectives effectively.

RESOURCES, PROCEDURES AND PROPOSED SOLUTION

This section details the methods and materials utilized in crafting the proposed monitoring architecture, encompassing IoT sensors and embedded systems. For hardware, the Fitbit Charge 5 bracelet was chosen, complemented by IoT sensors and an embedded system to enhance functionality. In terms of software development and database management, cloud computing through AWS was leveraged. Fig. 1 showcases the design of the serverless architecture implemented using AWS services, integrating IoT sensors and embedded systems seamlessly.

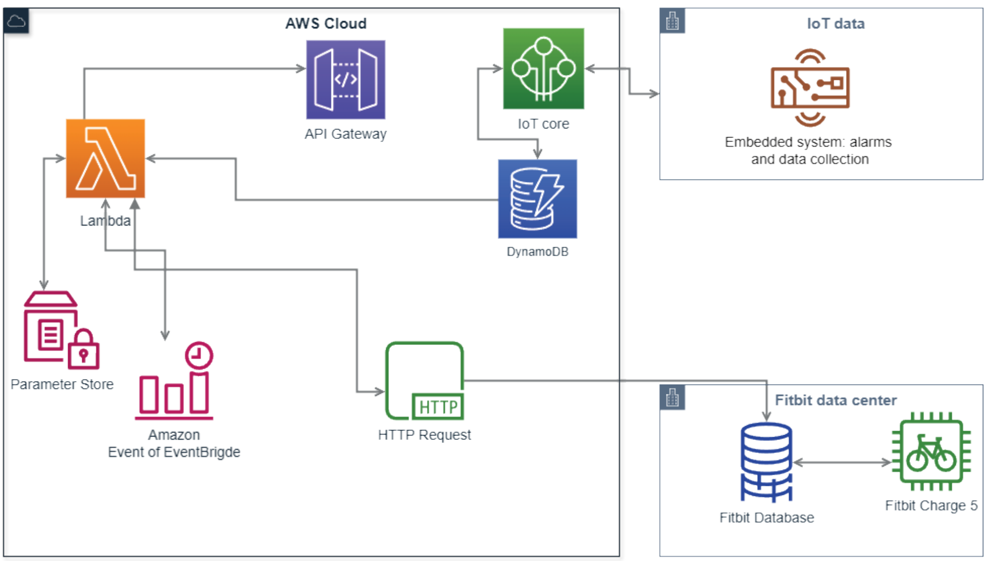

Fig. 1 AWS serverless architecture

Hardware description

Wearable devices

Wearable consumer activity trackers have gained significant popularity in recent years. These devices offer continuous monitoring of physical activity, heart rate, and sleep patterns [15].

The Fitbit Charge 5 is a commercial activity wristband designed to collect biometric data, including 24-hour heart rate monitoring and active zone minutes, while also providing stress management capabilities through its EDA sensor. Additionally, it tracks sleep patterns, oxygen saturation levels, physical activity, and calorie consumption [16].

Biometric data collected by Fitbit is automatically stored on its remote server via the companion app. Accessing this data from sources other than the Fitbit app requires registering a new app to make Web API calls. Fitbit Web APIs facilitate the retrieval of this data. However, specific authorization credentials, such as access tokens and user IDs obtained after registering a new app on the Fitbit developer website, are necessary. Validating and obtaining access tokens entails following an authorization process compliant with the OAuth 2.0 standard. Through this process, users can select the scopes to which they authorize access. The OAuth application generates client IDs and secrets for accessing tokens. Each access token remains valid for 8 hours by default before requiring renewal, although this duration can be adjusted during token generation [17], [18].

For the purpose of monitoring loneliness, specific requests have been identified:

- Get Daily Activity Summary: This request returns a summary and list of user activities and log and activity entries for a given day. In the list can be found the distance and steps taken, the calories burned, and the type of activity performed [19].
- Get Heart Rate Series by Date: This request returns a time series of the heart rate during a specific period. In this series can find the maximum frequency, the minimum frequency, the resting heart rate and the different calorie-burning zones.
- Get Sleep Log by Date: This request returns a list of the user's recorded sleep logs for a given date. In the list can find the sleep efficiency, the duration even the details of the sleep level.

Embedded system

The embedded system prototype utilizes two key microcontrollers:

- Raspberry Pi 4 Model B: Developed by the Raspberry Pi Foundation, this single-board computer boasts 4GB of RAM, USB-C power supply compatibility, Micro HDMI ports, USB interfaces, and Gigabit Ethernet connectivity.
- ESP32 Development Kit: Manufactured by Espressif Systems, the ESP32 microcontroller offers dual-core Xtensa LX6 processors, integrated Wi-Fi and Bluetooth connectivity, GPIO pins for interfacing with external devices, and support for various communication protocols. It serves as a robust foundation for the prototype, enabling seamless integration with other devices and communication with the Raspberry Pi via serial communication, enhancing the system's functionality and versatility.

In our endeavor to monitor loneliness, the acquisition of precise and varied data was essential. To achieve this, a selection of sensors was meticulously chosen based on their specific functionalities. Among these sensors, the CCS811 by Digilent emerged as a vital component for assessing indoor air quality, crucial for understanding the environmental context in which feelings of loneliness may manifest. Renowned for its accuracy, the CCS811 sensor measures volatile organic compounds (VOCs) and equivalent CO2 levels, providing invaluable insights into air quality conditions within the household environment. Additionally, the DHT11 sensor by Adafruit played a pivotal role in our study, enabling the monitoring of ambient temperature and humidity levels, factors that can significantly impact an individual's comfort and well-being in their home environment.

Furthermore, motion detection was facilitated by the PIR Sensor EKMB1305111K by Panasonic, a highly sensitive sensor capable of detecting infrared radiation emitted by moving objects. This sensor proved indispensable in identifying physical activity patterns and potential interactions within the household, offering insights into daily routines and social engagement levels. Lastly, the 3-axis Accelerometer ADXL345 by Adafruit was employed to measure acceleration, providing valuable data on movement patterns and activity levels. Its integration with other sensors allowed for a holistic understanding of the individual's physical movements and environmental context, aiding in the development of strategies to mitigate loneliness and enhance overall well-being.

Mechanic design

We devised a 3D enclosure to safeguard the sensor and accommodate additional specifications outlined later in the process, as can be seen in fig. 1. The mechanical blueprint was meticulously crafted using SolidWorks 2023 [20], a sophisticated CAD software. Once the design for the case and its accompanying components was finalized, we proceeded with prototype production utilizing additive manufacturing techniques. During the fabrication phase, we utilized the open-source slicing software CURA (V4.9.9), provided by Ultimaker, to prepare the design files for printing. The chosen printer was the Ultimaker 3 Extended [21], renowned for its precision and reliability in producing intricate designs.

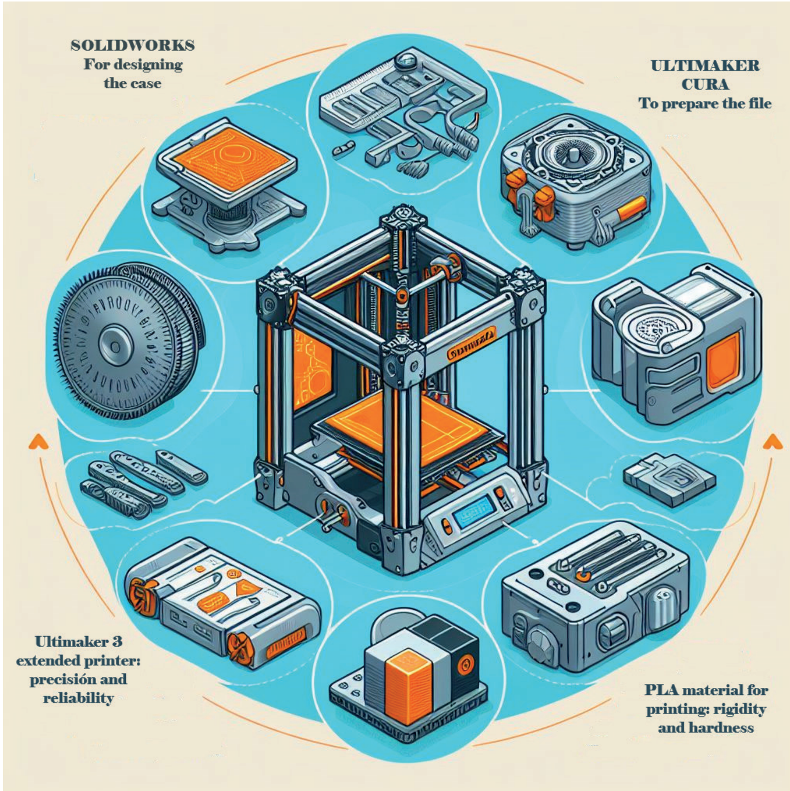

Fig. 1 Printed box

In our printing process, we employed both Polylactic Acid Filament (PLA) and ABS materials for fabrication. PLA was specifically selected for its superior balance of rigidity and hardness, as illustrated in Fig. 2. This careful choice guarantees robust protection for the sensor housed within the enclosure.

Fig. 2 Printed box

Experiment

To thoroughly validate the proposed architecture, a series of laboratory tests were conducted to ensure the seamless connection not only with the Fitbit database but also with the IoT device. The primary objective was to verify the accurate recording of data in the AWS DynamoDB table. The test cohort comprised three users, all aged 18 years or older. Utilizing data collected from laboratory tests conducted since January 2023, spanning approximately three months, the consistency of performance was evaluated, as illustrated in Fig 3.

Fig. 3 Graphic Representation of User Interaction with the System

Throughout the testing period, the same Fitbit bracelet was utilized across different users, enabling a comprehensive assessment of the architecture's functionality. Additionally, stringent validation of the connection with the IoT device was conducted to ensure the robustness and reliability of the entire system.

Serverless architecture

In this study, we used the services provided by the general-purpose IoT platform AWS, taking advantage of the benefits of serverless architectures.

- Lambda: Lambda is a service designed to execute code in response to events without the need for server management. It furnishes a programming interface capable of processing input events from both internal and external sources [22].
- IoT Core: The IoT Core facilitates connectivity between various IoT devices and the cloud by managing their connections and implementing sufficient permissions to ensure their protection. Moreover, it employs TLS protocol encryption to secure communications and enables real-time message forwarding to other microservices [23]. Communication between AWS and the IoT device primarily utilizes the Message Queuing Telemetry Transport (MQTT) protocol by default, employing publication or subscription to different topics.

- API Gateway: The API Gateway service offers several advantages, including enhanced security, monitoring capabilities, publication support, and maintenance features for REST, WebSocket, and HTTP APIs, supporting PUT, POST, GET, PATCH, and DELETE methods. When combined with Lambda, users can develop APIs and seamlessly integrate them with other AWS services. Notably, the API Gateway facilitates the exposure of endpoints for accessing data, acting as an intermediary for other applications or clients [18].
- DynamoDB: DynamoDB is a fully managed NoSQL database service engineered to support high-performance applications at any scale. It provides robust security measures, backup functionalities, in-memory caching capabilities, and the ability to import or export data. Data stored in DynamoDB can be accessed, added to, and edited via API calls or Lambda functions [18].
- Amazon EventBridge: Serving as a facilitator for the development of event-driven applications, Amazon EventBridge establishes connections between events and various AWS services. It enables the embedding, filtering, transformation, and delivery of events, while also providing options for managing event buses [24].
- AWS Systems Manager: Functioning as a secure operations center for managing applications and resources, AWS Systems Manager offers a range of functionalities. Notably, the Parameter Store feature proves particularly useful, providing a secure and hierarchical storage solution for managing configuration data and secrets such as passwords and database strings. Access to the store is facilitated through scripts [25].

RESULTS

Fitbit Serverless architecture description

Once the access tokens and renewal of access to the data have been obtained through the Fitbit Dev application registration, two string-type parameters have been created in the systems Manager service parameter store. This way the tokens can be modified when they are updated and are protected with additional security.

To make requests to the Fitbit database, the AWS lambda function called "fitbit_connect" is called through an event, as shown in Fig 4. The function obtains the tokens stored in the store and calls the internal function "get_activity" by passing it the desired tokens, date, and activity. This function makes the requests following the examples that Fitbit proposes. If it detects an error in the request because the access token is expired, call the "get_tokens" function and then redo the request changing the old security access with the new tokens in the correspondence parameter store. Once the request has been made successfully, the data is filtered to obtain a JSON that allows it to be correctly saved in the database. To this JSON the data of the company, professional and user are added. Finally, the internal function "storing_data" is activated, which sends the JSON to the Dynamo DB database table.

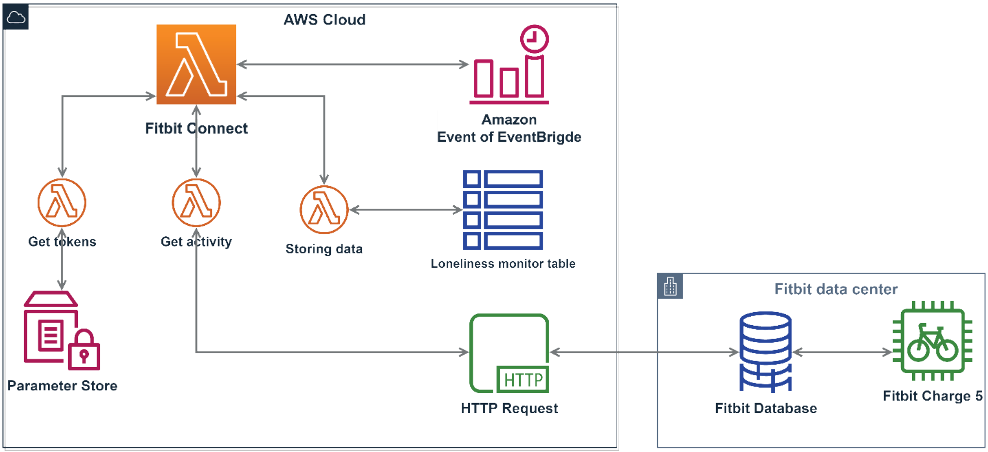

Fig. 4 “Fitbit_connect” lambda function functionality and connection between other services

Continuous data collection is facilitated through a structured process. To request data from the Fitbit server, we established a rule within the default event bus of Amazon EventBridge. This rule triggers the Lambda function activation periodically to check the evolution of data throughout the day. Specifically, the selected time interval for these activations is set at 15-minute intervals. This systematic approach ensures consistent and frequent updates, enabling real-time monitoring and analysis of user metrics.

IoT device serverless connection

Our IoT system hardware setup incorporated a Raspberry Pi embedded system alongside temperature, motion, and accelerometer sensors, as shown in fig. 5. Communication between components utilized the I2C protocol and digital I/O communication. Due to compatibility issues with the Raspberry Pi, the air quality sensor connected to the ESP32 via I2C. Each system included two LEDs as alarm indicators for internet connection status, with alarms to alert of risky CO2, temperature, or humidity levels. Powering the boards, we designated the Raspberry Pi 4 as the main source, with the ESP32 powered by its general-purpose pins at 5V. We established connections using jumper cables and configured communication with the Amazon Web Services platform using the MQTT protocol.

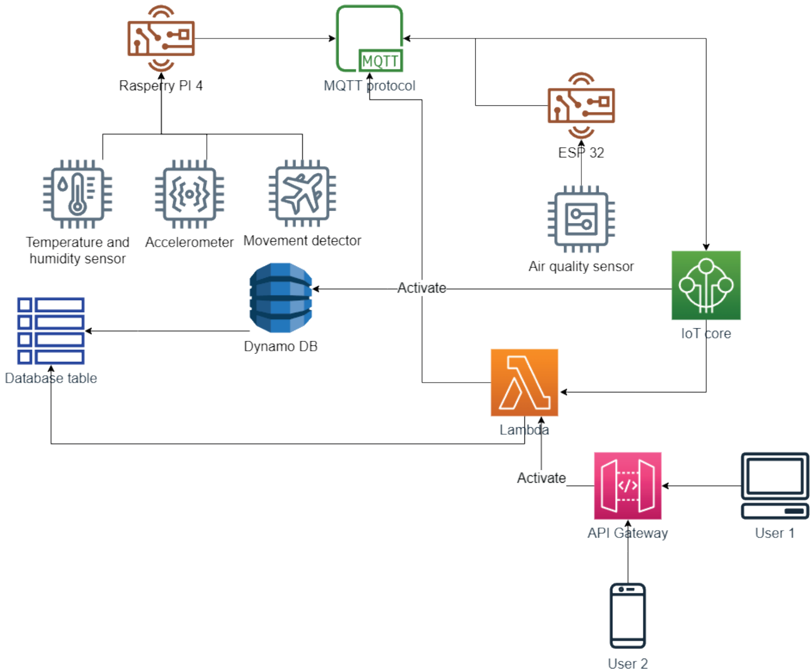

Fig. 5 serverless architecture of the IoT device.

1. **Embedded System Configuration**: The embedded system has been configured to be remotely controlled from Amazon's infrastructure using the IoT Core microservice. Within the IoT Core service, two distinct entities, often referred to as "things" or "devices," have been created. These entities are specifically generated to acquire the necessary permissions and security credentials required for establishing a secure connection between the embedded system and the AWS server.

2. **Data Routing and Processing**: The IoT Core service facilitates the creation of rules, allowing for the implementation of customized data handling procedures. One such rule has been programmed to route incoming data streams, typically transmitted via the MQTT protocol, to activate specific functionalities within the AWS ecosystem. In this case, the rule is configured to trigger the execution of a Lambda function upon receiving data from the embedded system. An example can be seen in Fig. 6. The Lambda function is responsible for processing the incoming data and executing predefined tasks. In this context, the Lambda function is programmed to store the variable data obtained from the embedded system into a designated table within the DynamoDB database.

lambda_function × Execution results ×

```python

from esp32.publishState import startEsp32
from raspberry.changeShadow import startRaspberry
#from fitbit.startFitbit import getDataFitbit
import json
import time
import boto3

def lambda_handler(event, context):

    try:
        username=event["username"]
        print(username)
        print (event["esp32"])
        dictRaspberry = event["raspberry"]
        print(dictRaspberry)
        dictEsp32 = event["esp32"]
        print(dictEsp32)
        print("start")
        startRaspberry(dictRaspberry,username)
        #startRaspberry(dictRaspberry)
        startEsp32(dictEsp32, username)

    except Exception as e:
        return {
            'statusCode': 200,
            'message': e
        }

    return {
        'statusCode': 200,
        'message': 'State has been successfully changed!'
    }

```

Fig. 6 Data route for starting data collection

3. **Data Collection and Transmission**: Once the data is collected by the embedded system, it initiates the process of transmitting the data to AWS. This is achieved by publishing a message on a specific topic within the MQTT protocol, with the data structured in a JSON format. Subsequently, the IoT Core microservice receives the message containing the data and redirects it to other AWS services. Specifically, the data is forwarded to DynamoDB, where it is stored in an Amazon DynamoDB table. Notably, each attribute of the payload is meticulously written into separate columns within the DynamoDB database, ensuring efficient organization and accessibility of the stored information.

Database table

We've crafted a DynamoDB database table, as outlined in Table 1, employing a combination of partition key and sort key. The parameters are arranged from left to right based on the data hierarchy, aiming to optimize query efficiency from Lambda functions. Considering the diverse companies involved in the project, the table accommodates five distinct types of data. These companies oversee professionals responsible for monitoring device users, necessitating a flexible and comprehensive data structure to support varied organizational needs.

Type	Partition Key	Sort Key
Company	COMP	COMP#<company_name>
Professional	PROF#<professional_name>	COMP#<company_name>
User	USER#<user_name>	PROF#<professional_name>
Device	SENSOR#<sensor_type>	USER#<user_name>
Measurement	USER#<user_name>	Sensor_type#<timestamp>

Table 1. Partition and sort keys for each type of data for the DynamoDB Table

Database requests are seamlessly executed through Lambda functions, enabling efficient data analysis and visualization on the web. Within Lambda, we implemented read, upload, and delete operations to interact with the database. This function is designed to receive information on the action and variables required to execute the requested operation, fig.6 shown a graphic summary.

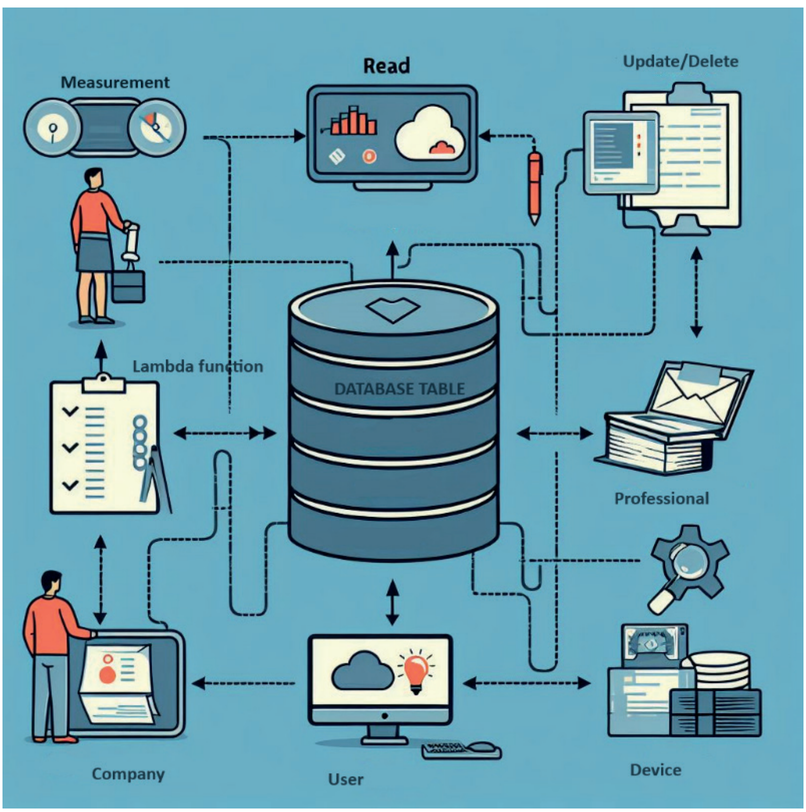

Fig. 6 Data route for starting data collection

Below, we summarize the implemented actions of the Lambda function:

- User actions: create and delete individual users for a specific professional, also read all the existing users or users for a professional.
- Company actions: create and delete individual companies, and also read all the existing companies.
- Professional actions: create and delete individual professionals for a specifically created company, also read all the existing professionals or professionals for a company.
- Device actions: create and delete individual devices for a specific user, also read all the existing devices or devices for a user.
- Measurement actions: save, and delete a measurement requested to Fitbit and also read all the existing measurements for a user.

Measurement experiments

The proposed architecture underwent rigorous testing through a laboratory experiment involving the use of a Fitbit bracelet. In this experiment, three distinct test users were registered within the mobile application. Each user wore the bracelet continuously for a week before handing it over to the next user, allowing for comprehensive data collection across varying usage scenarios. To ensure data integrity and user privacy, the bracelet was reset for each user's turn, thereby associating the device with the corresponding user account in the Fitbit application.

For each user, two variables were generated within the system to store the access tokens required for accessing their individual user accounts. Following the completion of the testing period, it was verified that the collected data was successfully stored in Amazon DynamoDB, as depicted in Fig. 6. Additionally, the system was designed to validate the user accounts stored in the system. However, data transmission to the database was contingent upon the presence of registered activity. If the system detected no registered activity during a specific time interval, data transmission to the database was withheld, ensuring that only relevant data was recorded.

Moreover, it's important to note that each test participant also had access to a prototype of the IoT device, which they used concurrently with the Fitbit bracelet. This arrangement allowed for the simultaneous collection of data from both the Fitbit bracelet and the IoT device prototype, providing a comprehensive view of the users' activities and interactions within their environment.

PK	SK	heartRateVariability	restingHeartRate	timeSleep
USER#ainhoa	fitbit#1664449565.696445	154	91	19860000
USER#ainhoa	fitbit#1664449556.734373	154	91	19860000
USER#ainhoa	fitbit#1664449547.714125	154	91	19860000
USER#ainhoa	fitbit#1664449538.672716	154	91	19860000
USER#ainhoa	fitbit#1664449529.034131	154	91	19860000

Fig. 6 Example with part of the data collected and saved to the DynamoDB table following the partition key and sort key combination.

RESULTS ANALYSIS

The developed monitoring system showed great potential and is an attractive approach for real-time monitoring in healthcare. Regarding the use of wearable consumer activity trackers, it has the advantages of security and privacy communication with the user using mobile phone authentication sessions in the communication app. As Perumal et al. in [26] conclude that the communication with wearable devices with the encryption of mobile phones provide secure and private communications. Also has applications in detecting more possible medical episodes than conventional monitoring [27].

As for the serverless architecture proposed, it has the advantages of scaling and instant deployment of services which allows the increase of users using the monitor which could help increase the use of possible monitors in the health system [28]. Also, the communication system using API complies with the requirements of cross-border transactions involving massive IoT application data and considers the privacy of users [28], [29]. Storing the data in the DynamoDB database allows cost-efficient and secure IoT systems supported by AWS security systems for health data taking into account that third-party auditors regularly test and verify the effectiveness of AWS security [30].

When exploring the utility of IoT sensors, like those monitoring air quality, it's crucial to recognize their broader significance. Loneliness and indoor air quality intertwine as critical factors shaping individuals' well-being, especially apparent during extended indoor periods like the COVID-19 pandemic[31]. Research highlights that loneliness correlates with negative mental health impacts such as stress and depression during lockdowns, while poor indoor air quality exacerbates discomfort and health issues [32]. Addressing both concerns is essential for holistic well-being, necessitating measures like social support, fostering connections, and ensuring indoor air quality through ventilation. Understanding the nexus between loneliness and indoor air quality underscores the importance of integrating social and environmental considerations to enhance overall well-being, particularly in prolonged indoor confinement.

This design also has limitations, this are the data collection capacity and the precision of the same of the selected commercial bracelet. It must also take into account the rigorous control that should be in the residences. A bracelet change could occur and not be notified, which would result in the data being misclassified.

This research showed the potential benefits that a serverless solution coupled with the use of sensors could provide in health care and third age.

CONCLUSIONS

In conclusion, this chapter proposes a novel system that combines wearable devices with Serverless Computing technology architecture to monitor loneliness. The developed architecture exhibits significant potential for streamlined data collection, analysis, security, personalization, real-time inference, and scalability of sensors and actuators. Its application in the healthcare sector holds promise for mitigating cases of depression and loneliness among vulnerable populations.

However, certain limitations exist, including the challenge of conducting extensive tests with real users in residences and homes for the elderly. Additionally, the method of extracting data from Fitbit may result in accumulated rather than detailed data throughout the day.

Future research endeavors aim to address these limitations by collecting more extensive data through the deployment of the designed system in residences, involving caregivers and psychologists. Statistical analysis and data cleaning techniques will be applied to the collected data to develop artificial intelligence models using AWS Sagemaker. These models will be programmed to detect potential loneliness alarms and generate warnings to professionals, enabling proactive intervention to mitigate the risks of loneliness in users.

REFERENCES

1. D. L. Surkalim *et al.*, “The prevalence of loneliness across 113 countries: systematic review and meta-analysis,” *BMJ*, vol. 376, Feb. 2022, doi: 10.1136/BMJ-2021-067068.

2. P. Qualter, M. Mund, L. Heu, L. Zhao, and L. Wu, “The Association between Social Participation and Loneliness of the Chinese Older Adults over Time—The Mediating Effect of Social Support,” *International Journal of Environmental Research and Public Health 2022, Vol. 19, Page 815*, vol. 19, no. 2, p. 815, Jan. 2022, doi: 10.3390/IJERPH19020815.

3. “Social Isolation and Loneliness in Older Adults,” *Social Isolation and Loneliness in Older Adults*, May 2022, doi: 10.17226/25663.

4. J. Holt-Lunstad and A. Steptoe, “Social isolation: An underappreciated determinant of physical health,” *Curr Opin Psychol*, vol. 43, pp. 232–237, Feb. 2022, doi: 10.1016/J.COPSYC.2021.07.012.

5. R. O'Sullivan, G. Leavey, and B. Lawlor, "We need a public health approach to loneliness," *BMJ*, vol. 376, Feb. 2022, doi: 10.1136/BMJ.O280.

6. C. L. Crowe, L. Liu, N. Bagnarol, L. P. Fried, and N. Bagnarol, "Loneliness prevention and the role of the Public Health system," *Perspect Public Health*, Jul. 2022, doi: 10.1177/17579139221106579/ASSET/IMAGES/LARGE/10.1177_17579139221106579-FIG2.JPEG.

7. M. A. Malli, S. Ryan, J. Maddison, and K. Kharicha, "Experiences and meaning of loneliness beyond age and group identity," *Sociol Health Illn*, vol. 45, no. 1, pp. 70–89, Jan. 2023, doi: 10.1111/1467-9566.13539.

8. J. Waycott, R. M. Kelly, S. Baker, B. Barbosa Neves, K. S. Thach, and R. Lederman, "The Role of Staff in Facilitating Immersive Virtual Reality for Enrichment in Aged Care: An Ethic of Care Perspective," *Conference on Human Factors in Computing Systems - Proceedings*, p. 17, Apr. 2022, doi: 10.1145/3491102.3501956.

9. A. Site *et al.*, "Managing Perceived Loneliness and Social-Isolation Levels for Older Adults: A Survey with Focus on Wearables-Based Solutions," *Sensors 2022, Vol. 22, Page 1108*, vol. 22, no. 3, p. 1108, Feb. 2022, doi: 10.3390/S22031108.

10. T. Elmer and G. Lodder, "Modeling social interaction dynamics measured with smartphone sensors: An ambulatory assessment study on social interactions and loneliness," *https://doi.org/10.1177/02654075221122069*, vol. 40, no. 2, pp. 654–669, Aug. 2022, doi: 10.1177/02654075221122069.

11. Y. Li, Y. Lin, Y. Wang, K. Ye, and C. Z. Xu, "Serverless Computing: State-of-the-Art, Challenges and Opportunities," *IEEE Trans Serv Comput*, Mar. 2022, doi: 10.1109/TSC.2022.3166553.

12. A. Ayub Khan, A. A. Wagan, A. A. Laghari, A. R. Gilal, I. A. Aziz, and B. A. Talpur, "BIoMT: A State-of-the-Art Consortium Serverless Network Architecture for Healthcare System Using Blockchain Smart Contracts," *IEEE Access*, vol. 10, pp. 78887–78898, 2022, doi: 10.1109/ACCESS.2022.3194195.

13. M. Golec, S. S. Gill, A. K. Parlikad, and S. Uhlig, "HealthFaaS: AI based Smart Healthcare System for Heart Patients using Serverless Computing," *IEEE Internet Things J*, 2023, doi: 10.1109/JIOT.2023.3277500.

14. A. O. Sanchez, U. G. Ugarte, M. J. G. Herrera, O. J.- Bastidas, and B. Garcia-Zapirain, "Design and implementation of food quality system using a Serverless Architecture: case study of gluten intolerance," *Telematique* , pp. 7475–7486, Dec. 2022.

15. D. Mayorga-Vega, C. Casado-Robles, I. López-Fernández, and J. Viciana, "Activity wristband-based physical activity recommendations in young people," *Sci Sports*, vol. 37, no. 4, pp. 303–315, Jun. 2022, doi: 10.1016/J.SCISPO.2021.05.007.

16. "Pulsera de actividad + salud avanzada | Compra una Fitbit Charge 5." Accessed: Jul. 03, 2023. [Online]. Available: https://www.fitbit.com/global/es/products/trackers/charge5

17. "Web API." Accessed: Jul. 03, 2023. [Online]. Available: https://dev.fitbit.com/build/reference/web-api/

18. S.-H. Sim, T. Paranjpe, N. Roberts, and M. Zhao, “Exploring Edge Machine Learning-based Stress Prediction using Wearable Devices,” *2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 1266–1273, Dec. 2022, doi: 10.1109/ICMLA55696.2022.00203.

19. “Get Daily Activity Summary.” Accessed: Jul. 03, 2023. [Online]. Available: https://dev.fitbit.com/build/reference/web-api/activity/get-daily-activity-summary/

20. M. Angélica, L. Solorio, Y. Damián, and C. Milian, “Transformando la ingeniería con SolidWorks,” *Milenaria, Ciencia y arte*, no. 22, pp. 26–29, Dec. 2023, doi: 10.35830/MCYA.VI22.415.

21. “Hardware designs for the Ultimaker 3 and Ultimaker 3 Extended released - UltiMaker.” Accessed: May 03, 2024. [Online]. Available: https://ultimaker.com/es/learn/hardware-designs-for-the-ultimaker-3-and-ultimaker-3-extended-released/

22. “AWS Lambda Documentation.” Accessed: Jul. 03, 2023. [Online]. Available: https://docs.aws.amazon.com/lambda/?icmpid=docs_homepage_featuredsvcs

23. P. Dubey and A. K. Tiwari, “Solution Architecting on Remote Medical Monitoring with AWS Cloud and IoT,” pp. 189–202, 2023, doi: 10.1007/978-981-99-0377-1_12.

24. V. S. V. M. Krishna, T. S. R. Vamsi, and S. Kavitha, “Automation of Forensic Analysis for AWS Aurora using EventBridge and Athena,” *6th International Conference on Electronics, Communication and Aerospace Technology, ICECA 2022 - Proceedings*, pp. 262–269, 2022, doi: 10.1109/ICECA55336.2022.10009126.

25. “AWS Systems Manager Documentation.” Accessed: Jul. 03, 2023. [Online]. Available: https://docs.aws.amazon.com/systems-manager/

26. P. Perumal and S. Subha, “An analysis of a secure communication for healthcare system using wearable devices based on elliptic curve cryptography,” *World Review of Science, Technology and Sustainable Development*, vol. 18, no. 1, pp. 51–58, 2022, doi: 10.1504/WRSTSD.2022.119327.

27. C. S. Webster, T. W. L. Scheeren, and Y. I. Wan, “Patient monitoring, wearable devices, and the healthcare information ecosystem,” *Br J Anaesth*, vol. 128, no. 5, pp. 756–758, May 2022, doi: 10.1016/J.BJA.2022.02.034.

28. M. Golec, S. S. Gill, A. K. Parlikad, and S. Uhlig, “HealthFaaS: AI based Smart Healthcare System for Heart Patients using Serverless Computing,” *IEEE Internet Things J*, 2023, doi: 10.1109/JIOT.2023.3277500.

29. R. Ouyang *et al.*, “A Microservice and Serverless Architecture for Secure IoT System,” *Sensors 2023, Vol. 23, Page 4868*, vol. 23, no. 10, p. 4868, May 2023, doi: 10.3390/S23104868.

30. A. Lakhan *et al.*, “Restricted Boltzmann Machine Assisted Secure Serverless Edge System for Internet of Medical Things,” *IEEE J Biomed Health Inform*, vol. 27, no. 2, pp. 673–683, Feb. 2023, doi: 10.1109/JBHI.2022.3178660.

31. J. Fernández-Agüera, S. Domínguez-Amarillo, M. Á. Campano, and H. Al-Khatri, “Effects of covid-induced lockdown on inhabitants’ perception of indoor air quality in naturally ventilated homes,” *Air Qual Atmos Health*, vol. 16, no. 2, pp. 193–212, Feb. 2023, doi: 10.1007/S11869-022-01239-3/FIGURES/6.

32. S. Rabipour, “Chemical Review and Letters Air chemical particles and pollution quality association with mental health: A case for depression and anxiety among Iranian university students in Iranian cities with lower air quality,” *Chem Rev Lett*, vol. 6, pp. 66–78, 2023.