INTEGRATED ELECTRONIC LOCK SYSTEM WITH WEB SERVER FOR ENHANCED SECURITY

Pedro Quadros de Freitas[a, b],Ana Luiza Cantharino Maciel[a, b], Daniel Januario de Aragão Rocha[a, b],André Lucas Coêlho[a, b],João Victor Freire Silva[a, b],Gustavo Gomes Barreto[a, b].
[a] Senai Cimatec, Brazil
[b] Junior Researcher at IEEE CAS CIMATEC

Abstract: This article introduces a prototype of an integrated electronic lock system with a web server for enhanced security in the contemporary post Covid-19 society and the changing in locking mechanisms. The goal is to combine key, RFID, and password-based access control with a remote web server for managing permissions. The methodology includes programming hardware control and data management algorithms. The firmware utilizes the ESP32 microcontroller and custom libraries to control inputs like RFID and a matrix keyboard. The server-side enables real-time communication with a Firebase database using Websockets. The project also includes the design of printed circuit boards and the mechanical structure of the lock.

Keywords: Lock; Security; Web Server; ESP32; RFID.

SISTEMA DE TRANCA ELETRÔNICA INTEGRADA COM SERVIDOR WEB PARA AUMENTO DE SEGURANÇA

Resumo: Este artigo apresenta um protótipo de um sistema integrado de fechadura eletrônica com um servidor Web para aumentar a segurança na sociedade contemporânea pós-Covid-19 e a mudança nos mecanismos de travamento. O objetivo é combinar controle de acesso baseado em chave, RFID e senha com um servidor Web remoto para gerenciar permissões. A metodologia inclui a programação de algoritmos de controle de hardware e gerenciamento de dados. O firmware utiliza o microcontrolador ESP32 e bibliotecas para controlar entradas como RFID e um teclado matricial. O servidor permite a comunicação em tempo real com um banco de dados Firebase usando Websockets. O projeto também inclui o design de placas de circuito impresso e a estrutura mecânica da fechadura.

Palavras-chave: Tranca; Segurança; Servidor Web; ESP32; RFID.

1. INTRODUCTION

The practice of utilizing mechanisms to secure doors and safeguard valuable possessions dates back to ancient times, notably with the Romans. However, in response to growing concerns about violence and the need for modern solutions to personal and executive security, electronic lock technology has undergone significant advancements in recent decades [2-4].

Furthermore, the global smart door lock market has experienced remarkable expansion, with a valuation of $2.13 billion in 2022 and a projected increase to $8.21 billion by 2030, commencing from $2.49 billion in 2023 [5]. This surge is particularly attributed to the pandemic, as well as the escalating demand for devices that reduce the need for human interactions.

In light of the diverse range of applications, a multitude of mechanisms for activating latches have emerged, including keys, passwords, pins, cards, RFID cards, biometric methods, and more [6][7][8]. These technologies span from basic mechanical structures to intricate and highly secure products. However, despite this variety, there is currently no commercially available mechanism that seamlessly integrates all three primary systems: key, RFID, and password-based access control, coupled with a remote server for centralized control of entries and system updates.

Therefore, the goal of this article is to propose a prototype of an integrated electronic lock system with a web server for enhanced security using key, RFID, and password-based access control.

2. METHODOLOGY

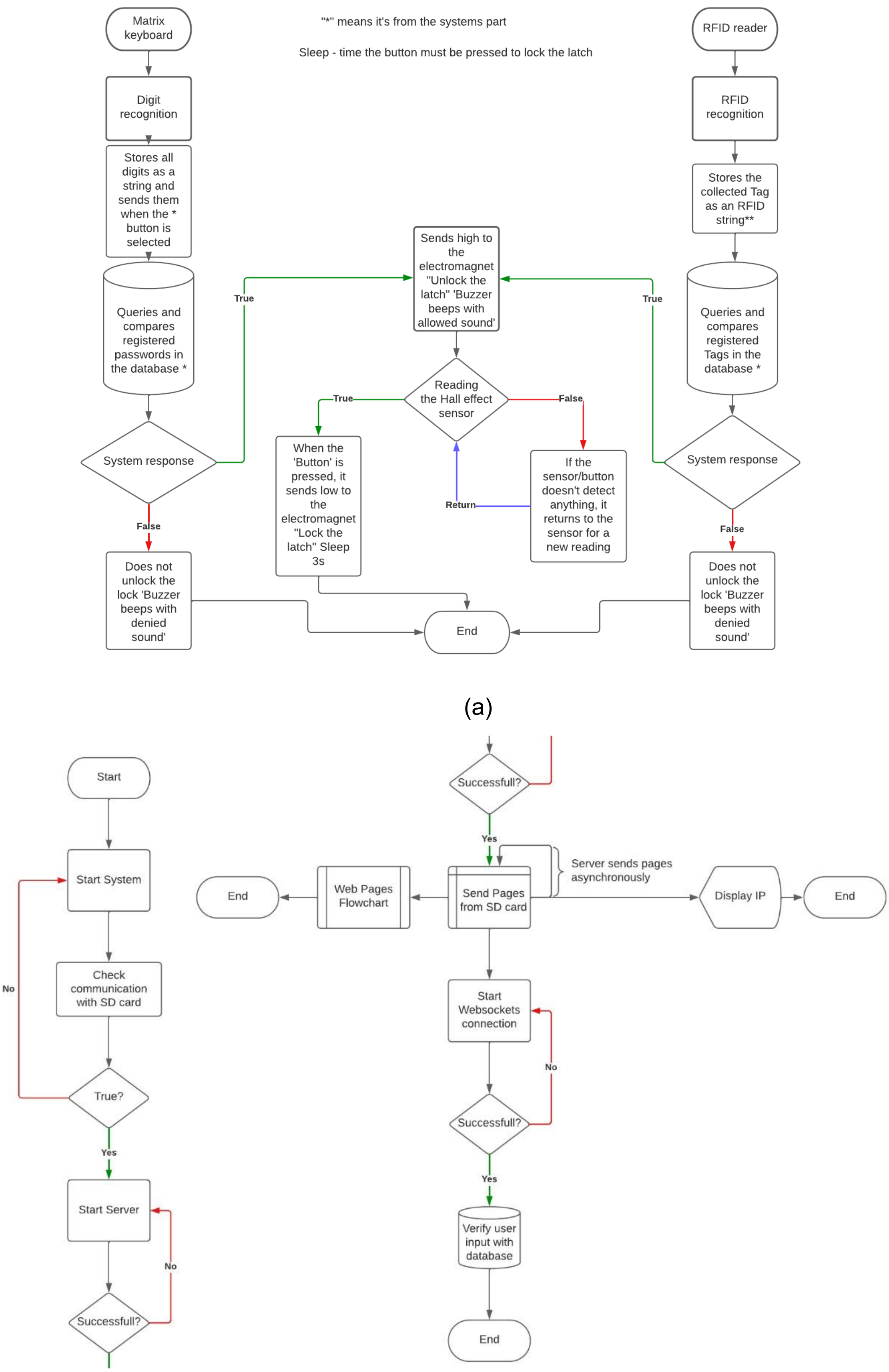
The main goal of this article was to design the programming functionalities for an electronic lock system and its corresponding web server management. To meet the requirements of such a system, an exploration of alternative materials for project integration was carried out. This investigation laid the foundation for the development of both programming and mechanical models for future prototype testing.

In terms of programming, this project can be categorized in two primary parts: hardware control and data management. Both components involve the integration of various input readings and task structures. Hardware control concentrates on managing multiple tasks and controlling external actuators. The data section primarily focuses on registering and managing permissions within the security system.

To integrate the different mechanisms of electronic locks, as previously defined in this article (RFID, physical keys, and passwords), an algorithm was developed using the C++ programming language. To provide an overview of the thought process behind the code for both hardware and data management sections, flowcharts were created, as shown in Figure 1.a and Figure 1.b, respectively [9-10].

For the physical assembly of the project, the lock is powered by Li-Ion batteries, and internal movement is facilitated by a stepper motor. The system incorporates two motherboards with ESP 32, designated for the server and the lock. The 3D model for the lock case and printed circuit boards (PCB) were developed by the project team. A list of materials is provided in Table 1.

Figure 1. Programming Flowchart. (a) System Control, (b) Lock management



(b)

ISSN: 2357-7592

Table 1. List of materials

Component	Quantity (unit)
28BYJ48 stepper motor	1
Li-Ion battery pack	1
Lock motherboard with ESP32	1
Server motherboard with ESP32	1
3D printed case for lock	1
Electrical cables	N.A.
Designed PCB's	2
RFID reader	1
Matrix keyboard	1
Buzzer	1

3. RESULTS AND DISCUSSION

Based on the flowcharts in Figure 1, the project's algorithm was developed, dividing it into three components: locking mechanism, server system, and firmware. Additionally, to apply the program, printed circuit boards and a mechanical structure model were constructed.

The lock code project was evolved within the Visual Studio Code development environment. PlatformIO extension was utilized for programming the ESP32 microcontroller and organizing its core functionalities into custom libraries through the use of classes.

3.1 Controlling the lock

The control code is divided into four distinct sections: System, RFID, Keyboard, and Server.

The "System" segment governs the core system operations. It features an initialization function responsible for creating the main system task, which runs in an infinite loop and executes various system-related tasks.

The "RFID code" manages communication between the MFRC522, an RFID reader module, and the lock system. It incorporates an initialization function to set up the RFID reader, create tasks, and manage queues (objects designed to facilitate asynchronous communication between two or more tasks) for storing read IDs. A primary task tracks ID readings, monitors queue events, and dispatches IDs to another queue. Additionally, functions handle RFID reader initialization and address interruptions during the reading process.

The "keyboard" division focuses on a matrix keyboard, facilitating character input. It defines the keyboard layout, incorporates a class for keyboard functionality, and provides functions for keypress detection. A dedicated task is initiated to continuously read keyboard inputs and send the characters to a queue, operating in a loop with slight delays between each read.

The "server code" encompasses essential libraries for the project, establishes a Wi-Fi connection, and executes HTTP requests. Functions are employed to send POST and GET requests to specific URLs, manage HTTP responses, and display corresponding messages. Moreover, PHP code is utilized to process requests received from the device, generate HTTP headers based on received variables, and manage responses, making the PHP server responsible for handling requests sent by the device.

3.2 The server

To establish communication with the server, a microcontroller connected to a WiFi network is utilized. This microcontroller transmits an HTML-formatted text document via a local IP. This text document is enhanced by a JavaScript script, which facilitates page functions and dynamic updates, and a CSS (Cascading Style Sheets) document for styling page elements. Following login and password entry, the page allows users to check the lock entry history, modify access levels and passwords, create and remove users.

For real-time communication with the server and the database, the page utilizes connection functions with Firebase, as explained earlier. To validate user inputs and enable real-time interactions, a Websockets communication protocol (full-duplex transmission control protocol) is employed. This protocol initiates a connection with a handshake in HTTP between the page and the server, maintaining bidirectional communication to expedite data transmission and automatically terminating data transfer upon communication loss.

The page documents are stored on an SD card and accessed through a microcontroller connection module, allowing direct server-side sending. Additionally, the system is equipped with an RFID module, as previously described, for registering new TAGs. While initially, there was a proposal to use ".txt" format text documents on the SD card for storing user data, an analysis of the effort required revealed the practicality and versatility of using the realtime database service from Firebase, leading to its adoption.

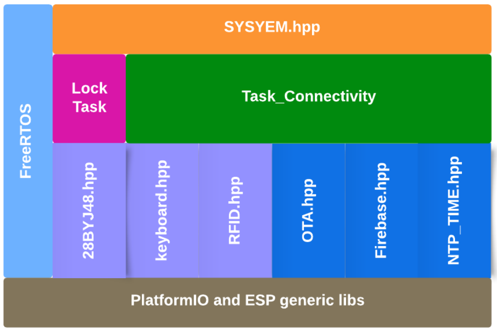
3.3 The Firmware

The imported libraries include many functionalities, including communication with the Firebase database, NTP server connectivity, RFID module control, and Real-Time Operating System (FreeRTOS) management. Custom libraries like "28BYJ48.hpp" offer enhanced control over stepper motors for lock creation.

For communication purposes, the "keyboard.hpp" library identifies button presses and dispatches requests via a queue to the connectivity task, while the "RFID.h" library monitors RFID cards and transmits data through a queue to the connectivity task. The "OTA.hpp" library enables code updates via Wi-Fi without the need for a direct PC connection. Additionally, the "Firebase.hpp" library simplifies the connection to Google's Firebase server, and "lib NTP_TIME.hpp" streamlines time management with time-stamp functions.

The "Task_Connectivity" task is responsible for managing internet connectivity, server interactions, and OTA access point creation. In parallel, the "Lock_Task" task governs lock-related outputs, such as the motor and buzzer, using a state machine. The "lib SYSTEM.hpp" library stores task and connection information. This structured approach facilitates the development of an advanced and versatile lock system and is summarized in Figure 2.

Figure 2. Firmware programming diagram

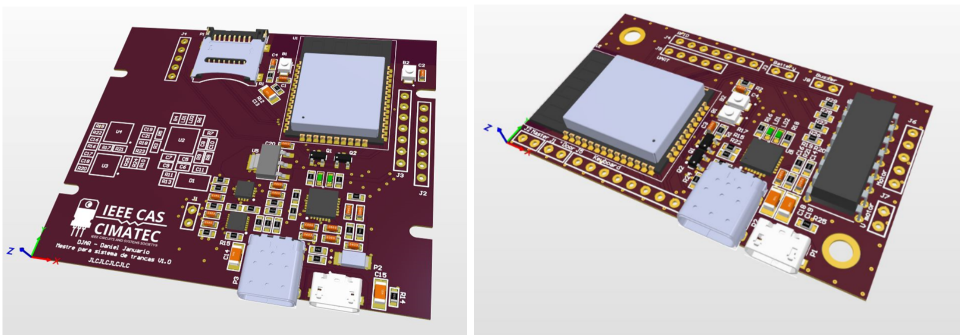


3.4 Printed Circuit Board (PCB)

The printed circuit board (PCB) was designed using Altium, specialized PCB design software. In Figure 3.a, the server board features two USB connectors (USB C and micro-USB), providing flexibility for the connector by efficiency or availability. Additionally, it includes an SD card slot for storing web pages essential for database management, code recording, RFID connection, debugging, and battery control.

For the lock component, a board akin to the server board was developed, as shown in Figure 3.b. It shares similarities with the server, including the two USB connectors, battery system, and RFID capabilities. However, it distinguishes itself with dedicated outputs for two sensors, a matrix keyboard, and a stepper motor. Notably, the most prominent IC on the right side of the board controls the motor. Moreover, the board features an output for a buzzer, providing a lock status warning to the user. Acting as the controller for this system is the ESP32.

Figure 3. Project's Printed Circuit Boards. (a) Lock board, (b) Server board.



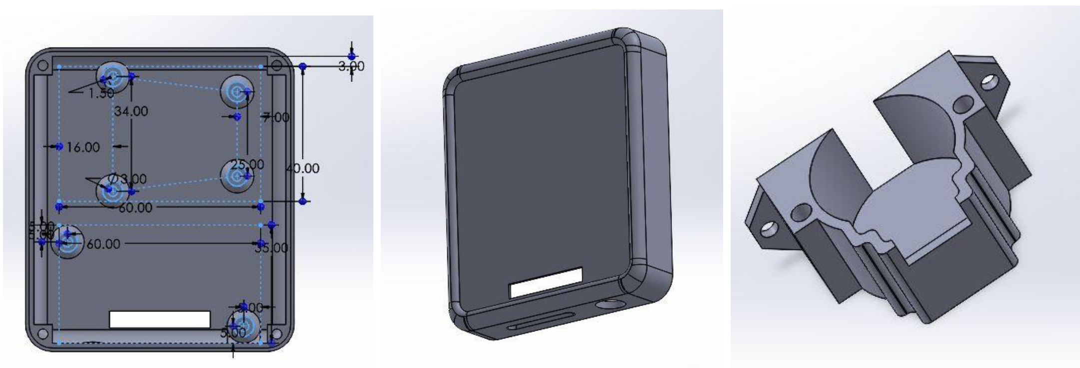
. (a) (b)

3.5 Mechanical Structure Modeling

The lock modeling process is currently in progress using SolidWorks software. Figure 4 illustrates the model employed for the cabinet lock, with the top portion indicating the outline of the RFID module and the bottom representing the plate that will be manufactured. A channel is included beneath the model to accommodate the cable of the matrix keyboard, which will be positioned at the front.

In Figure 4.b., a recess is noticeable, designed for the placement of the keyboard on the front side. The lower section of this figure showcases the plate's connections for essential cables and USB. Figure 4.c illustrates the support structure for the 28BYJ48 stepper motor, which will facilitate the rotation of the cabinet latch's axis.

Figure 4. 3D Model of the Mechanical Structure. (a) Inside view, (b) Outside view, (c) Motor Support.



(a) (b) (c)

4. CONCLUSION

This document provides a thorough and detailed overview of the construction of an electronic lock integrated with a web server capable of receiving RFID signals and passwords, while also enabling mechanical unlocking via a motor mechanism.

Furthermore, it encompasses the design of printed circuit boards and the development of effective communication among all project components, including firmware, hardware, and software for both the server and the lock.

Moving forward, the subsequent phases of this project involve 3D printing the designed mechanical model, manufacturing the printed circuit boards, prototype testing and orchestrating the assembly of all system components for deployment at the IEEE Cimatec Headquarters.

Acknowledgments

To IEEE Circuits and Systems CIMATEC (CAS CIMATEC) and LAB MAKER CIMATEC.

5. REFERENCES

[1]Britannica, T. Editors of Encyclopaedia. "lock." Encyclopedia Britannica, May 12, 2023. Available at: https://www.britannica.com/technology/lock-security. Accessed on: 27 jun. 2023.

[2]SMITH COLLEGE. Ancient Inventions. Available at: https://www.smith.edu/hsc/museum/ancient_inventions/hsc09b.htm. Accessed on: 27 jun. 2023.

[3]HISTORICAL LOCKS. Roman Door Locks. Disponível em: https://www.historicallocks.com/en/site/h/other-locks/19-keys-and-locks-from-imperial-rome/roman-door-locks/. Accessed on: 27 jun. 2023.

[4]LOCK COLLECTORS EUROPE. Lock Database - BAR1. Available at: http://www.lockcollectors.eu/members/lockdatabase/bar1/_index.htm. Accessed on: 27 jun. 2023.

[5]Smart Door Lock Market Size, Share, Trends | Global Report [2027]. Available at: <https://www.fortunebusinessinsights.com/industry-reports/smart-door-lock-market-100215>. Accessed on: 04 sept. 2023.

[6]Longo, Vinicius de Almeida Uma abordagem das tecnologias, dispositivos e aplicações de sistemas eletrônicos de segurança. Universidade Estadual Paulista (Unesp), 2022. Available at: http://hdl.handle.net/11449/235776. Accessed on: 27 jun. 2023.

[7]SILVA, Alfredo Rodrigo Sousa. IF access: sistema de controle de acesso eletrônico utilizando tecnologia RFID e microcontrolador. 2022. Projeto de conclusão de curso. Available at: https://repositorio.ifpb.edu.br/handle/177683/2464. Accessed on: 27 jun. 2023.

[8]PINO, Alessandro Mouras de Oliveira. Desenvolvimento de protótipo de um sistema embarcado de fechadura eletrônica para controle de homologação e acesso. 2019. Available at: https://repositorio.ufersa.edu.br/handle/prefix/6643. Accessed on: 27 jun. 2023.

[9]STEVEWHIMS. Exemplo de código C/C++ definindo prioridade da tarefa - Win32 apps. Available at: https://learn.microsoft.com/pt-br/windows/win32/taskschd/c-c-code-example-setting-task-priority. Accessed on: 30 jul. 2023.
[10]AGO, D. #UTOPIAN-IO • 5 Y. MFRC522 RFID Reader Arduino Library Setup and Functions Usage Shown With an Example Project - Tutorial. Available at: https://steemit.com/utopian-io/@drencolha/mfrc522-rfid-reader-arduino-library-setup-and-functions-usage-shown-with-an-example-project-tutorial. Accessed on: 30 jul. 2023.