Developing Web Services for the Production of Mixed Reality Games Based on RFID Sensors

Samuel Lima[1], **Victor Sarinho**[1]

[1] Laboratório de Entretenimento Digital Aplicado - LEnDA
Universidade Estadual de Feira de Santana - UEFS
Feira de Santana - Bahia - Brasil

samvlima10@gmail.com, vsarinho@uefs.br

Abstract. *The Internet of Things (IoT) is a technology that enabled the construction of accessible systems that allow the connection of several physical and virtual objects. This work presents the development of a collection of web services capable of assisting the production of mixed reality games based on RFID (Radio Frequency Identification) sensors. To this end, hardware devices, REST and IoT communication services are presented, as well as the production of educational games based on these proposed services for validation purposes.*

Keywords: *Mixed reality; IoT; Open hardware; RFID; REST; Educational games.*

1. Introduction

Mixed reality games are performed by digital and analog games that provide situations and problems in which the player must interact with the physical world towards the virtual world to achieve possible solutions [Bohn 2004]. This play style generally seeks to improve the player learning, as it aims to make the activity more didactic and dynamic, and encourage the interaction with other players [Miglino et al. 2014]. As a result, mixed reality games have become applicable to support the teaching and learning process [Herring et al. 2010], encouraging activities that involve interactions with groups of people [Hinske and Langheinrich 2007][Deshmukh and Baru 2013].

Mixed reality appears by the use of concepts and components from the Internet of Things (IoT) that allowed a low-cost link between physical objects and virtual objects in a single universe [Jia et al. 2012]. In this sense, as an attempt to provide a standardized integration of IoT with the mechanics and dynamics of educational board games, this work presents the implementation of hardware/software assets for the identification and integration of physical and virtual objects using RFID sensors and REST services. For this, a hardware device together with IoT communication services were developed, as well as examples of educational games produced by these assets for validation purposes, in order to assist the creation of games based on the mixed reality concept.

2. Related Work

Some initiatives have developed mixed reality game applications using RFID sensors. In the field of board games, for example, Deshmukh, Sonal and Baru created a hardware/software architecture based on mixed reality whose main objective was to help the chess player learn [Deshmukh and Baru 2013]. Another structure based on the use of

RFID technology was built to automate the identification of board game pieces in a war strategy game [Hinske and Langheinrich 2007].

As an example of an initiative in the United Kingdom aimed at supporting the autonomous learning of special children, teachers used an image exchange communication system to support the education of children with Autism Spectrum Disorder (ASD) [Herring et al. 2010]. Based on this system, a virtual teacher was developed to interact with the child through images that contained embedded RFID tags, which allowed the emergence of autonomous interactions during the learning process.

Regarding the definition of low-cost hardware/software architectures capable of supporting the production of mixed reality games that use RFID technology, Lima and Sarinho proposed the use of the MQTT protocol for communication between physical devices and responsible softwares. They were used to develop a classic Tic-Tac-Toe game based on RFID devices [Lima and Sarinho 2019], and a memory game with the objective of identifying feelings in represented faces [Lima and Sarinho 2020].

3. Methodology

The objective of this work is to provide hardware/software assets able to assist the communication process between digital game applications and physical devices based on RFID sensors. The idea is to provide a communication structure between RFID hardware devices embedded in real-world objects with the responsible software for the IoT communication and for the virtual processing of the desired game.

3.1. Hardware Architecture

Regarding the required RFID device, it is necessary to provide a hardware architecture able to detect the RFID tag information, and to send such information to the applications that have registered interest on it. In this sense, it is necessary to provide a hardware structure able to: establish a connection to a WiFi network; process the unique identification of RFID tags; and make requests to a communication API.

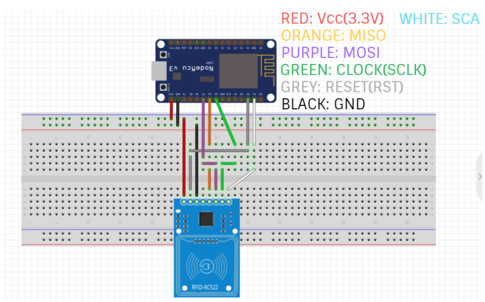

Figure 1. Schematic of the hardware architecture.

Thus, for the proposed hardware of the detection device, it is composed by a NodeMCU Lolin 1.0 and an RFID sensor with the MFRC522 chip model from the NXP company [NXP 2016]. The NodeMCU Lolin 1.0 is a microcontroller capable of executing algorithms for the processing of information obtained by sensors and input devices, generating, as a result, a desired response by the system. For the RFID sensor, it presents a small dimension, low consumption, working current among 13-26 mA at 3.3 V, and the capability of reading the information stored in the RFID tags (Figure 1).

3.2. Physical Device Algorithm

The algorithm of the physical device seeks to process the *ids* of radio tags detected by the RFID sensor attached to the physical device, sending this information to a communication API. To carry out this sending, it is necessary that the physical device establish a connection to an internet network and send the requests to the interested HTTP servers.

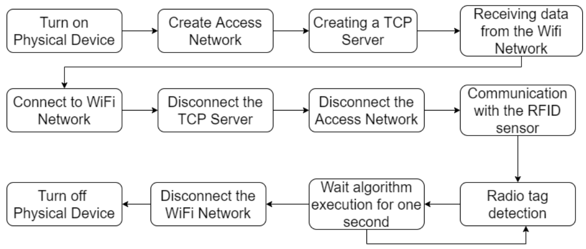

Figure 2. Flowchart execution of the physical device algorithm.

The algorithm has a procedure for creating an access network for cell phones and computers. This access network allows a connected cell phone or computer to connect to the TCP server which is run by the algorithm of the physical device. The TCP server must receive the network data (*name* and *password*) that will be used to connect to the WiFi network determined by the player (Figure 2).

By linking the device to an internet network, the algorithm is in charge of communicating with the RFID sensor to fulfill the requirement of detecting radio tags. Thus, the physical device prepares a JSON (Java Script Object Notation) with the information detected that must be passed to the API. Inside this JSON, other information can also be stored according to the developer interest, such as: the physical device identification; the connection data with the server; extra configurations for the game logic; and so on.

3.3. Communication API

Regarding the communication API organization, it will manage some pre-defined resource information, such as *categories* and *settings*, the monitored *data*, and the *user* information. They will provide the basic structure for the access and management of the detected data, as well as the representation of modeled information about the settings of the developed games, such as: the linking of the unique identity of the RFID tags with cards in a game; an image theme; matrix arrangement of the positioning of the game pieces; and so on.

For the communication API execution, five *endpoints* are available to work with the *categories* resources (Figure 3): two from the GET verb, one POST, one PUT and one DELETE. GET requests carry out the process of listing all saved categories or listing a stipulated category instance that has a fixed ID, which is provided as a parameter in the request. POST and PUT type requests have a validation process, which consists of not allowing more than one instance with the same name and that no attribute has an empty value. The DELETE request performs the removal of an instance that has a certain ID provided as a parameter by the request. For the *settings* resource, *endpoints* will verify the existence of the configuration category by the name of the *category* that was passed as a parameter in the request. If the configuration category has been found, the actions of reading, creating, deleting, and editing the associated *settings* can be performed. If it has not been found, a message response will be sent informing that the request was not accepted. For the *data* resource, the physical device makes a POST request containing the detected information and the connection *tag* that will be sent to all the clients that have linked to it. Finally, the *users* resource is responsible to prepare an access control to the web services, where the *categories* and *settings* resources are linked to the user of the system that created them.

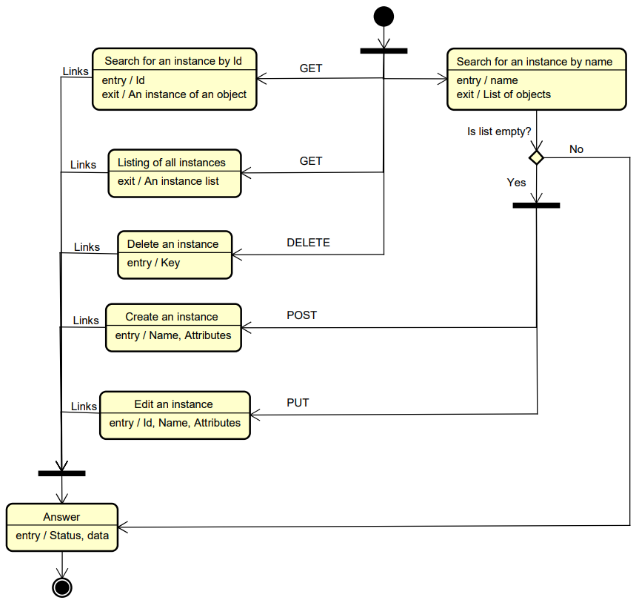

Figure 3. Activity diagram for the endpoints *categories* resource.

3.4. Mixed Reality Games Development

Considering the production of mixed reality games, the physical device algorithm allows the connection to any WiFi network available for the device. The game application connects to the open network of that device and sends the necessary data to establish the TCP connection to the device (Figure 4).

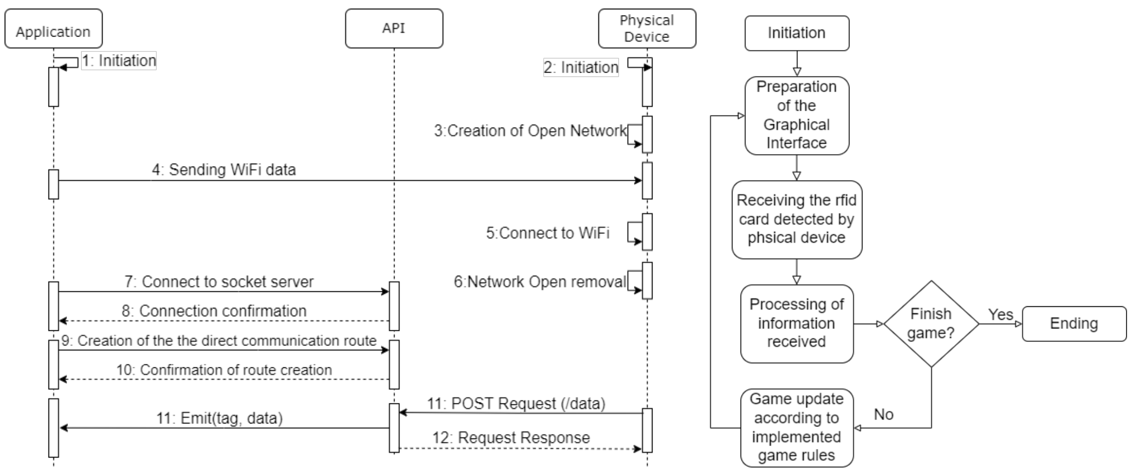

Figure 4. Sequence diagram for the physical device communication and flowchart for a basic game cycle.

For the application execution, Figure 4 demonstrates the basic game cycle of a game that uses the proposed architecture. The game, after its launch, must render a graphical interface and wait the reception of the information detected by the physical device. When it receives the information sent by the API, the game must process the information received and perform the game update, which is associated to the mechanics and dynamics of the proposed game. The game will be running until the application reaches a termination criteria for the game end, thus activating its termination routines.

4. Results and Discussions

Regarding the products/services offered by the proposed hardware/software framework, the first is a physical device capable of reading radio tags on physical objects. The second is the management of the settings that can be made available in the developed game, such as the digital linking to the unique identification of the radio tag with the physical object. The third represents the communication protocol itself, which aims to transport the information captured by the physical device to be used by the desired game.

Through these three products/services, two games were developed to validate the production of mixed reality games, according the proposed game cycle: Trimemória and FrutasFID (Figure 5). These games use cards that are linked to different RFID tags, and do not have complex features such as competition, ranking points, inventory, among others.

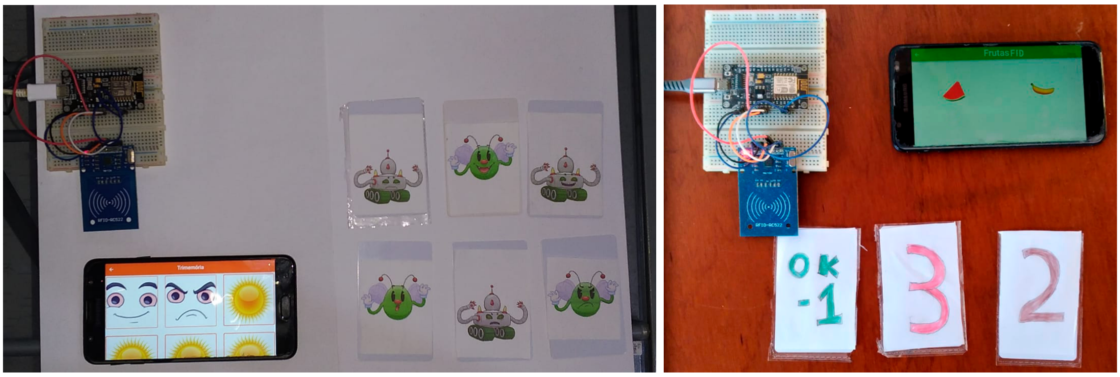

Figure 5. Trimemória and FrutasFID games, showing the physical devices, linked cards with RFID tags, and the respective mobile apps.

Trimemória (Figure 5) is similar to the classic memory card game, where the player must form pairs with cards that need to be discovered by their players. For the Trimemória game, the player must find three images of the same creature with different facial expressions. It is available to run on mobile or Desktop platforms. FrutasFID (Figure 5) consists of presenting a supply of fruits, and the player must count how many are being presented, which must inform the calculated amount. The supply of total amount of fruits will be made from the cards that contain radio tags linked to them. This game is available only for mobile platform.

5. Conclusion and Future Works

This work presented a hardware/software architecture able to support the production of games with mixed reality characteristics based on RFID sensors. To this end, some interesting assets were developed, such as: a model of a physical device for reading and transmitting identified radio tags; a communication protocol between the physical device and the support servers based on the REST standard; and examples of developed games capable of showing the viability of the proposed hardware/software architecture.

Regarding the use of *open hardware* and *software* technologies, which support the development of IoT solutions, these proved to be viable for the creation of a physical device prototype and the construction of webservices that used the information detected by him. The use of an API facilitated the creation of services, due to the ease of construction, maintenance, hosting, and the API consumption process.

As future work, it is intended to create an API documentation, so that the developer has support material on its use. The creation of new games and the improvement of Trimemória and FrutasFID games with the development of new functionalities, such as *multiplayer* games, is also necessary. Finally, we aim to create a hardware board that connects the NodeMCU and the RFID sensor without wires and *breadboard*, which facilitates the mass production of the physical device and its use in other environments.

References

Bohn, J. (2004). The smart jigsaw puzzle assistant: Using rfid technology for building augmented real-world games. In *Workshop on Gaming Applications in Pervasive Computing Environments at Pervasive*, volume 2004.

Deshmukh, S. and Baru, V. (2013). Applications of rfid in interactive board games. *International Journal of Electronics, Communication and Soft Computing Science & Engineering (IJECSCSE)*, page 87.

Herring, P., Sheehy, K., Jones, R., and Kear, K. (2010). Designing a virtual teacher for non-verbal children with autism: Pedagogical affordances and the influence of teacher voice.

Hinske, S. and Langheinrich, M. (2007). An rfid-based infrastructure for automatically determining the position and orientation of game objects in tabletop games. *Concepts and Technologies for Pervasive Games-A Reader for Pervasive Gaming Research*, 1:311–336.

Jia, X., Feng, Q., Fan, T., and Lei, Q. (2012). Rfid technology and its applications in internet of things (iot). In *2012 2nd international conference on consumer electronics, communications and networks (CECNet)*, pages 1282–1285. IEEE.

Lima, S. V. and Sarinho, V. T. (2019). Developing an open hardware rfid architecture for electronic board games. In *Proceedings of the XVIII SBGames*, pages 587–590. SBC.

Lima, S. V. and Sarinho, V. T. (2020). A memory game proposal for facial expressions recognition in health therapies. In *International Conference on Entertainment Computing*, pages 264–269. Springer.

Miglino, O., Di Ferdinando, A., Di Fuccio, R., Rega, A., and Ricci, C. (2014). Bridging digital and physical educational games using rfid/nfc technologies. *Journal of e-Learning and Knowledge Society*, 10(3).

NXP (2016). Mfrc522 standard performance mifare and ntag frontend. Último acesso: 18/10/2021.