

People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific
Research



University of Saad Dahlab Blida

Faculty of Technology
Department of Electronics

Master's Memory

Presenting by :

Melaoui Billel
Houssef-Eddine Hammouda

Development of a platform
website for the intelligent
management and monitoring of the
CPi-SPA data center

Proposing by : Mr. ANOU Abderrahmane

Co-supervisor : DAHIA Mohamed Nabil DGA S.I. (CPI spa - Bank of
Algeria branch-)

Year: 2023/2024

Acknowledgements

First and foremost, I would like to thank ALLAH, who has helped me and given me patience and courage throughout these years of study. I want to express my deep gratitude and sincere appreciation to my supervisor, Mr. Anou, as well as to the CPI team, for their help and advice throughout the completion of this project.

I would like to express my gratitude to the members of the jury who honor me by evaluating this final study memory.

I also thank my entire family, especially my dear parents, for their encouragement, patience, and unwavering support throughout these years of study.

Finally, I would like to thank all the people who have helped us, directly or indirectly, during our memory.

Thank you very much

خلاصة

من أجل ضمان المتابعة والإدارة الذكية، نقترح على الشركة تصميم وتنفيذ منصة . بالاستفادة من قدرات Raspberry Pi وArduino، يمكن للمنصة جمع ومعالجة بيانات المستشعرات بكفاءة، وأتمتة عمليات الفروع، وتعزيز الأمان . توفر أنظمة الواجهة الأمامية والخلفية تفاعلاً سلساً مع المستخدمين، وإدارة قوية للبيانات، وتحليلات متعمقة .

لذلك، سيكون هدف مشروعنا هو إنشاء منصة لمركز بياناتهم والمناطق الحساسة التي تحمي البنية التحتية لتكنولوجيا المعلومات، مما يوفر لهم إدارة أسهل وأكثر فعالية، بالإضافة إلى جعله أكثر استقلالية واستدامة . سنستخدم مستشعرات مختلفة في عدة أماكن، بما في ذلك الحرارة والرطوبة والضغط ومستشعرات الحركة، وغيرها، التي سنقوم بربطها بهذه المنصة، مما سيمكننا من الحصول على جميع المعلومات اللازمة في الوقت الفعلي وكذلك إشعارات التنبيهات، مما يتيح للإدارة اتخاذ قرارات سريعة وفعالة في أي وقت .

سيساعد مشروعنا الشركة على إنشاء نظام قادر على اكتشاف تشوهات المعدات المرتبطة بمخاطر الحوادث مثل الأعطال والحرائق، وإعلام الموظفين بمثل هذه المشكلات . بالإضافة إلى ذلك، سيمكن هذا النظام من إدارة الوصول خلال ساعات العمل لتجنب التسلات غير المرغوب فيها أو الخبيثة من قبل الموظفين .

Abstract

In order to ensure intelligent monitoring and management, we propose to the company the design and realization of a platform by leveraging the capabilities of Raspberry Pi and Arduino, the platform can effectively collect and process sensor data, automate branch operations, and enhance security. The frontend and backend systems provide seamless user interaction, robust data management, and insightful analytics

The goal of our project will therefore be to implement a platform for their Datacenter and sensitive areas that protects the IT infrastructure, which will offer them easier and more efficient management of the latter, in addition to making it more autonomous and sustainable. We'll be using different sensors in several places, including, (heat, humidity,

Abstract

pressure, motion sensors, etc.) that we will connect to this platform, which will allow us to have all the necessary information in real time as well as the notification of alerts, thus allowing management to make a quick and efficient decision at any time.

Our project will help the Corporation put in place a system that can detect equipment anomalies associated with the risks of incidents such as failures and fires, alerting staff to such problems. In addition, this system will allow access to be managed during working hours to avoid awkward or malicious intrusion by staff.

Resumé

Afin d'assurer un suivi et une gestion intelligents, nous proposons à l'entreprise CPA la conception et la réalisation d'une plate-forme. En tirant parti des capacités de Raspberry Pi et Arduino, la plate-forme peut collecter et traiter efficacement les données des capteurs, automatiser les opérations des branches et améliorer la sécurité. Les systèmes frontend et backend offrent une interaction parfaite avec les utilisateurs, une gestion robuste des données et des analyses approfondies.

L'objectif de notre projet sera donc de mettre en place une plateforme pour leur Datacenter et les zones sensibles qui protège l'infrastructure informatique, ce qui leur offrira une gestion plus facile et plus efficace de ce dernier, en plus de le rendre plus autonome et durable. Nous allons utiliser différents capteurs dans plusieurs endroits, y compris, (chaleur, humidité, pression, capteurs de mouvement, etc.) que nous allons connecter à cette plate-forme, ce qui nous permettra d'avoir toutes les informations nécessaires en temps réel ainsi que la notification des alertes, permettant ainsi à

Abstract

la direction de prendre une décision rapide et efficace à tout moment.

Notre projet aidera la Société à mettre en place un système capable de détecter les anomalies d'équipement associées aux risques d'incidents tels que les pannes et les incendies, en alertant le personnel à de tels problèmes. En outre, ce système permettra de gérer l'accès pendant les heures de travail afin d'éviter des intrusions désagréables ou malveillantes par le personnel.

Table of content

- General introduction 1
- Chapter I:Current State Analysis..... 3
 - 1.1 Introduction..... 4
 - 1.2 Presentation of the host organization 4
 - 1.3 The C.P.I spa shall:..... 4
 - 1.4 Presentation of the case study 5
 - 1.5 General Organization..... 6
 - 1.6 Physical Security Policy..... 7
 - 1.5.1 Physical security management processes and surveillance system alerts..... 7
 - 1.6 Human Resources Management Process 9
 - 1.6.1 Roles and Responsibilities 9
 - 1.6.2 Existing Approach to Human Resources Management..... 9
 - 1.6.3 Management of new recruits 9
 - 1.6.4 Management of departures10
 - 1.6.5 Reactivation management10
 - 1.7 Diagnostic and Critical.....10
 - 1.8 Risk Analysis.....13
 - 1.10 Security goals.....38
 - 1.10 Areas for improvement.....39
- Chapter II: State of the Art42
 - 2.1 Introduction.....43
 - 2.2 HISTORY AND EVOLUTION.....43
 - 2.3 EDUCATION AS THE MAIN FOCUS OF THE FOUNDATION43
 - 2.4 WHAT IS RASPBERRY PI?.....44
 - 2.6 History of Arduino.....45
 - 2.7 What is Arduino?.....46
 - 2.10 The difference between Raspberry pi and arduino47
 - 2.10.1 Processing Power:48
 - 2.10.2 Operating System:48
 - 2.10.3 Applications:48
 - 2.10.4 Besides:48
 - 2.11 Overseeing Framework.....49
 - 2.12 Main Achievements & Technologies49
 - 2.12.1 Integration between Raspberry Pi and Arduino:.....49
 - 2.12.2 Web Interface through HTML/CSS/JS:50
 - 2.12.3 Backend Development with Python and Flask:50

Table of Content

2.12.4 Database Management with SQLite:	50
2.12.5 Real-Time Monitoring and Control:	50
2.12.6 Advanced Access Control:	51
2.12.7 Automated Email Notifications:	51
2.13 Use Cases and Benefits	51
2.13.1 Operational Efficiency:	51
2.13.2 Security and Compliance:	51
2.13.3 Proactive Maintenance:	52
2.14 Conclusively	52
Chapter III: Conception and Realisation.....	53
3.2 Integration and Implementation:	56
3.2.2 Configuration of the sensors:	57
3.3- Hardware Installation and Configuration	58
3.1 Setting up and adjusting the Raspberry Pi 4 Model B:.....	58
3.3.2 Installation and Setup of the Water Sensor:.....	60
3.3.3 Installing and configuring the DHT11 temperature and humidity sensor:	63
3.3.4 Installation and configuration of the Motion Capture MH- SR602:.....	67
3.3.5 Installation and configuration of the MH-Sensor series:..	71
3.3.6 Installation and setup of the MQ135 capteur with Arduino Uno:.....	74
3.3.6.2 Step 2: Write and upload the Arduino code.....	75
3.3.7 Installation and configuration of DY50 digital fingerprint	75
3.3.9 Installation and setup of a web camera:	81
3.3.10 Honeypot:	84
3.4 platform:.....	89
3.4.1 Overview of the Platform	89
3.4.2 Integrating Real-Time Data:	90
3.4.3 Improving Cybersecurity:	90
3.4.4 Notification System:	91
3.4.5 Technology Employed	91
3.5 Integration of Composants	97
3.5.1 Process of Integration:	97
3.5.2 deployment in the datacenter	97
3.6 Methods of Testing and Obtained Results	97
3.6.1 Methods of Test:	97
3.6.2 Results obtained:	98

Table of Content

3.7- Tests and Validation.....	98
3.7.1 Testing Methodology	98
3.7.2 Tests of integration:	98
3.7.3 Performance tests	99
3.7.4 Evaluation of Performance:	99
3.7.6 Validation of Functionalities	100
3.7.7 Identifying Limitations and Proposals for Improvement...	101
3.7.8 Propositions for Improvement:	101
3.8 Comparison with Traditional Methods	102
3.9 Conclusion:.....	102
General Conclusion	Error! Bookmark not defined.
References	107

List of Figures

Chapter I

Figure 1.1: CPI spa organizational chart.....	6
---	---

Chapter II

Figure 2.1: the architecture of Raspberry Pi.....	44
Figure 2.2: Arduino and it forms.....	47
Figure 2.3: Arduino Architecture.....	47
Figure 2.4: Raspberry pi 4 vs Arduino Uno.....	49

Chapter III

Figure 3.1: Overview of the whole process.....	54
Figure 3.2: Raspberry Pi 4 Model B.....	58
Figure 3.3: Water Sensor.....	60
Figure 3.4: Python for water_sensor.py.....	61
Figure 3.5: python water_sensor.py Dans le terminal.....	62
Figure 3.6: temperature and humidity sensor.....	63
Figure 3.7: Example code for reading the sensor temperature and humidity python temphum.py.....	66
Figure 3.8: python temphum.py Dans le terminal.....	66
Figure 3.9: Figure 3.8: Motion Capture.....	67
Figure 3.10: Example code for reading the captures Motion Capture	69
Figure 3.11: python motion.py Dans le terminal.....	70
Figure 3.12: MH-SenSor series.....	71
Figure 3.13: Example code for reading the captures MH-SenSor.....	72
Figure 3.14: python flame.py Dans le terminal.....	73
Figure 3.15: the MQ135 capteur with Arduino Uno.....	74
Figure 3.17: DY50 digital fingerprint sensor with Arduino Uno.....	75
Figure 3.19: how's arduino sends data to the raspberry pi.....	78
Figure 3.20: Example code for reading Arduino Uno with a Raspberry Pi.....	80
Figure 3.21: web camera.....	81
Figure 3.22: Example code for reading web camera.....	83
Figure 3.23: python Dans le terminal.....	84
Figure 3.24: Honeypot.....	84
Figure 3.25: Example code for reading Honeypot.....	Error! Bookmark not defined.
Figure 3.26: how the honeypot actually works.....	89
Figure 3.27: the application's login.....	93
Figure 3.28: Therats.....	94
Figure 3.29: Add Employee.....	94
Figure 3.30: MangerUser.....	95
Figure 3.31: Mange Users.....	95
Figure 3.32: Add User.....	96

List of Figure

Figure 3.33:Sensors Settings.....96

List of Tables

Table 1.1: Risk analysis, sources of threats..... 17
Table 1.2: Risk analysis, evaluation metrics..... 18
Table 1.3: Risk analysis, availability scale..... 19
Table 1.4: Risk analysis, integrity scale..... 19
Table 1.5: Risk analysis, confidentiality scale..... 20
Table 1.6: Risk analysis, traceability scale..... 20
Table 1.7: Risk analysis, severity scale..... 21
Table 1.8: Risk analysis, likelihood scale..... 22
Table 1.9: Risk analysis, identified assets..... 25
Table 1.10: Feared events..... 33
Table 1.11: Risk analysis, traceability scale..... 34
Table 1.12: Risk analysis, criticality matrix..... 37
Table 1.13: Risk analysis, integrity scale..... 37
Table 1.14: Risk analysis, integrity scale..... 39

KEY-WORDS

- 1 Intelligent Management (IM)
- 2 Applications Programming Interface (API)
- 3 Interbank Pre-Clearing Center (CPI)
- 4 Credit Popular of Algeria (CPA)
- 5 Internet of Things (IoT)
- 6 Java Enterprise Edition (JavaEE)
- 7 JavaScript (JS)
- 8 Hypertext Markup Language (HTML)
- 9 Cascading Style Sheets (CSS)
- 10 Structured Query Language (SQL)
- 11 Raspberry Pi (RP)
- 12 Arduino
- 13 Data Center (DC)
- 14 Sensors (temperature, humidity, pressure, motion)
- 15 Security
- 16 Automation
- 17 Real-Time Data
- 18 Frontend Systems
- 19 Backend Systems
- 20 Data Management (DM)
- 21 Analytics
- 22 Equipment Anomalies (EA)
- 23 Incident Risks (failures, fires)
- 24 Staff Alerts
- 25 Access Management (AM)
- 26 Cybersecurity
- 27 Notification System (NS)
- 28 Performance Testing (PT)
- 29 Functionalities
- 30 Limitations and Proposals for Improvement

General Introduction

General introduction

Synergy Research reports that in 2018, Investments in data centers reached a record \$150 billion, an increase of 17% over the previous year. The ever-increasing demand for data, traffic, and processing power in data centers is partly responsible for this phenomenal rise. Data centers are essential to our modern digital civilization and to our daily lives. Everything these days is linked to data centers. This includes our cellphones, laptops, connected devices, fridges, and even cardiac stimulators.

But there are certain difficulties in managing these data centres just because they exist. Dangers of harm from calamities like fires, Floods, and hostile incursions are among these obstacles. Equipment crashes and other hardware-related issues are also possible. Consequently, These issues must be resolved if data centers are to function efficiently and securely.

With The primary objective of establishing and running an automated system for clearing large payments dematerialized by commercial banks, The Bank of Algeria has set up the Interbank Pre-Calculatation Centre (C.P.I spa). Because of its critical role in controlling account balances and handling customer account fluctuations, This system is susceptible to a number of threats; hence, It is imperative to identify a solution that can foresee and prevent these threats, Or swiftly resolve them if they do occur.

In the first chapter, We cover the existing state of affairs; in the second, We go over the current tools and keys we going to use to create that platform, Covering subjects like raspberry pi, Arduino, And the platform we going to create, And information system security; And in the third, We cover the

General Introduction

design and realization of our platform and how it functions, Which makes up the contribution part of this document.

The first of this chapter's four sections will focus on the current state of affairs at the CPI spa level, outlining the issues with the current remedies. Researching the intended system is the next stage. The last step is to plan and execute the target system's design.

Chapter I: Current State Analysis

1.1 Introduction

Before approaching the study of the solution, It is essential to carry out a thorough analysis of the current state of supervision and security of the environment that houses the A.T.C.I. mass payment system in this section, We will first introduce our host organization. Then, We will provide organizational and technical details on the supervision and security of the current environment that houses the A.T.C.I mass payment system, in the context of a typical organization that is the subject of our case study. Finally, we will conclude with a diagnosis to identify gaps and a risk analysis to define the main areas for improvement.

1.2 Presentation of the host organization

On 04 August 2004, the Banque d'Algérie created the Interbank Pre-Clearance Centre (C.P.I), whose main objective is the implementation and operation of an automated system for clearing dematerialized mass payments by commercial banks.

The C.P.I spa is a technical operator that manages and administers the Interbank Tele-Compensation (T.C.I) system by delegation from the Banque d'Algérie. The A.T.C.I system ensures the electronic clearing of cheques, bills, card transactions, transfers and direct debits exchanged between the participants who are the Bank of Algeria, the banks, the Treasury and Algeria Post.

1.3 The C.P.I spa shall:

- The management of the system that ensures the electronic clearing of mass payment instruments (cheques, bills, transfers, direct debits and electronic payment transactions).
- The execution of the due diligence necessary for the smooth running of technical operations that condition the operation of the A.T.C.I.

- The calculation and dumping of multilateral tele-compensation balances into the real-time gross settlement system.
- Archiving of data, scanned images and telecom values.

1.4 Presentation of the case study

The Centre de Pre Compensation Interbancaire (CPI) operates the mass payment clearing and exchange system. It has four main functions: exchange management, tele-clearing, net settlement movements and data archiving.

The central Tele-Clearing platform, managed by the CPI, and for which it is responsible, is designed to control and ensure a secure and automated interbank exchange of mass payments and their clearing according to the rules of neutrality and transparency.

Because of the importance of the centre's role in national monetary transactions, its security is a crucial issue. In particular, we will focus on the physical security of the center's datacenter. It can be vulnerable to a wide variety of threats (employee access at unauthorized hours, unauthorized access, fire).

As part of our work, we are conducting our study of the existing. In the following, we will present the general organization, the existing business and technical. As well as the anomalies identified and a risk analysis. Finally, we propose a set of areas for improvement.

1.5 General Organization

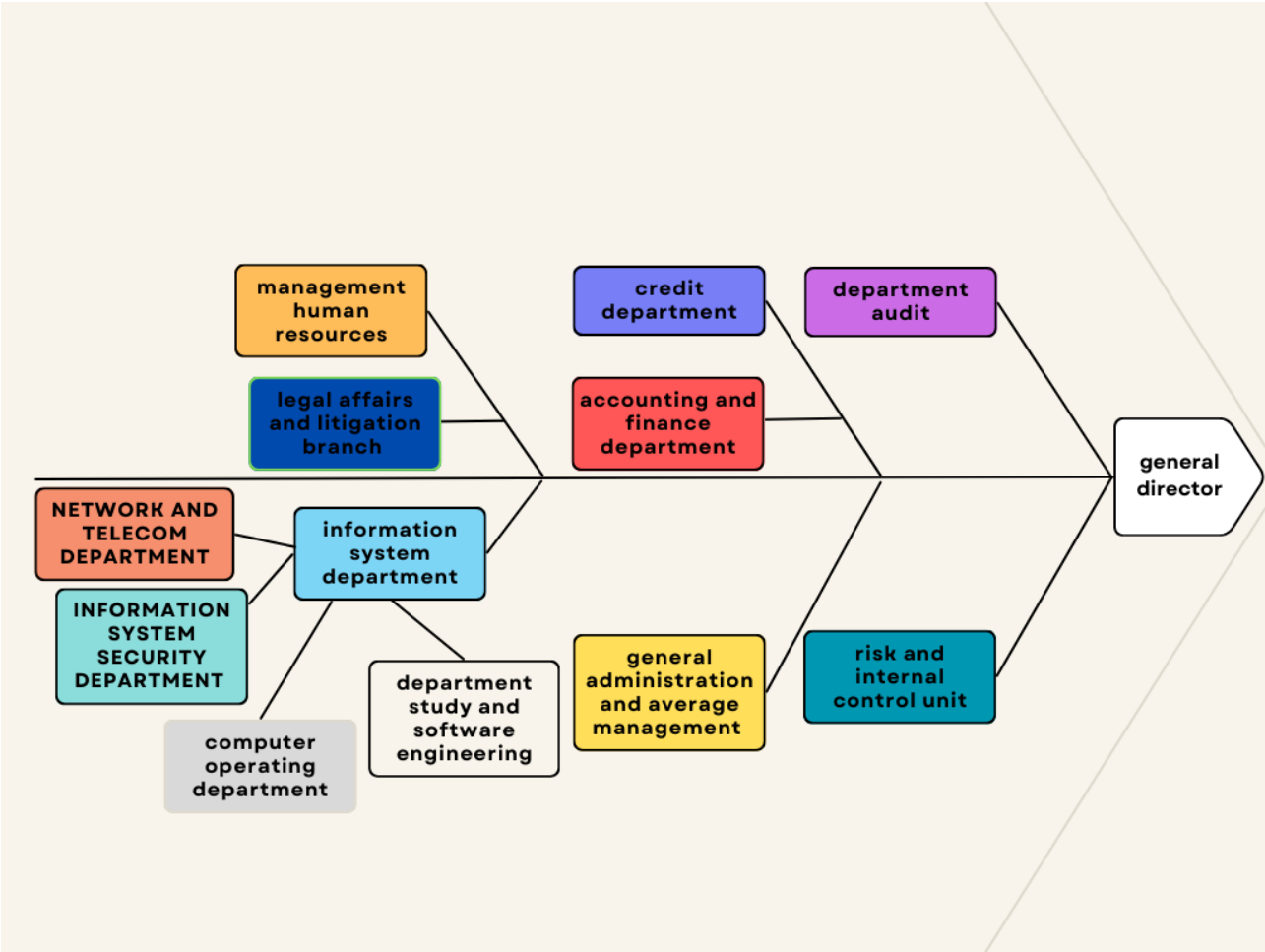


Figure 1.1: CPI spa organizational chart.

CPI spa which is the subject of our study is structured according to the following organizational chart: A General Directorate, the Audit Department, the Risk and Internal Control Cell are directly linked to the General Directorate. The rest of the functions are divided into directorates that ensure the operational and administrative aspect of the organization.

For our purposes, we will detail the composition of the Information System Department where the unit responsible for managing logical access is located in the form of the Information Systems Security Department.

1.6 Physical Security Policy

Physical security refers to the application of physical and technical safeguards to prevent unlawful access to classified information.

Minimum safeguards shall be determined on the basis of a risk assessment and shall be mandatory in all places where classified information is kept or processed. Depending on their use, these places are divided into administrative zones and secure zones. The security officer must ensure that the protective measures meet the established requirements.

The physical security measures of a Datacenter depend on its size. Datacenters often contain a large amount of IT equipment (servers, switches and routers, power and cooling infrastructure and communication). This equipment can be contained in a cabinet, which is easy and simple to protect with a physical lock, or in a warehouse, where additional physical security measures (for example, access by badge, video surveillance, alarms or vigils) may be more appropriate.

1.5.1 Physical security management processes and surveillance system alerts

The first protection measure is to install cameras and security guards around the perimeter. The Datacenter entrances are monitored by cameras. Datacenters do not have glass windows, which creates additional security. However, each door represents a risk for physical security: cameras, locks and security guards are therefore a protection against this level of attack.

Video surveillance is always very useful for data centers. Closed circuit cameras with full panoramic, tilt and zoom capabilities should monitor exterior access points as well as

all interior doors and server room. Camera images must be digitally saved and archived offsite to avoid unauthorized manipulation.

Multi-Factor Authentication System

Each datacenter must follow "zero trust" security procedures that incorporate multi-factor authentication.

In addition, each access point must require at least two forms of identification or authorization to ensure that no one will be allowed to enter through security if it lacks some form of authentication.

If an attacker successfully passes through a door, the next level of physical security is a Faraday cage. The attacker cannot cross it without permission and the corresponding key. This key can be a traditional key, a code to enter into a security device, a card to scan or a biometric system. Biometric systems are the safest, but they are also the most expensive. Level 4 datacenters are always equipped with one of these systems.

Visitors to a datacenter are closely monitored, as very few people have to roam the premises. Visitors must have limited access to equipment and be accompanied by an employee. Visitors receive a badge during their visit and a signature is required in a register at each entry and exit of the premises.

1.5.2 Alarm System

In addition to the risk of property and material damage resulting from a fire or safety incident, there is also a risk of loss of operational performance and reputation.

1.6 Human Resources Management Process

The analysis of the human resources management policy allowed us to represent the management procedure as follows:

1.6.1 Roles and Responsibilities

The policy defines the responsibilities of each of the stakeholders in the procedure as well as the administrative rules they must respect.

The responsibilities are as follows:

- The business coordinator: this is the initiator of a request to create, assign, modify, or remove privileges.
- The system administrator: responsible for implementing access rights in the information system

1.6.2 Existing Approach to Human Resources Management

Based on the information gathered during interviews with the persons in charge of the management of these human resources in order to become aware of the practices applied to date in terms of granting, modifying and revoking access, we determined that the actual approach taken by CPI spa employees for access management differs from the procedure. Below are the processes as reported:

1.6.3 Management of new recruits

When a new employee is recruited within the organization, a human resources manager informs by email the administrators of systems/ networks and IT platforms within the Information Systems Department.

The administrator creates the privileges necessary for the execution of tasks on the basis of the matrices of the authorisations formalised for this purpose.

1.6.4 Management of departures

When an employee leaves (permanently or temporarily) the organization, a human resources manager informs by email the administrators of systems/ networks and IT platforms within the Information Systems Directorate.

The administrator disables all access rights to IT resources.

1.6.5 Reactivation management

When an employee resumes service within the organization, a human resources manager informs by email the administrators of systems/ networks and IT platforms within the Information Systems Department.

Administrators reactivate all access rights to previously suspended IT resources. Thus, communication on access requests is carried out manually through an exchange of emails allowing no control over the content of the request. In addition, the organization does not use any computer tools dedicated to the management of access rights and the processing of requests for granting, modifying, suspending, deleting, and reactivating access. The approach followed is supported only by office automation tools such as:

- Microsoft Office Suite: Excel, Word, etc. that are used daily for various tasks.
- A mailing tool to ensure communication between the various actors involved.

1.7 Diagnostic and Critical

The analysis of the documents and information collected as part of our project allowed us to identify the major problems. We adopted the Ishikawa cause-effect model to present the anomalies identified:

Anomaly: fire**Causes**

- Fires coming from outside.
- Lines and cables.
- Ventilation spaces.
- Heat and smoke.
- Heating, ventilation and air conditioning or HVAC equipment.
- Combustible materials.

Consequences

- **On a human level:** Datacenters have teams on site who are potentially in danger in the event of a fire, either because of flames and/or smoke emanation that can be corrosive and that decrease visibility in case of evacuation and emergency response. It is also necessary to take into account people living nearby, in the case of datacenters located in inhabited areas.
- **At the economic level:** The damage is generally divided between material (physical destruction of servers and equipment on site) and intangible (loss of professional data for hosted companies, closure of their sites and servers).
- **Environmental impact:** Datacenters are full of electronic components and rare metals, materials that pollute the environment.

Anomaly: Excessive Temperatures**Causes**

- Hot spots in a data center are defined by ASHRAE TC 9.9 as areas where air entering servers, storage systems, routers or other electronic equipment will be greater than 27°C. The rear

area of racks and hot aisle areas are not considered hot spots.

Consequences

- Hot spots can reduce reliability and damage electronic equipment due to the inability to dissipate the generated heat. Manufacturers of servers and computer equipment can justify the refusal of the warranty service due to the breach of the terms of the service contract by the presence of hot zones.

Anomaly: leaks and water damage

Causes

- Flooding may occur outside of a flood zone and in moderate precipitation, and groundwater levels may become a concern.
- Ducts, draught chambers and trenches can fill with water and become a channel for moisture.

Consequences

- Water and moisture can cause instantaneous problems such as short circuit and partial discharges.

1.5.4 Gas and Smoke Anomaly

Causes

- If the fire starts, it will be fanned by the constant supply of fresh air and fumes may spread through the duct system.

Consequences

- Smoke is one of the most common propagation vectors. It is hot and combustible. Moreover, they cause irreversible damage to electronic equipment by the soot they deposit.

Anomaly: Non-compliance with access management policy**Causes**

- Requests for access transmitted directly from the initiator to execution without going through validation. - Lack of a consistent application tool.
- Lack of awareness of information security by members of the organization. - Neglect and abandonment of best practices within the organization.

Consequences

- Failure to follow security rules while assigning, modifying, and disabling access rights.
- Grant sensitive access to persons not authorised to exploit them. - No traceability of the access request or its validation and therefore no party can be held responsible in the case of unlawful access.
- Increase the risk of illicit access to the organization's IS.

1.8 Risk Analysis

In this section, we present the results of the risk analysis we carried out based on the anomalies listed above. The objective of the latter is to corroborate the relevance of our diagnosis and to guide us in the development of the guidelines of our target system.

There are multiple methods and standards to cover all aspects of the risk management process in the company, we have based on the comparative study of R.R. CHALAL on the available

methods of risk analysis. We have therefore decided to follow the approach defined by the EBIOS method in our study.

Context study:

Objective of the study

The aim of this study is to analyse and manage the environmental risks of the A.T.C.I. mass payment system. The need expressed by the CPI spa is as follows: it wants to monitor and secure the environment housing the interbank tele clearing system ATCI, given the sensitivity of the system hosted at the site level, the areas will be supervised and secured in terms of the environment (temperature, movement...), so we will have to study all the sources of the threats related to the environment that houses the A.T.C.I mass payment system.

The ICC also wishes to ensure that security management processes in the future comply with the recommendations and best practices of the ISO 27001 and 27002 standards, as well as the ITIL and the Sarbanes-Oxley "SOX" legislation.

Scope of the study Supervision process and securing the environment housing the A.T.C.I.

According to the needs expressed by the CPI, and research that we have been able to conduct on the different standards and methods for the supervision and security of Datacenters, it agreed that the scope of our study will focus on the following processes:

- Fire Risk Management Process

The CPI wants an alert to be issued as soon as a fire is detected with information specified on it.

- Excessive Temperature Risk Management Process

The ICC wants an alert to be issued if there are changes in the temperature of the environment with specified information about these changes.

- Leak Risk and Water Damage Management Process

The ICC wants an alert to be issued if there are leaks and water damage in the environment with details of these changes.

- Gas and Smoke Risk Management Process

The CPI wants an alert to be issued if there are gases or fumes in the environment with specified information about them.

- Remote monitoring process

The CPI wants the Datacenter to be monitored 24/7.

- Compliance with Human Resources management policy

The CPI has formalized and centralized in a clear way the Human Resources management process, so that the policy implemented by it is respected.

- Access request management process (creation, revocation, and internal mobility requests). Shortcomings in access management processes will be analysed and covered during our study.

- Control of the authorization of users to acquire the requested privileges

Because 90% of all attacks are caused by human error (Kaspersky, 2020), we need to cover the risks of empowering users and the privileges they may have.

Sources of threats: To ensure the security of the information and the information system, the following criteria were used: availability, integrity, confidentiality and traceability.

Types of sources of threats	Hold or not hold	Example
natural phenomena	Yes	<ul style="list-style-type: none"> • Earthquakes • Flooding...etc.
Natural or health disaster	Yes	<ul style="list-style-type: none"> • Blackout • Internet shutdown • Fire
Untargeted virus	Yes	<ul style="list-style-type: none"> • Trojan horse
Internal human source, malicious, with low capabilities	Yes	<ul style="list-style-type: none"> • Malicious employee
Internal, malicious human source with significant capabilities	Yes	<ul style="list-style-type: none"> • Malicious manager or coordinator
Internal, malicious human source with unlimited capabilities	Yes	<ul style="list-style-type: none"> • Malicious employee with access to sensitive organizational data
External human source, malicious, with low capabilities	Yes	<ul style="list-style-type: none"> • Third-party provider

External, malicious human source with significant capabilities	Yes	<ul style="list-style-type: none"> • Contractual worker with access to the organization's platforms
External, malicious human source with unlimited capabilities	No	
Internal human source, without intent to harm, with low capacity	Yes	<ul style="list-style-type: none"> • Intern • Maintenance staff
Internal human source, without intent to harm, with significant capabilities	Yes	<ul style="list-style-type: none"> • An unscrupulous or unserious manager
Internal human source, without intent to harm, with unlimited capabilities	Yes	<ul style="list-style-type: none"> • Unscrupulous system administrator
External human source, without intent to harm, with low capabilities	Yes	<ul style="list-style-type: none"> • Staff entourage
External human source, without intent to harm, with significant capabilities	Yes	<ul style="list-style-type: none"> • Contractual participant
External human source, with no intent to harm, with unlimited capabilities	No	

Table 1.1: Risk analysis, sources of threats.

Evaluation Metrics:

The security criteria adopted: In order to ensure

the security of the information as well as the information system, the following criteria were retained: availability, integrity, confidentiality and traceability.

Criteria	Description (National Information Systems Security Agency, 2010a)
Availability	Access to information system resources must be permanent and flawless during the planned periods of use. Services and resources are accessible quickly and regularly.
Integrity	The data must be what is expected, and must not be altered in an accidental, illicit or malicious manner. Clearly, the elements considered must be accurate and complete.
Confidentiality	Only authorized people can have access to the information intended for them (concepts of rights or permissions). Any unwanted access must be prevented.
Traceability	Guaranteed that access and attempted access to the elements considered are traced and that these traces are preserved and usable.

Table 1.2: Risk analysis, evaluation metrics.

Availability scale:The following scale will be used to express security needs in terms of availability:

Ladder levels	Detailed description of the scale
More than 48 hours	The essential good may be unavailable for more than 48 hours.
Between 24h and 48h	The essential good must be available within 48 hours.
Between 4h and 24h	The essential good must be available within 24 hours.
Less than 4 hours	The essential good must be available within 4 hours.

Table 1.3: Risk analysis, availability scale.

Integrity scale: The following scale will be used to express security needs in terms of integrity:

Ladder levels	Detailed description of the scale
Detectable	The essential good may not be intact if the alteration is identified.
Mastered	The essential good may not be intact, if the alteration is identified and the integrity of the essential good found.
Integrated	The essential good must have rigorous integrity.

Table 4: Risk analysis, integrity scale.

Privacy scale:The following scale will be used to express security needs in terms of confidentiality:

Ladder levels	Detailed description of the scale
Audience	The essential good is public.
Limit	The essential good must only be accessible to staff and partners.
Reserved	The essential good must only be accessible to the (internal) personnel involved.
Private	The essential good must only be accessible to identified people who have a need to know.

Table 1.5: Risk analysis, confidentiality scale.

Traceability scale:The following scale will be used to express security needs in terms of traceability:

Ladder levels	Detailed description of the scale
Not traceable	The trace of the essential good is not recorded.
Traceable	The trace of the essential good must be saved and usable.

Table 1.6: Risk analysis, traceability scale.

Severity scale:The following scale will be used to

estimate the severity of feared events and risks:

Ladder level s	Detailed description of the scale
Negligible	The organization will overcome the impacts without any difficulty.
Significant	The organization will overcome the impacts despite some difficulties.
Severe	The organization will overcome the impacts with serious difficulties.
Critical	The organization will not overcome the impacts (its survival is threatened).
Catastrophic	The organization will not overcome the impacts (its survival is impossible).

Table 1.7: Risk analysis, severity scale.

Likelihood scale:The following scale will be used to estimate the likelihood of threat and risk scenarios:

Ladder level s	Detailed description of the scale
Extremely rare	This should never happen (again).
Rare	This should not happen (again).
Unlikely	This could happen (again).

Likely	This should (re)happen one day or another.
Frequent	This will (re)happen frequently.

Table 1.8: Risk analysis, likelihood scale.

The identified goods:The processes linked to access management included in the case of our study make it possible to process and cover all operations linked to logical user access within the organization. We will list the essential assets identified below. :

Business process	Essential processes	Essential information	Custodians
------------------	---------------------	-----------------------	------------

Remote monitoring management	Television monitoring installed in the supervision and maintenance room	<ul style="list-style-type: none"> • User guides • Documentation of materials • Personal information of users • Surveillance room access policy • List of contacts (Suppliers, Maintenance Authority) 	Business coordinator or HR
Alarm management	Monitoring of alarm equipment and maintenance	<ul style="list-style-type: none"> • User manual • Documentation of materials • List of contacts (Suppliers, Maintenance Authority) 	Business coordinator or HR
Respect for the access management policy	Implementation of processes respecting the best practices governed by	<ul style="list-style-type: none"> • Access management policy communicated to users • Process allocation and 	Business coordinator or HR

		management	
	standards mentioned previously	formalized access • Best practices related to access management	
Management process access requests	<ul style="list-style-type: none"> • Access creation request process • User access revocation process (Deprovisioning) • Processes linked to internal mobility • Processes related to user reactivation 	<ul style="list-style-type: none"> • User personal information • User organizational information (service, department, etc.) • User Organizational Roles • Authorizations of user access • User 	Business coordinator or HR

Control of authorizing users to acquire the requested privileges	Checking the user's authorization to receive the access requested upstream of its Creation	<ul style="list-style-type: none"> • User organizationa l role • Authorizations of the user • Authorization matrix the organization • User-requested access 	Business coordinat or HR
--	--	---	--------------------------

Table 1.9: Risk analysis, identified assets.

Studies of fearedevents

Feared event	Need for security	Source of threats	Impacts	Severity
Fire	Traceable	<ul style="list-style-type: none"> - Fires coming from outside. - Wires and cables. - The spaces ventilation. - Heat and smoke. - Heating, ventilation and air conditioning 	<ul style="list-style-type: none"> - teams on site who are potentially in danger - people living nearby, in the case of our Datacenter 	Catastr oph ic

		<p>or HVAC equipment.</p> <ul style="list-style-type: none"> - Combustible materials. 	<p>located in populated areas.</p> <ul style="list-style-type: none"> - physical destruction of servers and equipment on site - loss of professional data for hosted companies, closure of their sites and servers - Electronic components and rare metals, polluting materials for the environment 	
excessive temperatures	Traceable	<ul style="list-style-type: none"> - The rear area of the racks and the areas in the hot aisles 	<p>electronic equipment</p>	Critical

leaks and water damage	Limit	- flooding - The sheaths, draft chambers and trenches can be fill with water and become a pathway for moisture		Catastr oph ic
gas and smoke	Limit	- Unserious/malicious employee - Malicious intrusion into organizational systems	- Losses due to internal fraud - Judicial, disciplinary or administrative sanctions linked to non-compliance in financial results - Damage to the image of the organization	Critical
Counterfeiting of documents	Integrated	- Unserious/malicious employee - Malicious intrusion into organizational systems	- Financial losses due to internal fraud - Damage to the	Critical

			company's image	
Setting error	Traceable	Unserious employee	<ul style="list-style-type: none"> - Unauthorized access to sensitive data - Privilege escalation - Forgotten dormant accounts 	Severe
Failure to comply with legislative or regulatory obligations	Limit	Unserious employee	Judicial, disciplinary or administrative sanctions linked to non-compliance in the management of access to financial systems	Critical

Dysfunction of the activity of systems	Less than 4 hours	Malicious intrusion into organizational systems	Alteration or deletion of sensitive data	Catastrophic
Attempted digital identity theft (Increase in privilege level)	Traceable	- Malicious employee - Malicious external service provider	- Unauthorized access to sensitive data - Losses due to internal fraud - Alteration or deletion of sensitive data	Critical
Dysfunction of the activity of systems	Less than 4 hours	Malicious intrusion into organizational systems	Alteration or deletion of sensitive data	Catastrophic

<p>Attempted digital identity theft (Increase in privilege level)</p>	<p>Traceable</p>	<p>- Malicious employee - Malicious external service provider</p>	<p>- Unauthorized access to sensitive data - Losses due to internal fraud - Alteration or deletion of sensitive data</p>	<p>Critical</p>
<p>Hacking of the organization's applications</p>	<p>Less than 4 hours</p>	<p>Malicious intrusion into organizational systems</p>	<p>- Unauthorized access to sensitive data - Losses due to internal fraud - Alteration or deletion of sensitive data - Damage to the image and reputation of the</p>	<p>Catastrophic</p>

			organization	
Malicious exploitation of dormant accounts within organizational systems	Limit	- Malicious employee - Malicious external service provider - Poor management of user accounts (forgetting to deactivate)	- Alteration or deletion of sensitive data - Disclosure of Sensitive Information	Severe
Disclosure of information or theft of data	Limit	- Malicious employee - Malicious external service provider	- Damage to the company's image - Financial losses due to damage to the company reputation	Critical

Malicious use of systems by a third party provider	Traceable	- Malicious service provider - Forgotten revocation a service provider account	- Alteration or deletion of sensitive data - Disclosure of sensitive information	Severe
Access to sensitive resources by an unauthorized user	Limit	- Malicious employee - Malicious external service provider - Error in access allocation	- Alteration or deletion of sensitive data - Disclosure of Sensitive Information	Catastrophic
Propagation of malicious code between computer systems	Limit	- Malicious employee - Malicious external intrusion into security systems the company	- Alteration or deletion of sensitive data - External fraud - Damage to the company's image	Catastrophic

Table 1.10: Feared events.**1.9 Risk studies**

Risk identification we were able to establish a list of possible risks based on the feared events listed above.

Risk identified	Risk level	
	Likelihood	Severity
Fire hazard	Catastrophic	Rare
Risk of excessive temperatures	Critical	Rare
Risks of leaks and water damage	Catastrophic	Rare
Gas and smokerisks	Critical	Rare
Risk of non-detection of unreported transactions (intentionally)	Critical	Likely
Risk of non-detection of counterfeit documents	Severe	Likely
Risk of error in assigning privileges to users	Severe	Unlikely
Risk of non-compliance with Regulations	Critical	Rare
Risk of intrusion which could cause malfunctions in systems beyond 4 hours	Catastrophic	Likely
Risk linked to the theft of the digital identity of an employee of the organization (Increase in privileges)	Critical	Likely

Risk of hacking applications the organization may lead to a stoppage beyond 4 hours	Catastrophic	Likely
Risk related to forgetting and malicious exploitation of dormant accounts within the organization's systems	Critical	Unlikely
Risk related to disclosure of information or theft of data	Critical	Likely
Risk linked to non-detection of malicious use of systems by an external service provider	Severe	Unlikely
Risk linked to the non-detection of an intrusion leading to the propagation of malicious code within the organization's systems	Catastrophic	Rare

Table 1.11: Risk analysis, traceability scale.

Therefore, the risks have been classified according to their seriousness and likelihood in the following.

Grav ity	catastro phic 5		-Risk linked to the non- detection of an intrusion leading to the propagatio		-Risk of hacking of the organizati on's applicatio ns which could lead to a	
-------------	-----------------------	--	--	--	---	--

			<p>n of malicious code within the organizati on's systems -Fire hazard - Risks of leaks and water damage</p>		<p>shutdown beyond 4 hours -Risk of intrusion which could cause malfunctio ns in systems beyond 4 hours</p>	
	Catastro phic 4		<p>-Risk of non- compliance with regulation s -Risk of excessiv e temperat ures -Gas and smokeris ks</p>	<p>dormant accounts within the organizati on's systems</p>	<p>-Risk of non- detection of unreported transactio ns (intention ally) -Risk linked to the theft of the digital identity of an employee of the organizat</p>	

					ion (Increase in privilege s) -Risk related to disclosure of informatio n or theft of data	
	Seve re 3			-Risk of error in assigning privileges to users -Risk linked to non- detection of malicious use		
				systems by a external provide r		

	Significant 2					
	Negligible 1					
		1. Extremely rare	2. Rare	3. Unlikely	4. Likely	5. Frequent
		Likelihood				

Table 1.12: Risk analysis, criticality matrix.

Caption :

Negligible risks	Significant risks	Intolerable risks
------------------	-------------------	-------------------

Table 1.13: Risk analysis, integrity scale.

The risks associated with managing logical access to the organization's platforms all represent a critical starting point. Point at which any type of attack with severe or even fatal consequences on the organization can occur. Thus, we can therefore note that all the risks listed above must be prevented or treated, none of these can be neglected during the development of the solution to manage access. In what follows, we will identify the most appropriate measure for

each risk and the security needs linked to them.

1.10 Security goals

The client mainly wishes to avoid or reduce risks relating to the management and control of logical access to the organization's systems.

The following table presents the identified security objectives (the crosses correspond to the first choices, the pluses in parentheses correspond to the other accepted possibilities)

Risk	Security objective		
	Avoidance	Reduction	Socket
Fire hazard	+	x	+
Risk of non-detection of counterfeit documents	+	x	
Risk of error in assigning privileges to users	x	+	+
Risks of leaks and water damage	x		
Risk of excessive temperatures	x		
Gas and smokerisks	x		
Risk of intrusion which could cause malfunctions in systems beyond 4 hours		x	+
Risk linked to the theft of the digital identity of an employee of the organization (Increase in privileges)	x	+	+
Risk of hacking of the		x	+

organization's applications which could lead to a shutdown beyond 4 hours			
Risk related to forgetting and malicious exploitation of dormant accounts within the organization's systems	x	+	+
Risk related to disclosure of information or theft of data	+	x	
Risk linked to non-detection of malicious use of systems by an external service provider	x	+	+
Risk linked to non-detection of an intrusion leading to the propagation of malicious code within the organization's systems	+	x	

Table 1.14: Risk analysis, integrity scale.

We can therefore note that for all of the significant risks identified, the most appropriate measure to take is avoidance. These risks can be avoided by taking the necessary measures upstream, such as regular document checks, or multi-level validation as well as permanent monitoring of user accounts. As for the intolerable risks identified during this study, two of them can be avoided by an access management solution, as for the others, this same solution will make it possible to reduce them considerably.

1.10 Areas for improvement

In order to remedy the anomalies identified and the risks analyzed, it is necessary to define the main guidelines of our

project which will make it possible to strengthen security. Thus, based on the recommendations of the ISO 27001 and 27002 standards, as well as the best practices of ITIL and the SOX legislation, we have determined the main areas of improvement for an access management information system:

At the technical level

- Automate communications between alerting, monitoring and access management system components.
- Reinforcement with missing sensors.
- Implementation of an alert management and automation platform.
- Integrating AI into infrastructure components.
- Implementation of an application, to control and receive alerts from the system remotely and in real time.
- Implementation of a layer to secure interoperability between components at the device, software and platform level.
- Equip the processes for processing access requests.
- Centralize and structure access to information on authorizations and privileges.
- Computerize documents relating to access management.

At the organizational level

- Improve traceability of access and authorizations.
- Optimize the distribution of responsibilities.
- Optimize HR processing processes.

1.11 Conclusion

The analysis of the existing situation allowed us to understand current access management processes, identify anomalies and determine potential areas for improvement. This understanding then helped us develop an appropriate solution to resolve the stated problem. In addition, the risk analysis carried out strengthened our understanding of the issues of access management and its impact on the proper functioning of the organization.

In the next chapter we going to propose a solution and talk more about the tools we going to use.

Chapter II: State of the Art

2.1 Introduction

In this chapter, we shall offer a creative apparatus to boost the efficacy and safety of data centers through Raspberry Pi and Arduino technology. To do so, we will combine these adaptable pieces of equipment to come up with a robust and scalable surveillance system that can be administrated by means of one single platform. In addition, we are hoping to develop a mechanism that is able not only to monitor and protect the data center but as well send real-time notifications and perform in-depth analysis based on driving insights. This solution, thus, presents an economical strategy that allows for customization while at the same time ensuring the security and integrity of data centers as well as programming languages that we are going to utilize.

2.2 HISTORY AND EVOLUTION

Raspberry Pi was born out of a concern about low enrollment levels in computer science classes in 2006. Eben Upton from the University of Cambridge along with his team consisting of engineers and computer scientists came together to form the Raspberry Pi Foundation aimed at reversing this trend through creating a cheap, accessible computing platform. The Raspberry Pi was an idea that materialized into a device that can be programmed and is both tough and fun for kids. First, there was the Raspberry Pi Model B, then series of other models followed each improving on performance and capabilities.[1]

2.3 EDUCATION AS THE MAIN FOCUS OF THE FOUNDATION

The foundation is part of a consortium running the National Centre for Computing Education in England which supports teachers and provides extensive resources for computing

education. Through coding clubs, online resources and competitions that have attracted millions from over one hundred countries, children have been drawn back into coding by these devices. [2]

2.4 WHAT IS RASPBERRY PI?

Raspberry Pi is a range of small, low-cost single-board computers developed by the Raspberry Pi Foundation; it's not just about their size but rather what they can do as little things; A charity organization born out of Britain

These credit card-sized computers are designed to promote computer science education, facilitate learning about programming and electronics, and enable a wide range of DIY projects and applications[3]

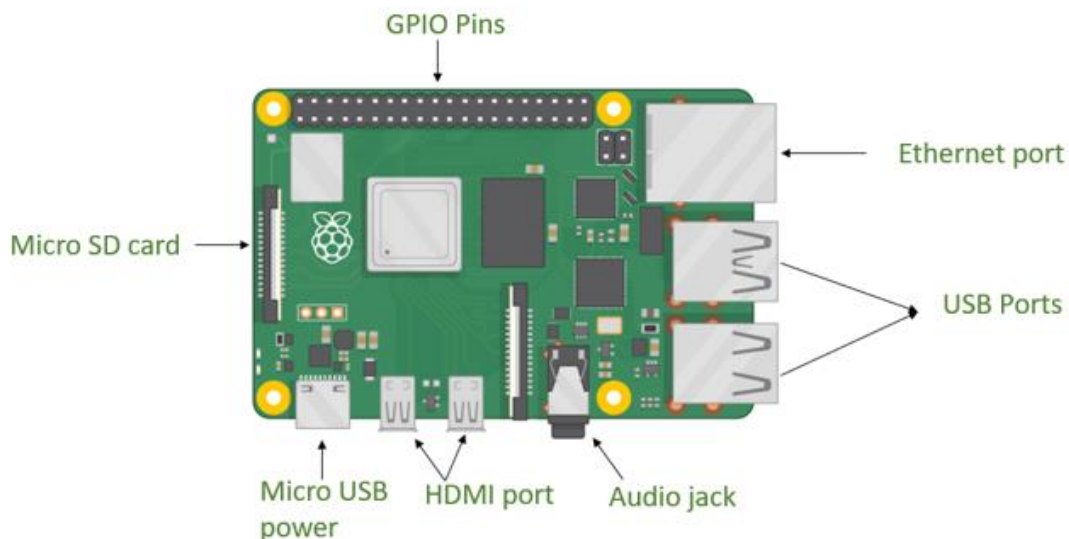


Figure 2.1: the architecture of Raspberry Pi

2.5 Variants and Models of Raspberry Pi

When it comes to single board computers, there is no doubt that Raspberry Pi has become a major name in the industry. There are different models that one can choose from but this may be confusing for some especially the beginners and even some experts who might find it difficult to understand which option is the best one for them such as Raspberry Pi Zero, Raspberry Pi 2 model B, Raspberry Pi 3, Raspberry Pi 4 Model B, and the latest raspberry pi 5.

We specifically selected a Raspberry Pi 4 Model B for our situation.

****For more informations about the Raspberry Pi check annex 1****

2.5 Results and Accomplishments:

- Development of a workable prototype.
- Skills in electronics improvement programming system integration.
- Real world implementation via open source contributions to community.

2.6 History of Arduino

The Arduino Project was started at the Interaction Design Institute in Ivrea, Italy. At first, the students used a \$50 BASIC Stamp microcontroller. As a part of his master's thesis project at IDII in 2004, Hernando Barragán developed Wiring under the direction of Massimo Banzi and Casey Reas. Among other things, Casey Reas is responsible for creating Processing software along with Ben Fry. The main purpose of this initiative was to provide cheap and simple tools for non-technical people who want to create digital projects. It consisted of an IDE based on Processing language, a printed

circuit board with ATmega128 microcontroller and library functions that make programming microcontrollers easier. In 2005, Massimo Banzi expanded Wiring's scope by adding support for cheaper ATmega8 microcontrollers together with David Mellis and David Cuartielles both IDII students. This fresh release which was based on Wiring was then called Arduino.

The early team members of Arduino included Massimo Banzi, Tom Igoe, Gianluca Martino, David Cuartielles and David Mellis.

After the release of this platform derivatives were released into open source community at lesser weights or less costly models. By my estimation in mid-2011 there had been over 300 thousand official Arduinos sold[4]

2.7 What is Arduino?

Arduino is an open source platform for building electronic projects. A physical programmable circuit board (often referred to as a microcontroller) and a piece of software (an Integrated Development Environment, or IDE,) that runs on your computer are the two parts that make up Arduino; this allows you to write and upload computer code to the physical board.

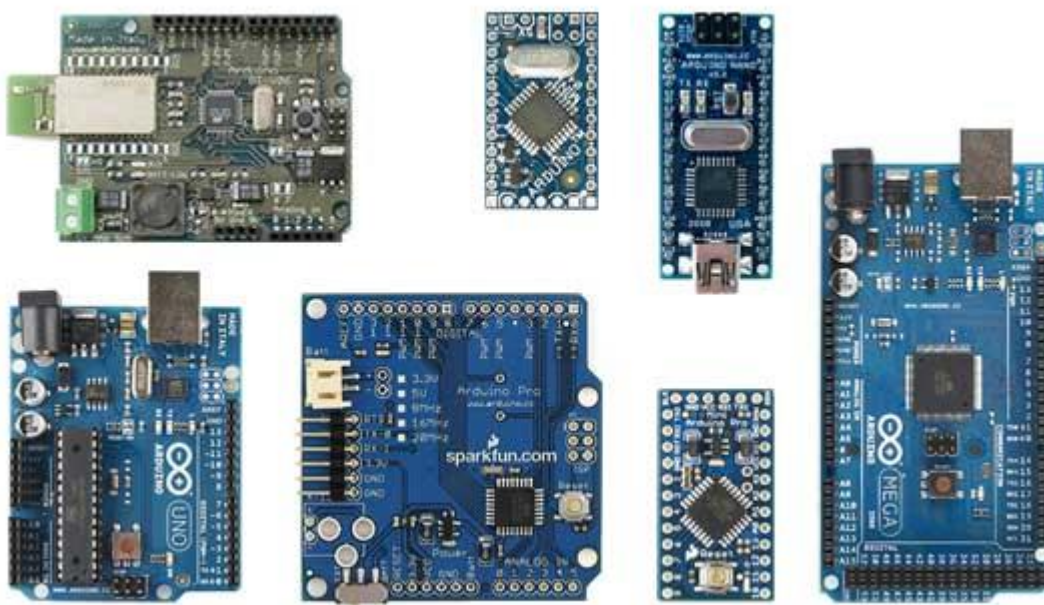


Figure 2.2: Arduino and its forms

The reason why Arduino has become so popular among people who are just starting out with electronics should not be underestimated. Unlike most other programmable circuit boards, Arduino does not require any separate hardware (a programmer) to load new code onto the board - all you need is a USB cable. With Arduino's IDE, learning programming becomes simpler because it uses a simplified version of C++. Lastly, Arduino comes with a standard form factor which breaks out the functions of the micro-controller into more accessible package. [5]

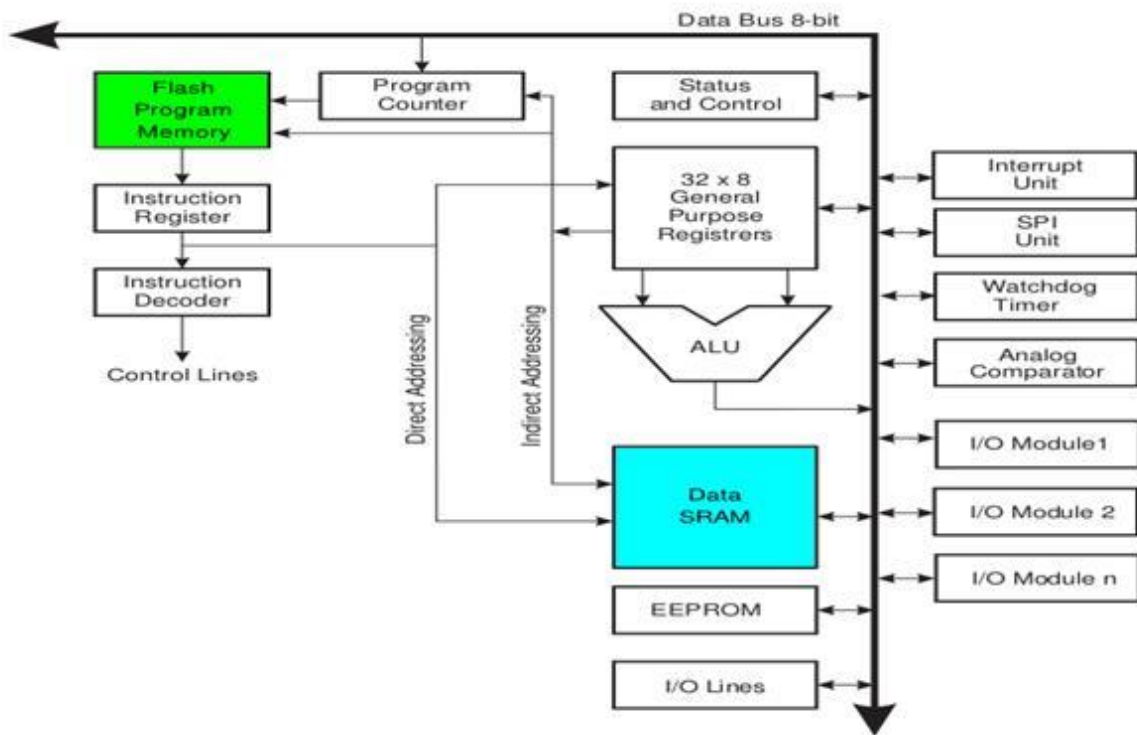


Figure 2.3: Arduino Architecture

For more informations about the Arduino check Annex 2

2.10 The difference between Raspberry pi and arduino

On the one hand, Raspberry Pi is a very popular board just like Arduino for electronic projects but they serve different needs. Here are some key differences between them:

2.10.1 Processing Power:

- Raspberry Pi - it is much powerful than the other one as it has General Purpose CPUs running at 1.6GHz (model depending) therefore making it suitable for complex tasks such as data analyses and computer visions.
- Arduino - with less power, this microcontroller runs at an average speed of 16 MHz. It can be ideal for simple task such as reading sensor or controlling motors.

2.10.2 Operating System:

- Raspberry Pi - It runs the full Linux OS that allows you to install all kinds of software and programs.
- Arduino - Not a traditional OS user, instead, programming environments are simplified to upload codes which are specifically designed on Arduino board.

2.10.3 Applications:

- Raspberry Pi: This is useful in applications that need a computer-like experience like robotics using complex control, data logging weather monitoring systems and media center.
- Arduino: Physical interactions with the environment focused projects are most suited here, ranging from LED control to some simple robot designs and collecting outputs from sensors.

2.10.4 Besides:

- Affordability: Usually, Raspberry Pi is more expensive than Arduino boards.
- User Friendliness: Arduino is a lot easier to pick up for beginners because it provides a simple programming

environment. While Raspberry Pi offers more flexibility, the learning curve is steeper.[6]

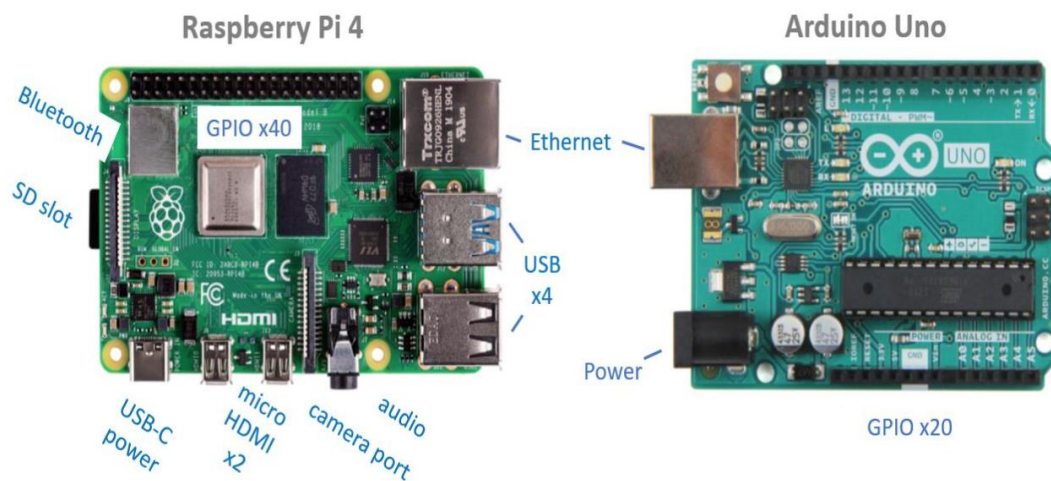


Figure 2.4: Raspberry pi 4 vs Arduino Uno

2.11 Overseeing Framework

We have developed an advanced Bank CPA platform that utilizes the power of Raspberry Pi and Arduino Technics in order to create intelligent and reliable systems. In this case, this platform combines several components and is capable of providing full-functionality such as live monitoring, data management, access control and automated notification system. Also, to construct an efficient web application that would be intuitive and resilient we used HTML, CSS, JavaScript Python Flask SQLite.

2.12 Main Achievements & Technologies

2.12.1 Integration between Raspberry Pi and Arduino:

Our platform takes advantage of the best features in both Raspberry Pi and Arduino so as to achieve a blend of processing capability with precise hardware control. The central processing unit that executes complex calculations; processes data; handles network communications; serves as the interface for all other functions is the Raspberry Pi hardware

platform. Real-time control and monitoring of hardware components are done by Arduino interacting with sensors and actuators due to its versatility.

2.12.2 Web Interface through HTML/CSS/JS:

We constructed our platform's user interface using HTML, CSS, and JavaScript to give it a seamless and interactive feel. HTML arranges the content, CSS adds visual interest with styling; while JavaScript provides dynamics that make the platform very responsive and easy to use. It is an online interface that enables users to access the platform from any device with internet connection.

2.12.3 Backend Development with Python and Flask:

The backend of our platform was built using python and flask which is a light weight web framework. The simplicity of python and its readability makes development faster as flask has all the tools needed for building powerful web applications. Combining them makes server-side operations such as data processing, user authentication, interacting with databases more efficient.

2.12.4 Database Management with SQLite:

Our database management system on this platform uses SQLite thus making it light in terms of weight but very effective regarding data storage and retrieval processes. This suitability comes from SQLite having few requirements in terms of setting up, being able to efficiently handle complex queries among other things. It keeps detailed logs of all sensor data and system activities thus enabling accurate performance analysis leading into problem solving or decision making.

2.12.5 Real-Time Monitoring and Control:

Our platform provides a real-time dashboard that gives information into the state and performance of all connected Raspberry Pi and Arduino components. This includes monitoring

a range of environmental parameters, health system metrics and operational statuses. The interface allows for immediate adjustments and control as to ensure that the system can adapt swiftly to any changes or issues.

2.12.6 Advanced Access Control:

Security is a paramount concern, and our platform includes a sophisticated access control mechanism. This system restricts access to authorized personnel only, using multi-factor authentication and encryption protocols to safeguard sensitive data and operations. Access logs are maintained in order to monitor entry points for audit purposes which further enhance security.

2.12.7 Automated Email Notifications:

We need to know about the status of the system at all times so we have implemented an automated email notification system. This system sends alerts whenever predefined events or anomalies occur such as when sensor thresholds are exceeded, hardware malfunctions or unauthorized access attempts are made among others. These notifications allow us to take prompt action thus minimizing downtime while maintaining integrity of the system.

2.13 Use Cases and Benefits

2.13.1 Operational Efficiency:

Moreover, the real-time monitoring and control capabilities of this platform yield operational effectiveness as it offers instant feedback and control of system apparatuses. Hence, it also reduces time required for issue identification, resolution thus providing seamless operations.

2.13.2 Security and Compliance:

Our platform has advanced access controls implemented with comprehensive logging that ensures only authorized staff can

access sensitive areas of the system. It prevents unauthorized entry while ensuring that all regulatory standards are being adhered to.

2.13.3 Proactive Maintenance:

This is facilitated by automated email notification that informs us in advance about any possible issues before they become critical ones. In so doing, a predictable approach helps to maintain continuity in operations and mitigates against unforeseen failures.

2.13.4 Scalability:

The design of our platform takes into account scalability such that additional components and sensors may be easily incorporated when necessary. This allows Bank CPA to grow the system without major modifications which would otherwise render it useless within a short period of time. [7]

2.14 Conclusively

For Bank CPA, our custom-built platform is a major leap forward in the adoption of Raspberry Pi and Arduino technologies as full systems. we made it an efficient, reliable and also scalable and secure implementation by integrating real-time monitoring, robust database management, secure access control mechanism, automated notifications; and a user friendly web interface developed using HTML, CSS JavaScript Python Flask SQLite. This move underscores our dedication to incorporating state-of-the-art technology into superior solutions responsive to the ever-changing requirements of Bank CPA.

Chapter III: Conception and Realisation

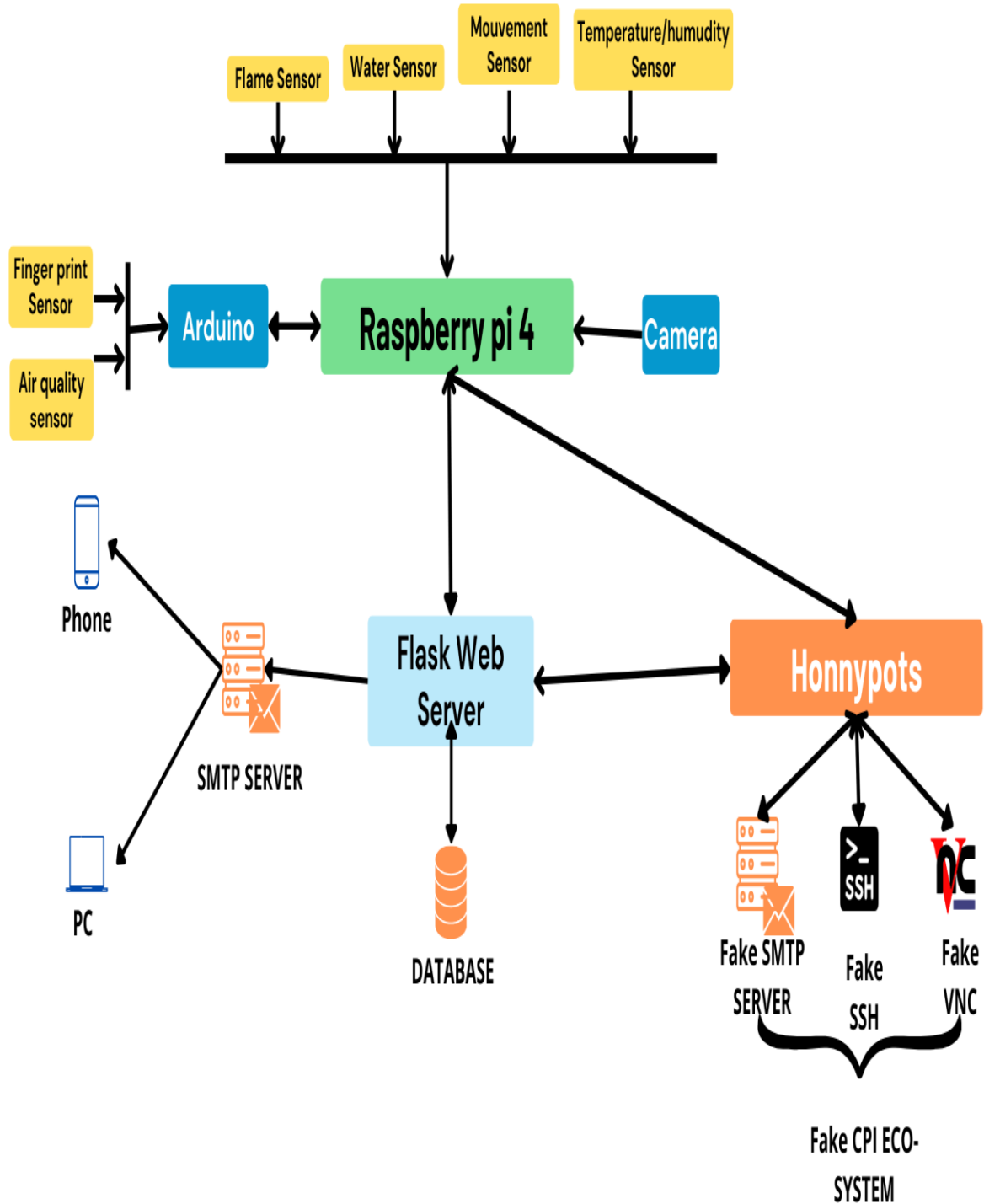


Figure 3.1: Overview of the whole process

3.1.Overview:

This chapter presents a practical implementation of the system that we have designed for intelligent control and monitoring of CPI Spa Datacenter. We will look at hardware installation and configuration, software development, integration of components and completed tests and validations. In addition to this, solution execution results will be provided with an analysis based on the goals and standard data center management approaches. This chapter aims at; Describe the hardware installation and configuration: In this sub-section, we shall provide step-by-step guidelines on how to install and configure data center management hardware. This entails selection of equipment, proper placement, as well as making connections necessary to ensure that various constituents of the system in use communicate well with each other. Describe software development: The following section concentrates on software development that facilitates surveillance and management of data centers. Major areas of focus include programming languages used, frameworks employed in the softwares creation process among others that are important such as real-time monitoring capabilities alerts as well as report generation systems through which users can effectively track events happening within their area of jurisdiction. Describe integration and deployment process: A description will then follow as to how different parts of both hardware and software were mixed together for this purposeThis involves establishing interfaces, handling dependencies, and compatibility of the system's myriad components. Also involved will be a description of putting the solution into production and other aspects like recovery procedures as well as backup procedures and how it is deployed inside datacenter environment. Analyzing system performance: We shall show what happens when we implement the solution. In this analysis we will compare the system's outcomes with goals used to develop it and with

traditional methods of managing datacenters. Assessing the cost-effectiveness, reliability and benefits of a solution in terms of reduced costs, increased security and operational optimization are done at this stage. By following this plan, we will have an overall picture on how C.P.I spa Datacenter implements its intelligent management & surveillance solutions for efficient operations despite several challenges faced that can guarantee smooth running of all activities. In addition to that, results obtained within such frameworks would also provide clear evidence for better adoption than conventional ways.

3.2 Integration and Implementation:

3.2.1 Strategic Alignment and Integration of Surveillance Units:

3.2.1.1 Positioning Strategically:

The strategic positioning of the sensors should be chosen based on the variables to be measured (temperature, humidity, movement, etc.).

Installation of Capteurs: Verify that the sensors are mounted at the proper heights and locations for precise measurements (e.g., keep temperature sensors out of direct sunlight).

3.2.1.2 Linking the Surveillance Units:

Connect each sensor to the Raspberry Pi's GPIO boards in accordance with the respective cable routing schematics.

Electronic Alimentation: Ensure steady power supply for every Raspberry Pi and sensor.

Réseau: Pour assurer la communication avec les serveurs et d'autres unités, connectez les Raspberry Pi au réseau local via Ethernet ou Wi-Fi.

3.2.2 Configuration of the sensors:

There are many steps involved in configuring sensors on a Raspberry Pi 4 Model B, including physical connection, software and pilot installation, and parameter setup. The specific steps vary depending on the sort of capteur you are using. Following is a general guide:

3.2.2.1 Physical connection:

Determine which Raspberry Pi GPIO (General Purpose Input/Output) connectors are available. Consult the Raspberry Pi brochage schematic to determine the proper location of the broches.

Use Dupont cables or solder wire to connect the sensor to the Raspberry Pi. Make sure you are connecting the sensor's pins to the Raspberry Pi's appropriate GPIO pins. Connect the sensors that require a separate supply of power to the appropriate power source.

3.2.2.2 Installation of software and pilots:

For most sensors to work with the Raspberry Pi, special drivers and software are required. These pilots and software are frequently available from the sensor manufacturer. Download and install the software and pilots required for your capture device. Respect the installation guidelines that the manufacturer has provided.

Installing other libraries or Python modules may also be necessary in order to communicate with the capteur.

3.2.2.3 Setting up the parameters:

After installing the drivers and software, you must set the camera's parameters. This may need the use of a configuration tool or the alteration of configuration files.

Consult the software's or the capteur's instructions to learn how to set up the specific parameters for your device.

The current parameters include the sensor's address, measurement range, sampling rate, and calibration.

3.2.2.4 Testing the recorder:

-After the setting is complete, you must test the sensor to make sure it is operating correctly.

-To read the sensor's data, run a test program or make use of a command line interface.

- If necessary, adjust the parameters to obtain precise readings.

3.3- Hardware Installation and Configuration

3.1 Setting up and adjusting the Raspberry Pi 4 Model B:

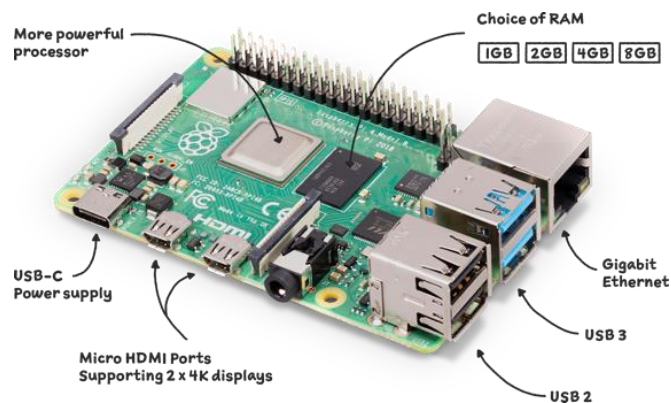


Figure 3.2:Raspberry Pi 4 Model B

By installing and configuring the Raspberry Pi 4 Model B, you may build a strong and adaptable mini-computer for a variety of uses. To get started, take these actions:

3.2.1 Steps for installation:

1. Setting up the MicroSD card:

- Get the Raspberry Pi OS operating system (previously known as Raspbian) from <https://www.raspberrypi.com/software/>.
- Use a tool like Raspberry Pi Imager (<https://projects.raspberrypi.org/en/project/imager-install>) to flash the OS image onto the microSD card [8]

2. Device connection:

o Attach the Raspberry Pi to a power outlet and the micro-USB power.

- Place the microSD card into the Raspberry Pi's microSD card slot.
- Attach the Raspberry Pi and a screen using the HDMI wire.
- Use the USB ports to connect the Raspberry Pi to the keyboard and mouse.

3. The Raspberry Pi was launched:

- Switch on the Raspbian Pi.
- The operating system for Raspberry Pis launches automatically.
- To finish the basic setup, which includes setting up a user account and connecting to the Wi-Fi network, follow the directions on the screen.

3.3.2.2 Setup and application:

Updates: Please execute the directive

- `sudo apt update`

- `sudo apt upgrade` pour maintenir votre système à jour
- Software installation:
- Install any additional software with the help of the command
- `Sudo apt install <nom_du_logiciel> [8]`

3.3.2 Installation and Setup of the Water Sensor:

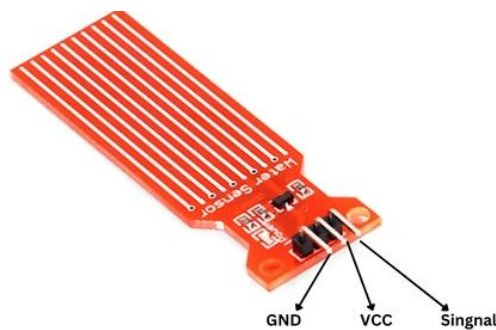


Figure 3.3:Water Sensor

This water analog cap operates on the principle of measuring the size of water droplet traces across a line using a series of parallel wires exposed to the water quantity to simulate the second plasticity based on the analog output values of the third-power, low-consumption, high-sensitivity cap directly connected to a microprocessor or other logical circuits. This educational module uses printed tracks to provide an analog tension based on water level. At the highest level, the sensor outputs "97%" and at the lowest level, it outputs "0%".[9]

3.3.2.1 Connecting the water sensor

Three broches are often present on a water sensor: VCC, GND, and DATA (or OUT). Here's how to attach them to your Raspberry Pi:

1. VCC: Attach to 3.3V.
2. GND: Attach the Raspberry Pi to GND.
3. DATA (OUT): Attach to a GPIO of your choice.

3.3.2.2 Installing the required libraries:

To install the RPi.GPIO library type `sudo apt install python3-rpi.Gpio` [10]

3.3.2.3 Write and run the Python code:

Create a Python file to read the sensor's data:

Launch a terminal and create a file named `water_sensor.py`

```
1 import RPi.GPIO as GPIO
2 import time
3 import requests
4 import json
5 import threading
6
7 def post_data(sensor_name,sensorvalue):
8     url = 'http://127.0.0.1:5000/sensorestatus' # Replace this with your API endpoint
9
10    # JSON data to be sent
11    data = {
12        "status": "ON",
13        "sensorname": sensor_name,
14        "sensorvalue":sensorvalue
15    }
16
17    # Convert the data to JSON format
18    json_data = json.dumps(data)
19
20    # Set the appropriate headers
21    headers = {'Content-Type': 'application/json'}
22    try:
23        # Make the POST request
24        response = requests.post(url, data=json_data, headers=headers)
25
26        # Check the response status
27        if response.status_code == 200:
28            print("")
29        else:
30            print("Failed to post data. Status code:", response.status_code)
31    except:
32        pass
```

Figure 3.4: Python for `water_sensor.py`

3.3.3 Installing and configuring the DHT11 temperature and humidity sensor:

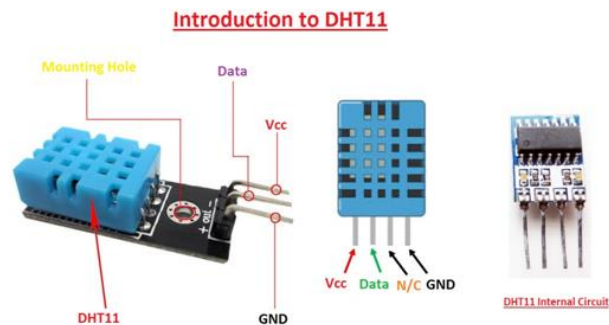


Figure 3 6:temperature and humidity sensor

An electronic device used to measure relative humidity and temperature is the DHT11 capteur. This widely used and reasonably priced sensor is frequently employed in Internet of Things (IoT) projects including bricolage, robotics, and data collection. Here is a more thorough explanation of its features and capabilities:

Features of the DHT11 sensor:

The measurement of temperature:

Temperature range: 0 to 50°C

Accuracy: $\pm 2^\circ\text{C}$

Resolution: 1°C

Measurement of humidity:

Measurement point: 20 to 90% relative humidity

Accuracy: $\pm 5\%$ relative humidity

Resolution: 1% relative humidity

User interface:

The DHT11 communicates with microcontrollers like the Raspberry Pi, Arduino, and others via a digital interface.

It communicates using just one data brooch.

The relatively straightforward steps involve installing and configuring a DHT11 sensor for automated datacenter management and monitoring using a Raspberry Pi 4. [11]

3.3.3.1 Step 1: Install the necessary tools:

- Install pip and other useful tools:

```
sudo apt install python3-pip python3-de
```

3.3.3.2 Step 2: activate the GPIO interfaces.**1. Configure the GPIOs:**

```
sudo apt install python3-pip python3-de
```

- To configure Raspberry Pi, navigate to the menu and select "Configuration".
- In the "Interfaces" menu, enable "GPIO" and "I2C" (if your sensor supports I2C).

3.3.3.3 Step 3: Installing the required libraries

```
sudo pip3 install RPi.GPIO
```

1. Install the Raspberry Pi GPIO library:

- The RPi.GPIO library is commonly used for manipulating GPIO.

1. Install sensor-specific libraries:

- For DHT sensors (e.g. DHT11, DHT22), use Adafruit_DHT.

```
sudo pip3 install Adafruit_DHT[12]
```

- For I2C sensors (e.g., BMP180), use Adafruit-BMP:

```
sudo pip3 install Adafruit-BMP[12]
```

3.3.3.4 Step 4: Connecting the Captor**1. DHT11 sensor (temperature and humidity):**

- Connections:

Connect VCC to 3.3V

GND to GND

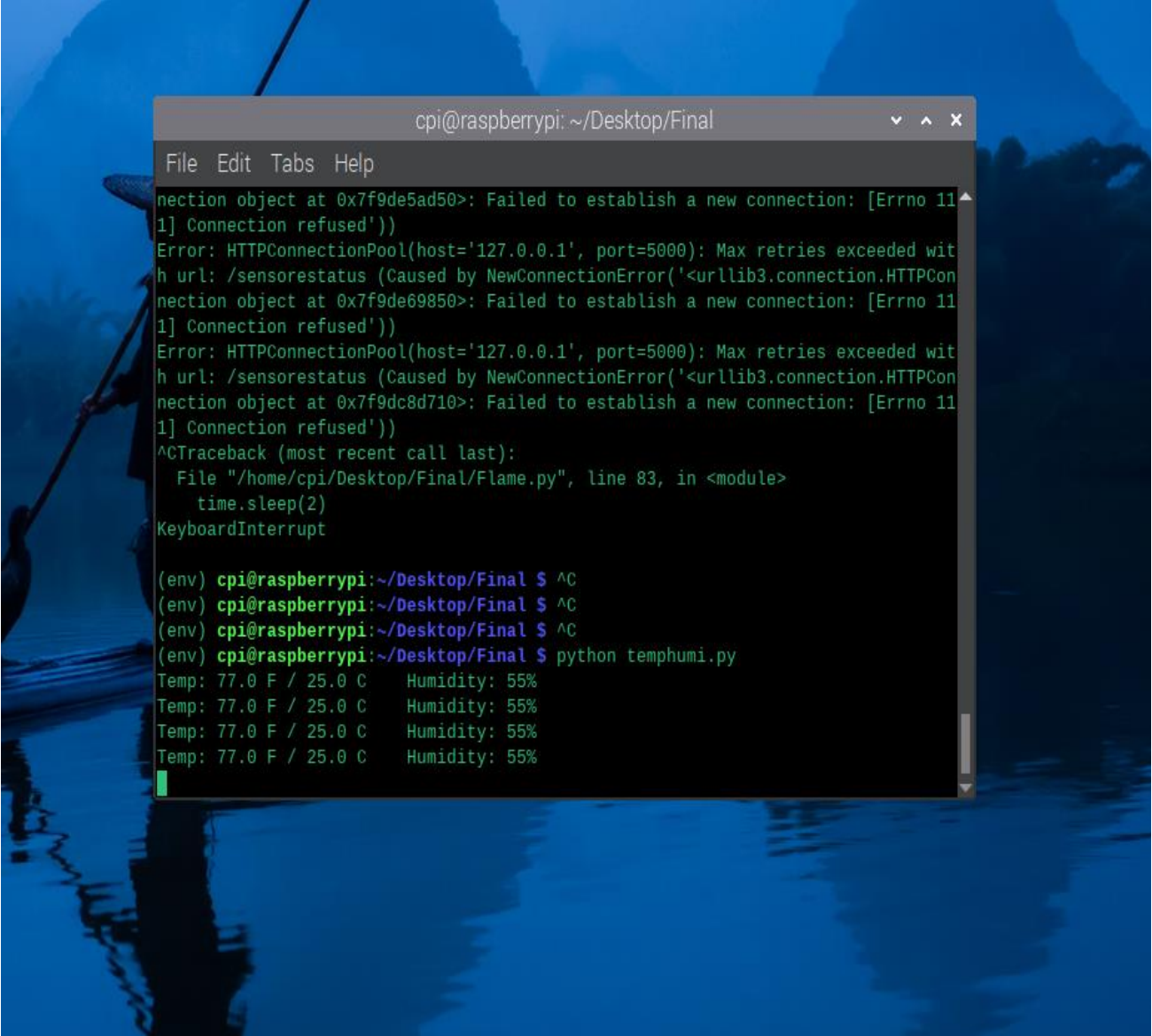
Data to GPIO 4 [13]

3.3.3.5 Step 5: Everything corresponds to the program below, including the role of each instruction.

Example code for reading the captures

```
1  # SPDX-FileCopyrightText: 2021 ladyada for Adafruit Industries
2  # SPDX-License-Identifier: MIT
3
4  import time
5  import adafruit_dht
6  import microcontroller
7  import requests
8  import json
9  import threading
10
11 def post_data(sensor_name,sensorvalue):
12     url = 'http://127.0.0.1:5000/sensorestatus' # Replace this with your API endpoint
13
14     # JSON data to be sent
15     data = {
16         "status": "ON",
17         "sensorname": sensor_name,
18         "sensorvalue":sensorvalue
19     }
20
21     # Convert the data to JSON format
22     json_data = json.dumps(data)
23
24     # Set the appropriate headers
25     headers = {'Content-Type': 'application/json'}
26     try:
27         # Make the POST request
28         response = requests.post(url, data=json_data, headers=headers)
29
30         # Check the response status
31         if response.status_code == 200:
```

Figure 3.7: Example code for reading the captures temperature and humidity python temphum.py



```
File Edit Tabs Help
connection object at 0x7f9de5ad50>: Failed to establish a new connection: [Errno 111] Connection refused'))
Error: HTTPConnectionPool(host='127.0.0.1', port=5000): Max retries exceeded with url: /sensorestatus (Caused by NewConnectionError('<urllib3.connection.HTTPConnection object at 0x7f9de69850>: Failed to establish a new connection: [Errno 111] Connection refused'))
Error: HTTPConnectionPool(host='127.0.0.1', port=5000): Max retries exceeded with url: /sensorestatus (Caused by NewConnectionError('<urllib3.connection.HTTPConnection object at 0x7f9dc8d710>: Failed to establish a new connection: [Errno 111] Connection refused'))
^CTraceback (most recent call last):
  File "/home/cpi/Desktop/Final/Flame.py", line 83, in <module>
    time.sleep(2)
KeyboardInterrupt

(env) cpi@raspberrypi:~/Desktop/Final $ ^C
(env) cpi@raspberrypi:~/Desktop/Final $ ^C
(env) cpi@raspberrypi:~/Desktop/Final $ ^C
(env) cpi@raspberrypi:~/Desktop/Final $ python temphumi.py
Temp: 77.0 F / 25.0 C Humidity: 55%
Temp: 77.0 F / 25.0 C Humidity: 55%
Temp: 77.0 F / 25.0 C Humidity: 55%
Temp: 77.0 F / 25.0 C Humidity: 55%
```

Figure 3 8:python temphum.py Dans le terminal

The Python script continuously reads temperature and humidity data from the DHT11 sensor and displays it in the terminal. You may modify the script to save the data to a file, send it to a web server, or use it to control activities based on

environmental

circumstances.

3.3.4 Installation and configuration of the Motion Capture MH-SR602:



Figure 3.9: Figure 3.8: Motion Capture

The MH-SR602 motion sensor is a passive infrared (PIR) sensor that detects the presence and movement of objects that emit heat, such as humans and animals, in its field of view. It is known for its low energy consumption and ability to operate with a low-tension supply. Stages of installation:

3.3.4.1 Step 1:Capteur connection:

The MH-SR602 motion sensor has three pins: VCC, GND, and OUT. Here's how to attach them to your Raspberry Pi:

1. Connect VCC to 5V
2. Connect to GND
3. Connect to your preferred GPIO

3.3.4.2 Step 2:Installation of the library:

Execute the following command to install the RPi.GPIO library:

```
sudo apt install python3-rpigpio [14]
```

3.3.4.3 Step 3: Write and execute the Python code

Add the following code to the file:[15]

```
1 import RPi.GPIO as GPIO
2 import time
3 import requests
4 import json
5 import threading
6
7 def post_data(sensor_name,sensorvalue):
8     url = 'http://127.0.0.1:5000/sensorestatus' # Replace this with your API endpoint
9
10    # JSON data to be sent
11    data = {
12        "status": "ON",
13        "sensorname": sensor_name,
14        "sensorvalue":sensorvalue
15    }
16
17    # Convert the data to JSON format
18    json_data = json.dumps(data)
19
20    # Set the appropriate headers
21    headers = {'Content-Type': 'application/json'}
22    try:
23        # Make the POST request
24        response = requests.post(url, data=json_data, headers=headers)
25
26        # Check the response status
27        if response.status_code == 200:
28            print("")
29        else:
30            print("Failed to post data. Status code:", response.status_code)
```

Figure 3.10: Example code for reading the captures **Motion Capture**

Execute the Python script:

In the terminal, run the script:

3.3.5 Installation and configuration of the MH-Sensor series:

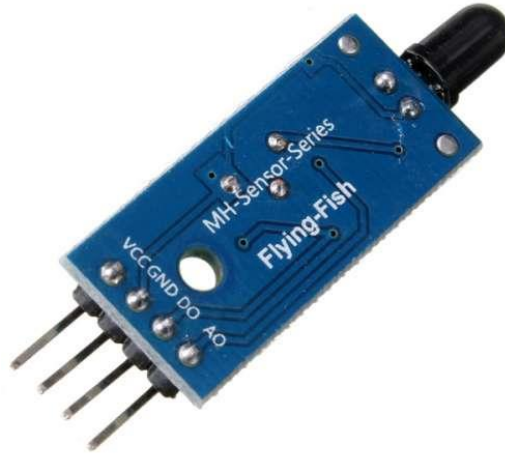


Figure 3.12: MH-Sensor series

The ST060 flame sensor is an electronic device used to detect the presence of fire in an environment. It is commonly used in fire safety systems, firefighting robots, and other flame detection applications.

3.3.5.1 Step 1: Connect the ST060 flame sensor.

The ST060 flame sensor typically has three connections: VCC, GND, and DO (Digital Output). Here's how to attach them to your Raspberry Pi:

1. Connect VCC to 3.3V (broche 1 on the Raspberry Pi).
2. Connect to GND (Broche 6 on the Raspberry Pi).
3. Connect to your preferred GPIO, such as GPIO 17 (broche 11 on Raspberry Pi). [16]

3.3.5.2 Step 2: Installing the required libraries

1. Install the Adafruit_DHT library.

```
pip3 install Adafruit_DHT [17]
```

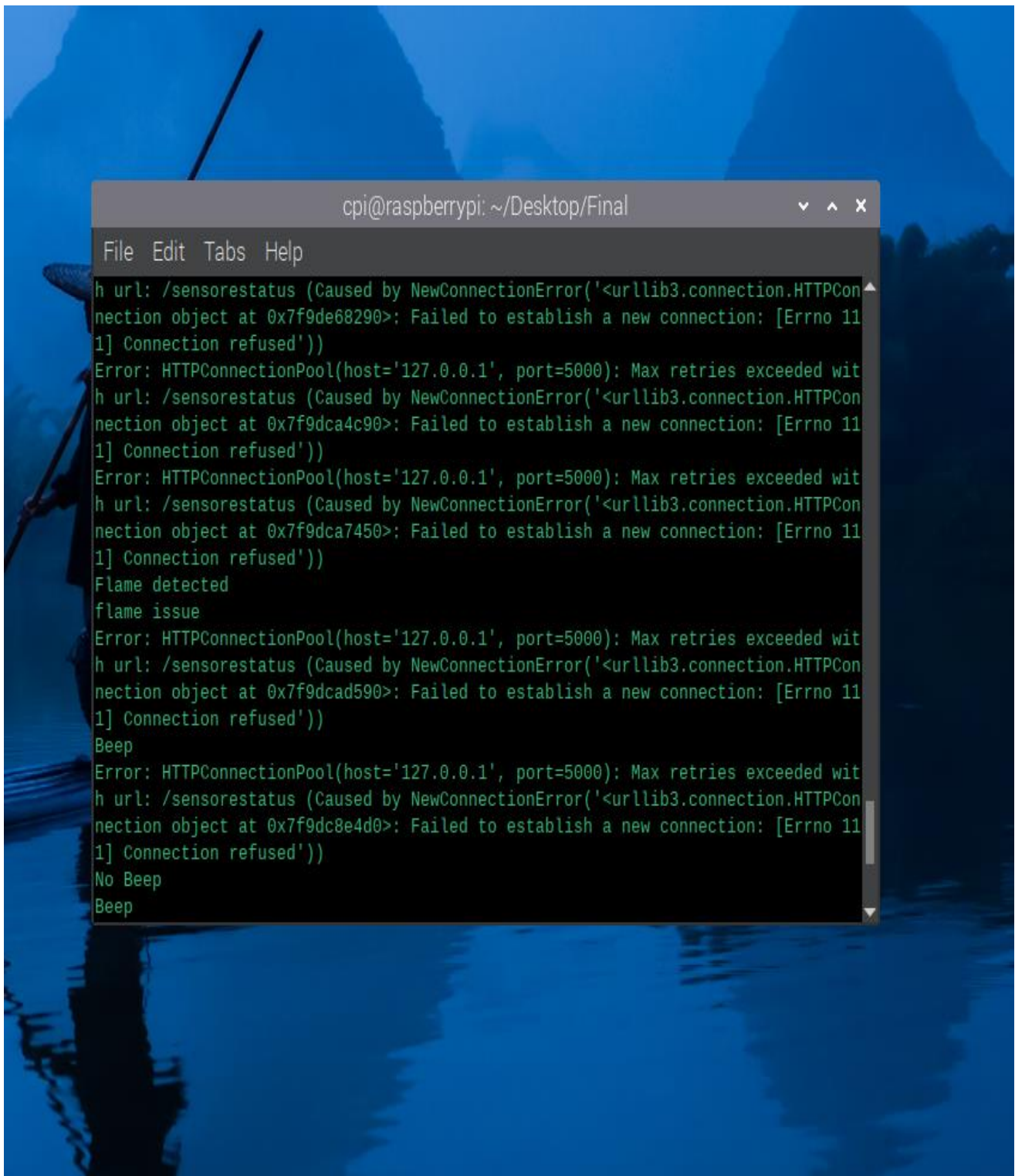
3.3.5.3 Step 3: Write and execute the Python code

1. Create a Python file to read the capteur data:

```
1 import RPi.GPIO as GPIO
2 import time
3 import subprocess
4 import requests
5 import json
6 import threading
7
8 def post_data(sensor_name,sensorvalue):
9     url = 'http://127.0.0.1:5000/sensorestatus' # Replace this with your API endpoint
10
11     # JSON data to be sent
12     data = {
13         "status": "ON",
14         "sensorname": sensor_name,
15         "sensorvalue":sensorvalue
16     }
17
18     # Convert the data to JSON format
19     json_data = json.dumps(data)
20
21     # Set the appropriate headers
22     headers = {'Content-Type': 'application/json'}
23     try:
24         # Make the POST request
25         response = requests.post(url, data=json_data, headers=headers)
26
27         # Check the response status
28         if response.status_code == 200:
29             print("Data posted successfully.")
30         else:
31             print("Failed to post data. Status code:", response.status_code)
```

Figure 3.13: Example code for reading the captures MH-Sensor

2. Open a terminal and create a file called `flame.py`.



```
File Edit Tabs Help
h url: /sensorestatus (Caused by NewConnectionError('<urllib3.connection.HTTPCon
Error: HTTPConnectionPool(host='127.0.0.1', port=5000): Max retries exceeded wit
h url: /sensorestatus (Caused by NewConnectionError('<urllib3.connection.HTTPCon
Error: HTTPConnectionPool(host='127.0.0.1', port=5000): Max retries exceeded wit
h url: /sensorestatus (Caused by NewConnectionError('<urllib3.connection.HTTPCon
Error: HTTPConnectionPool(host='127.0.0.1', port=5000): Max retries exceeded wit
Flame detected
flame issue
Error: HTTPConnectionPool(host='127.0.0.1', port=5000): Max retries exceeded wit
h url: /sensorestatus (Caused by NewConnectionError('<urllib3.connection.HTTPCon
Beep
Error: HTTPConnectionPool(host='127.0.0.1', port=5000): Max retries exceeded wit
h url: /sensorestatus (Caused by NewConnectionError('<urllib3.connection.HTTPCon
No Beep
Beep
```

Figure 3.14:python `flame.py` Dans le terminal

Explication of Code:

- The script imports the serial library and opens the appropriate serial port for the capteur.
- Prints the captured data to the console.
- The script pauses for a second before reading the data again.

3.3.6 Installation and setup of the MQ135 capteur with Arduino Uno:

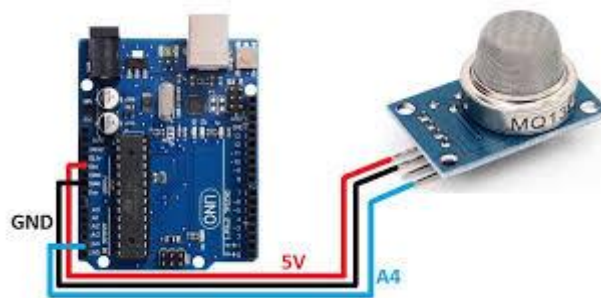


Figure 3.15: the MQ135 capteur with Arduino Uno

The MQ135 is a semi-conductor gas sensor that is commonly used to detect the presence of gases such as carbon monoxide (CO), ammonia (NH₃), ethanol (C₂H₅OH), and smoke. It is simple to use and may be used for a variety of applications, including gas detectors, ventilation systems, and air quality monitors.[18]

Stages of installation:

3.3.6.1 Step 1: Connection Diagram:

1. Connect the capteur's VCC to the Arduino's 5V pin.
2. Connect the GND of the capteur to the GND of the Arduino.
3. Connect the capteur's A connector to an Arduino analog connector (e.g., A0).[19]

3.3.6.2 Step 2: Write and upload the Arduino code

1. Open the Arduino IDE.
2. Write the code. Create a new sketch and add the following code.
3. Connect the Arduino Uno to your computer via USB cable and transfer the program by clicking "Upload" in the Arduino IDE.

Explication of Code:

- The script imports the pyserial library and opens the corresponding serial port.
- Prints the captured data to the console.
- The script pauses for a second before reading the data again.

3.3.7 Installation and configuration of DY50 digital fingerprint

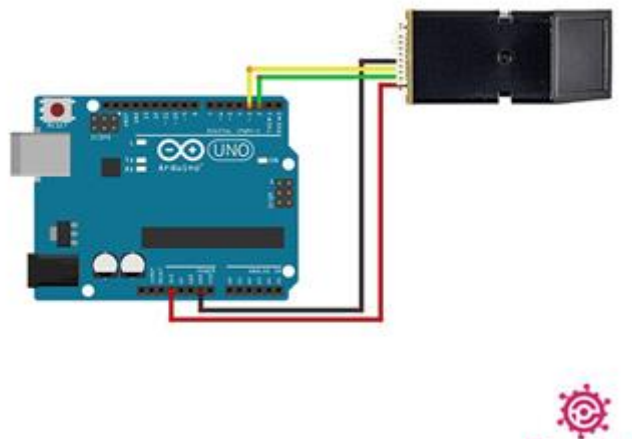


Figure 3.16: DY50 digital fingerprint sensor with Arduino Uno

The installation and configuration of a DY50 digital fingerprint sensor with an Arduino Uno entails connecting the sensor to the Arduino, installing the necessary libraries, and

writing code to interact with the sensor. Here's a general guide:

3.3.7.1 Captor's Connection:

1. Reattach the DY50 capteur's VCC, GND, TX, and RX connectors.

2. Connect the sensor to the Arduino Uno.

Connect the VCC of the capteur to the Arduino's 5V pin.

Connect the GND of the capteur to the Arduino's GND.

Connect the TX sensor to the Arduino's RX pin.

Connect the RX sensor to the Arduino's TX connector. [20]

3.3.7.2 Install the software:

Installation of the Library

1. Open the Arduino IDE.

2. Go to "Sketch" > "Inclure une bibliothèque" > "Gérer les bibliothèques".

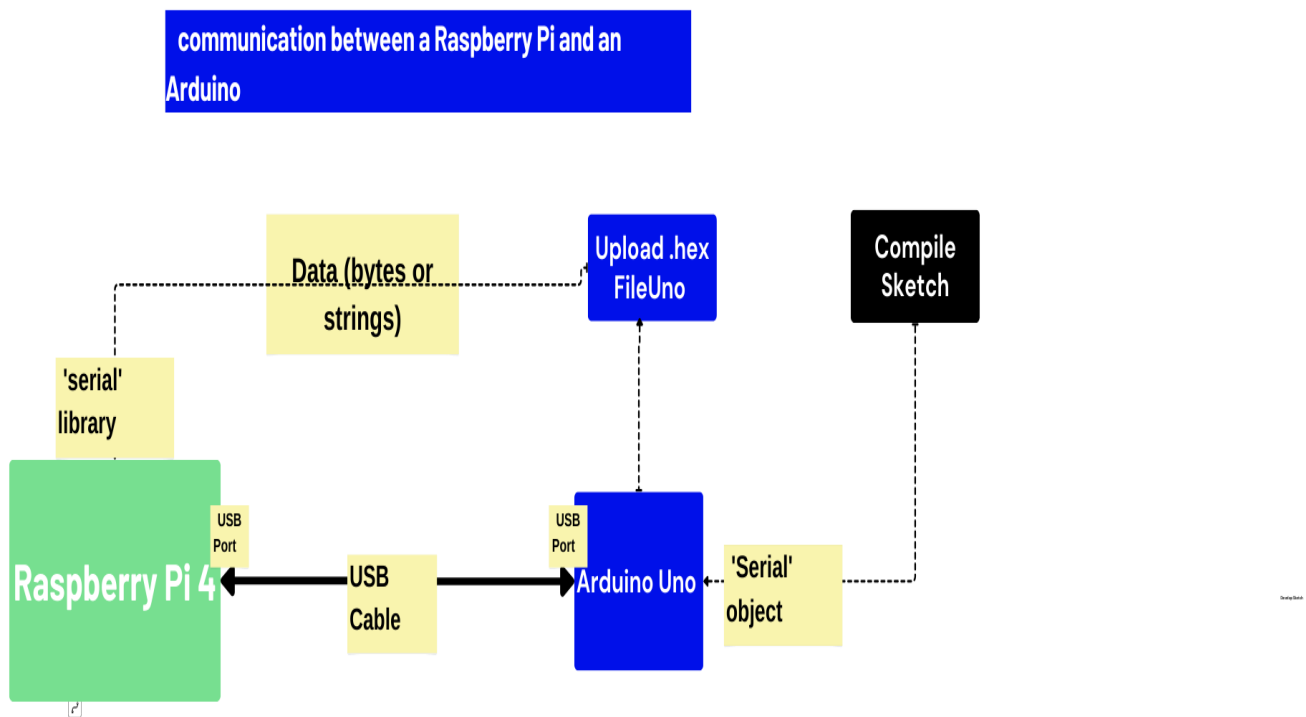
3. Search for "Adafruit Fingerprint Sensor Library" and click "Installer". [21]

4. Restart your Arduino IDE.

Code explanation:

- The Arduino code uses the Adafruit_Fingerprint library to interact with the digital fingerprint sensor.

- The `setup()` function initializes the serial port and digital fingerprint sensor.
- The `loop()` function continuously checks for the presence of a digital footprint. If a digital empreinte is detected, its



ID is read and shown on the serial monitor

3.3.8 Control an Arduino Uno with a Raspberry Pi:

Figure 3.17:how's arduino sends data to the raspberry pi

Follow the steps below to control an Arduino Uno from a Raspberry Pi using avrdude over USB. These steps will help you install the necessary software and correctly setup your environment.[22]

3.3.8.1 Step 1: Connect the Arduino to the Raspberry Pi.

Connect your Arduino Uno to the Raspberry Pi via a USB cable.

3.3.8.2 Step 2: Install avrdude on the Raspberry Pi

1. Open a terminal on the Raspberry Pi.
2. Update the package list and install avrdude with the following commands:

```
sudo apt install avrdude [23]
```

3.3.8.3 Step 3: Identify the Arduino's serial port.

After connecting the Arduino, use the following command to see which port number has been assigned:

```
ls /dev/ttyACM*
```

You should see something like /dev/ttyACM0. Take note of this port because you will need it for the aforementioned commands.

3.3.8.4 Step 4: Use Arduino to control the board.Create a hex file from your Arduino code.

Assume that you already have a hex file named blink.hex. [23]

2. Use avrdude to flash the hex file on the Arduino. Here is an example of an advanced command:

```
avrdude -v -p atmega328p -c arduino -P /dev/ttyACM0 -b 115200  
-D -U flash:w:blink.hex:i
```

Explication of Options:

-v: verbose mode

-p atmega328p: specifies the kind of microcontroller (Atmega328p is used in Arduino Uno).

-c arduino: use Arduino programming protocol -P /dev/ttyACM0: specify serial port where Arduino is connected -b 115200: set baud rate

-D: Disable flash memory erase before writing (optional).

-U flash:w:blink.hex:i: specifies the memory operation (writing the file blink.hex into the flash memory).

3.3.8.5 Step 5: Verify the connection and installation

1. After executing the command, check the terminal messages. If everything went well, you should see notifications indicating that the flashage was successful.

**3.3.8.6 Step 6 :Run the following Python script to communicate with the Arduino Uno:
Python**

```
1 import serial
2 import threading
3 import time
4 import os
5 import pygame
6 import requests
7 import json
8 #avrdude -v -p atmega328p -c arduino -P /dev/ttyACM0 -b 115200 -D -U flash:w:/home/cpi/Desktop/Final/scan/fingerprint.ino.hex:i
9
10 # Open serial connection to Arduino
11 os.system("avrdude -v -p atmega328p -c arduino -P /dev/ttyACM0 -b 115200 -D -U flash:w:/home/cpi/Desktop/Final/scan/fingerprint.ino.hex:i")
12 ser = serial.Serial('/dev/ttyACM0', 9600, timeout=1)
13 pygame.init()
14 def accessgranted():
15     match_sound = pygame.mixer.Sound('accessgranted.mp3')
16     match_sound.play()
17 def accessDenied():
18     match_sound = pygame.mixer.Sound('accessdenied.mp3')
19     match_sound.play()
20 def send_alert():
21     url = 'http://127.0.0.1:5000/sensorestatus' # Replace this with your API endpoint
22
23     # JSON data to be sent
24     data = {
25         "status": "ON",
26         "sensorname": "Fingerprint Sensor"
27     }
28
29     # Convert the data to JSON format
30     json_data = json.dumps(data)
```

Figure 3.18: Example code for reading Arduino Uno with a Raspberry Pi

Explication of Python script

- The script imports the serial library to provide series communication.
- The serial port /dev/ttyACM0 is open for communication with the Arduino Uno.
- A boucle that displays the serial port data indefinitely.
- A command is sent to the Arduino Uno by writing data on the serial port.
- The Arduino Uno's response is lue and décodée.
- The response is processed and shown on the console.
- A time limit is applied before sending the next order.

3.3.9 Installation and setup of a web camera:



Figure 3.19:web camera

The installation of a web camera on a Raspberry Pi 4 may be used to implement a basic access validation system that recognizes faces or objects. Here's an overview of the process:

3.3.9.1 Logistical requirements

- Python library for facial and object recognition (e.g., OpenCV, TensorFlow Lite). [24]

3.3.9.2 Steps: 1. Install the USB camera.

-Connect the USB camera to the Raspberry Pi.

-Install the fswebcam package to capture photos.

```
sudo apt install fswebcam. [24]
```

3.3.9. 3 Configuration of face recognition/objects:

- Install the Python library appropriate for your application. To use OpenCV for facial recognition, run

```
`sudo apt install python3-opencv`. [25]
```

- Configure the face recognition/object recognition module according to the instructions provided by the chosen library.

3.3.9.4 Development of the access validation script

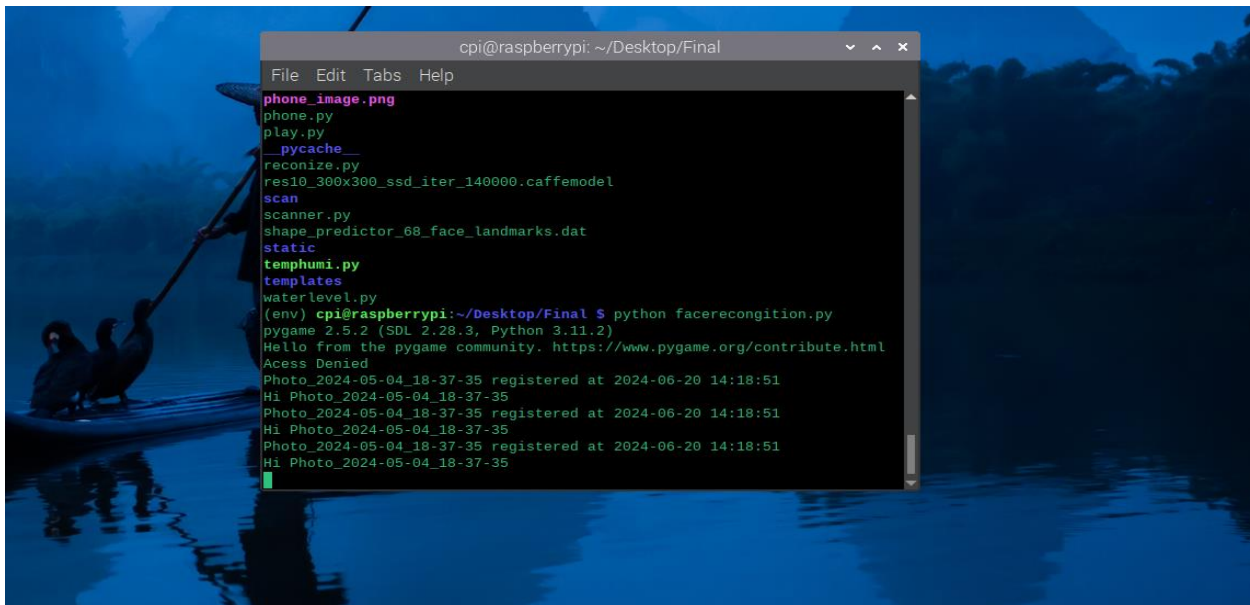
- Create a Python script to take an image using fswebcam and analyze it using the face recognition/object library. The script should compare the captured image to a database of authorized faces/objects and grant or deny access as a result. [27]


```
1  import os
2  import cv2
3  import face_recognition
4  from os import listdir
5  from os.path import isfile, join
6  from datetime import datetime
7  import time
8  import pygame
9  import threading
10
11  pygame.init()
12
13  # Initialize global variables
14  known_face_encodings = []
15  known_face_names = []
16  last_access_granted_time = 0
17  last_access_denied_time = 0
18
19  def access_granted():
20      global last_access_granted_time
21      if time.time() - last_access_granted_time > 2: # 2 seconds delay
22          play_sound(access_granted_sound)
23          last_access_granted_time = time.time()
24
25  def access_denied():
26      global last_access_denied_time
27      if time.time() - last_access_denied_time > 2: # 2 seconds delay
28          play_sound(access_denied_sound)
29          last_access_denied_time = time.time()
30
31  def play_sound(sound):
```

Figure 3.20: Example code for reading web camera

3.3.9.5 Execution of the access validation script

Configure the script to run automatically when the Raspberry Pi boots up, or start it manually when you want to validate access.



```
File Edit Tabs Help
phone_image.png
phone.py
play.py
__pycache__
recognize.py
res10_300x300_ssd_iter_140000.caffemodel
scan
scanner.py
shape_predictor_68_face_landmarks.dat
static
temphumi.py
templates
waterlevel.py
(env) cpi@raspberrypi:~/Desktop/Final $ python facerecognition.py
pygame 2.5.2 (SDL 2.28.3, Python 3.11.2)
Hello from the pygame community. https://www.pygame.org/contribute.html
Access Denied
Photo_2024-05-04_18-37-35 registered at 2024-06-20 14:18:51
Hi Photo_2024-05-04_18-37-35
Photo_2024-05-04_18-37-35 registered at 2024-06-20 14:18:51
Hi Photo_2024-05-04_18-37-35
Photo_2024-05-04_18-37-35 registered at 2024-06-20 14:18:51
Hi Photo_2024-05-04_18-37-35
```

Figure 3.21:python Dans le terminal

3.3.10HoneyPot:

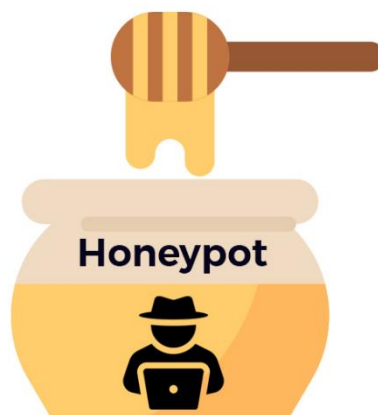


Figure 3 22:HoneyPot

A honeypot is a computer system designed to attract and capture cybercriminals. It allows you to detect intrusions, collect information on the strategies used by pirates, and improve the network's security posture. A Raspberry Pi 4 might be a great choice for installing a honeypot because to its low energy consumption, small size, and low cost.

The honeypots reply in a non-blocking manner and may be used as objects or called directly using the built-in auto-configuration scripts! Furthermore, they are simple to set up and adjust; spinning up a honeypot takes about 1-2 seconds. You can launch several instances of the same kind. To make integration easier, the output may be recorded to a Postgres database, file[s], terminal, or syslog.

This honeypots package is the only one that includes all of the following: dhcp, dns, elastic, ftp, http_proxy, http, https, imap, ipp, irc, ldap, memcache, mssql, mysql, ntp, oracle, pjl, pop3, postgres, rdp, redis, sip, smb, smtp, snmp, socks5, ssh, telnet, vnc. [28]

3.3.10.1 Services Commonly Targeted by Honeypots:

- **dhcp (Dynamic Host Configuration Protocol):** Not directly targeted by honeypots, but attackers might use DHCP to discover active devices on the network.
- **dns (Domain Name System):** DNS honeypots can be used to monitor for attackers attempting to redirect traffic to malicious websites or steal sensitive information.
- **ftp (File Transfer Protocol):** FTP honeypots can lure attackers into trying to exploit vulnerabilities in FTP servers.
- **http_proxy, http, https (Web protocols):** Honeypots can mimic web servers to detect web application attacks and analyze attacker behavior.

- **imap, pop3 (Email protocols):** Honeypots can be used to identify attempts to exploit vulnerabilities in email servers.
- **rdp (Remote Desktop Protocol):** RDP honeypots can deceive attackers trying to gain unauthorized access to remote desktops.
- **ssh (Secure Shell):** As you previously mentioned, SSH honeypots mimic SSH servers to trap attackers attempting to exploit vulnerabilities.
- **telnet (Remote access protocol):** Telnet is an unencrypted protocol and not commonly used anymore. However, Telnet honeypots can still be used to detect legacy attacks that target Telnet servers.[29]

3.3.11.2 Services Less Frequently Targeted by Honeypots:

- **elastic (Search and analytics engine):** While not a honeypot target itself, attackers might attempt to exploit vulnerabilities in Elasticsearch deployments. Security information and event management (SIEM) systems that utilize Elasticsearch can be integrated with honeypot data for better analysis.
- **ipp (Internet Printing Protocol):** Ipp honeypots are uncommon, but could be used in very specific situations.
- **irc (Internet Relay Chat):** IRC honeypots are not very common, but could be used to monitor for malicious activity within IRC channels.
- **ldap (Lightweight Directory Access Protocol):** Ldap honeypots are not as common as other honeypots, but could be used to detect attacks targeting directory services.
- **memcache (Caching system):** Not a direct honeypot target, but attackers might attempt to exploit vulnerabilities in memcache deployments.

- **mssql, mysql, oracle (Database management systems):** Database honeypots exist, but are less common than honeypots mimicking network protocols.
- **ntp (Network Time Protocol):** Not a honeypot target, but attackers might exploit vulnerabilities in NTP servers.
- **pjl (Printer Job Language):** Not a honeypot target.
- **pop3 (mentioned previously):** Email protocol.
- **redis (In-memory data store):** Not a direct honeypot target, but attackers might attempt to exploit vulnerabilities in Redis deployments.
- **sip (Session Initiation Protocol):** SIP honeypots are not very common, but could be used to monitor for attacks on VoIP systems.
- **smb (Server Message Block):** SMB honeypots can be used to detect attacks targeting Windows file sharing protocols.
- **smtp (Simple Mail Transfer Protocol):** Email protocol (mentioned previously).
- **snmp (Simple Network Management Protocol):** Not a honeypot target, but attackers might exploit vulnerabilities in SNMP implementations.
- **socks5 (Proxy protocol):** Not a honeypot target itself, but attackers might use SOCKS5 proxies to anonymize their activity. [30]

3.3.13.3 Not Relevant to Honeypots:

- **vnc (Virtual Network Computing):** VNC allows remote access to graphical desktops, but isn't directly relevant to honeypots.

Understanding the Honeypot Package

As discussed before, it's unlikely a single package offers honeypots for all the services listed. Here are the possibilities again:

1. **Honeygot Framework:** The most probable scenario. The package might be a honeypot framework that allows you to deploy individual honeypot modules for specific network protocols (DNS, SSH, etc.).
2. **Subset of Honeygot Tools:** There's a chance the package offers honeypot tools for some of the network protocol-based services, but not all.

Moving Forward

- **Package Investigation:** If you can share the name or source of the package, I can help you research its functionalities in detail.
- **Targeted Honeygot Deployment:** When deploying honeypots, focus on mimicking services relevant to your security concerns. Analyze what kind of attacks you're most worried about and choose honeypots accordingly (e.g., web application attacks - web server honeypot).

The use of a honeypot on a Raspberry Pi 4 can be a valuable tool for monitoring and protecting the C.P.I. Spa datacenter. A honeypot is an IT device designed to attract and trap cyber attackers, allowing it to detect breaches, collect knowledge about pirate strategies, and improve the network's security posture.

3.3.10.4 Stages of setup

1. Install the honeypot software. Choose a honeypot software that suits your needs, such as Glastopf, Deception Toolkit, or Honeyd. Install the selected honeypot software on the Raspberry Pi 4 according to the developer's instructions.

2. Configure the honeypot to make services and ports accessible to potential attackers. This entails defining the services to emulate, the protocols to be used, and the firewall rules to protect the rest of the network.
3. Set up the honeypot: Connect the Raspberry Pi 4 to the C.P.I. Spa datacenter network while adhering to security and access policies. Turn on the honeypot and start monitoring it to detect attack attempts.
4. Surveillance and analysis: Regularly monitor honeypot logs to identify potential attacks and gather information about attackers. Analyze the data gathered to better understand the strategies and tools used by pirates, as well as to identify potential security flaws in the datacenter network. [28]

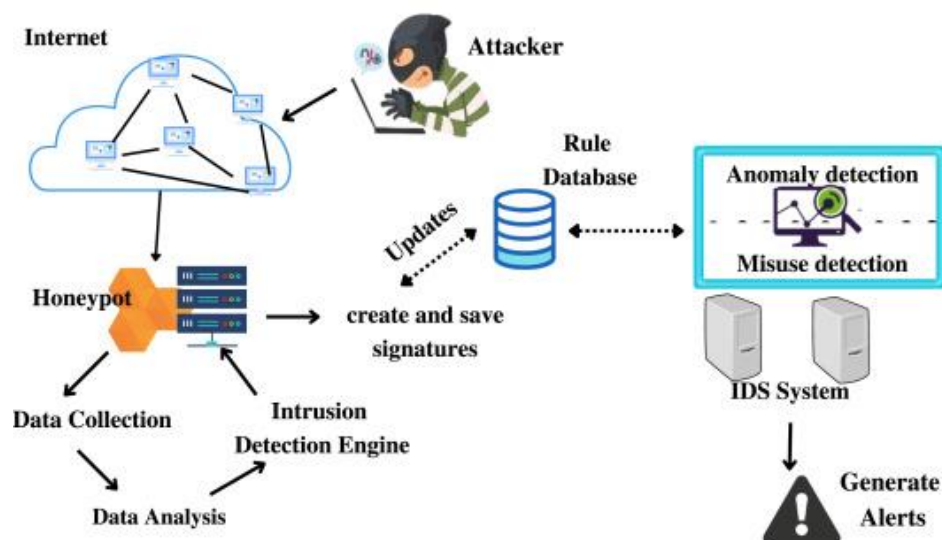


Figure 3.25:how the honeypot actually works

3.4 platform:

3.4.1 Overview of the Platform

Our platform is a comprehensive solution that uses real-time data from various sources to improve the efficiency and security of bank operations. Our platform's primary functions are combining sensor data.

Additionally, our platform has a strong warning system that improves the speed. The platform will automatically send email notifications to authorized staff in the event that any unusual or harmful activity is discovered. This ensures that they are promptly alerted and have time to take proper action. This is a more detailed summary that includes this feature:

3.4.2 Integrating Real-Time Data:

Collection of Sensor Data: Our platform is always gathering data from several sensors that are strategically placed across the infrastructure of the bank. These sensors keep an eye on important variables like network activity, physical security breaches, and environmental conditions.

Access to the Data Center: This sensor data is easily accessible and analyzed by authorized bank employees via the data center interface that is already in place. By ensuring that pertinent data is easily accessible, this integration improves operational decision-making.

3.4.3 Improving Cybersecurity:

Deployment of Honeypots: Our software uses a honeypot method to proactively identify and evaluate potential cyber threats. This technology attracts and studies attack vectors in a controlled environment by simulating weak points in the bank's network.

Data Collection on Threats: Through the honeypot's monitoring, our platform collects useful information about new threats and

attack trends. This data is essential for creating strong cybersecurity plans and anticipatorily reducing threats.

3.4.4 Notification System:

Notifications by Auto: Through sensor data or interactions with honeypots, the platform recognizes any unusual or risky conduct, which sets off an alert system.

Notifications by Email: Authorized personnel receive instant email notifications from the alert system, making sure they are informed right away about any unusual activity or potential dangers. This makes it possible to respond quickly to reduce risks and safeguard the bank's operations.

3.4.5 Technology Employed

Developing the Backend:

Python: Because of its ease of use and strong library support, Python was chosen to enable effective data processing and integration.

Flask: Our online application was built using this lightweight web framework, which allowed for quick development and simple platform deployment.[31]

Databases:

SQLite: Sensor data, user data, and threat intelligence collected from the honeypot were stored and managed using SQLite, a lightweight, self-contained database solution.[37]

Developing the Front End:

CSS & HTML: These core web technologies were used to design an intuitive and eye-catching user experience so that staff members could engage with the platform.[32]

JavaScript: JavaScript was incorporated onto the frontend for responsive features and dynamic content, improving the user experience overall.[34]

ORM, or Object-Relational Mapping:

SQLAlchemy: The SQLite database was interfaced with using this potent ORM tool, which offered a smooth and effective means of managing database operations via Python code.[36]

Integration of Email:

SMTP Protocol: When alerts are triggered, our platform can automatically send emails to specified recipients using the Simple Mail Transfer Protocol (SMTP).[38]

Principal Advantages

Increased Safety: Through ongoing sensor data monitoring and analysis of honeypot activities, the platform greatly enhances the bank's capacity to identify and address security issues.

Quick Reaction: Rapid response and mitigation are made possible by the automatic email alert system, which makes sure that authorized workers are instantly alerted about any suspicious or risky conduct.

Operational Efficiency: Employees may make well-informed decisions more rapidly thanks to real-time access to sensor data, which boosts bank operations' overall effectiveness.

Scalability: Our platform can develop to meet the bank's expanding needs by utilizing Flask and SQLite to handle more sensor data and handle increasingly complicated security scenarios.

In conclusion, our platform is a cutting-edge solution that boosts the security and effectiveness of bank operations by integrating real-time data, detecting sophisticated threats,

sending out automated alerts, and having an intuitive user interface. With the use of state-of-the-art technology and a thorough approach to data management, we can offer the bank and its staff a great deal of value.

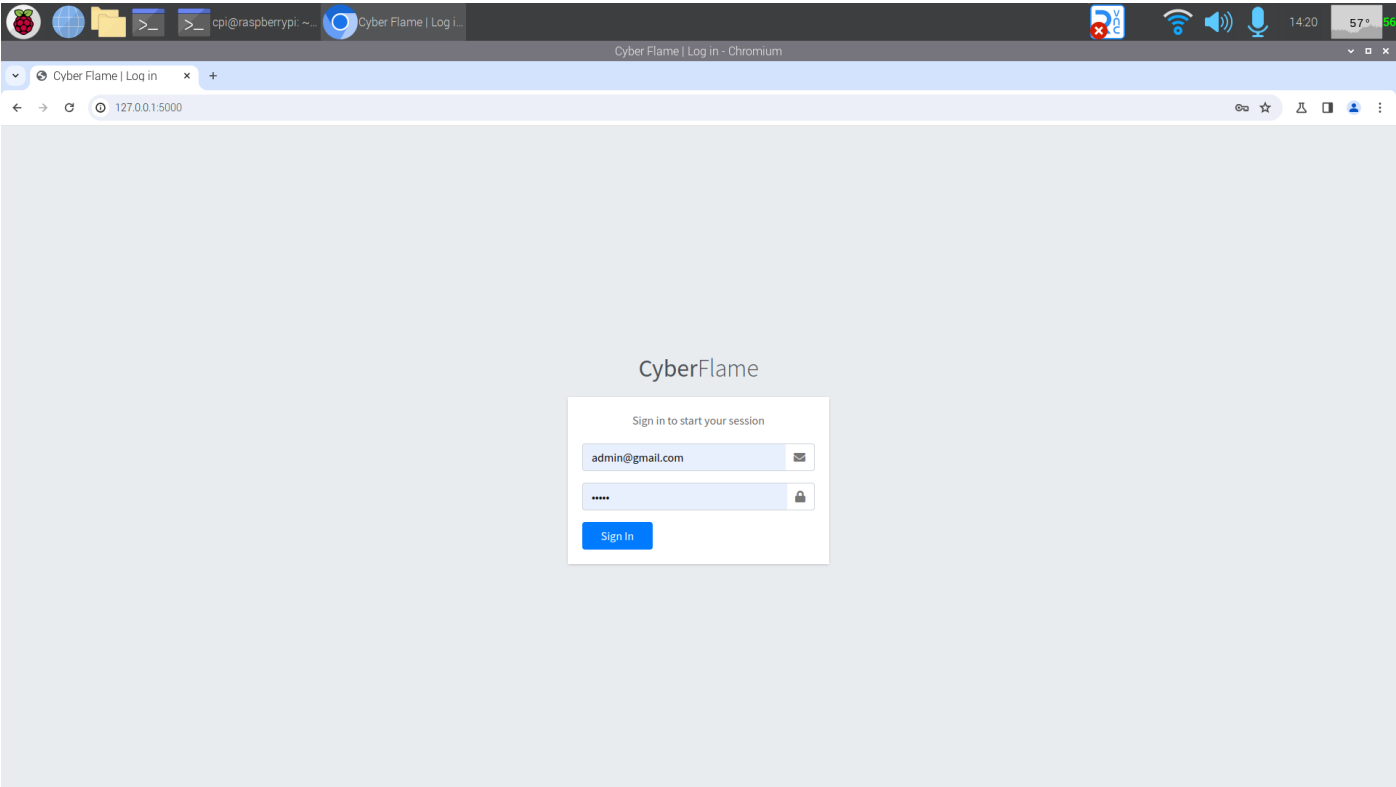


Figure 3.26: the application's login

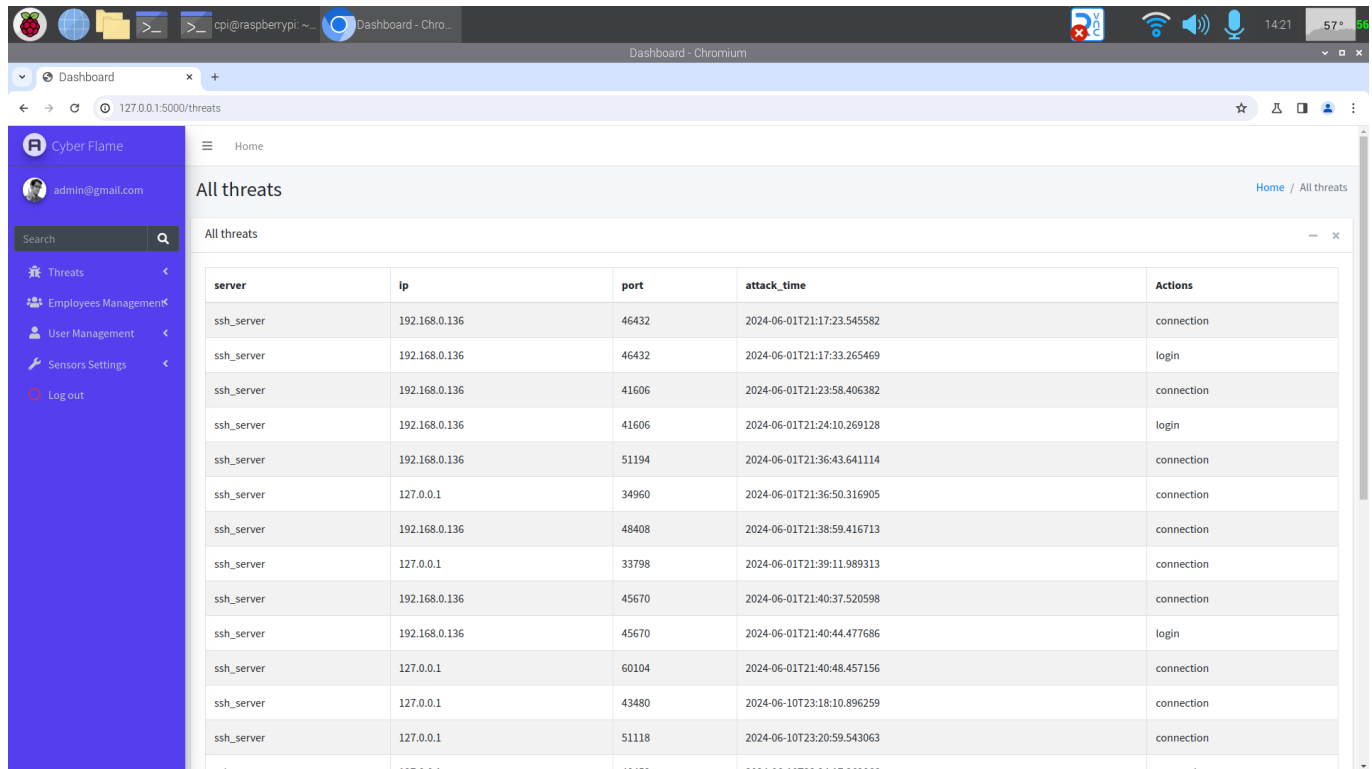


Figure 3.27: Therats

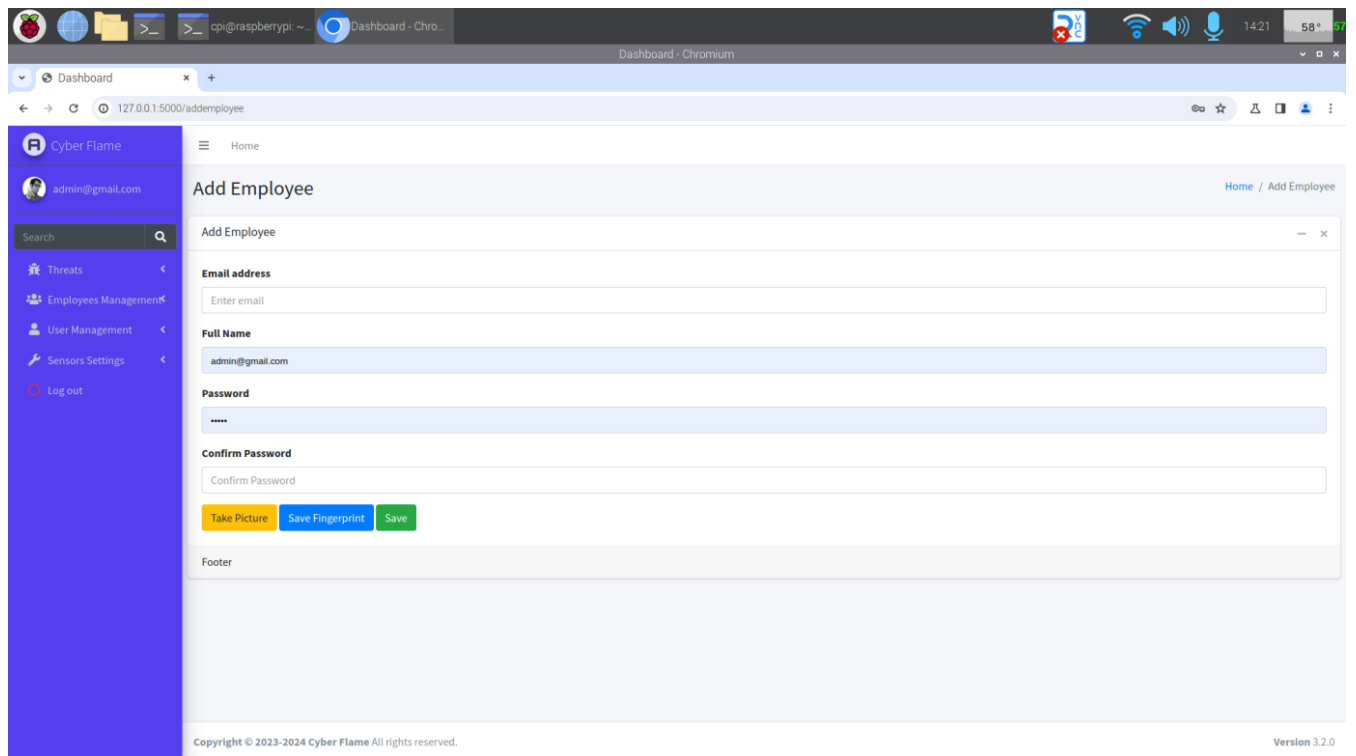


Figure 3.28: Add Employee

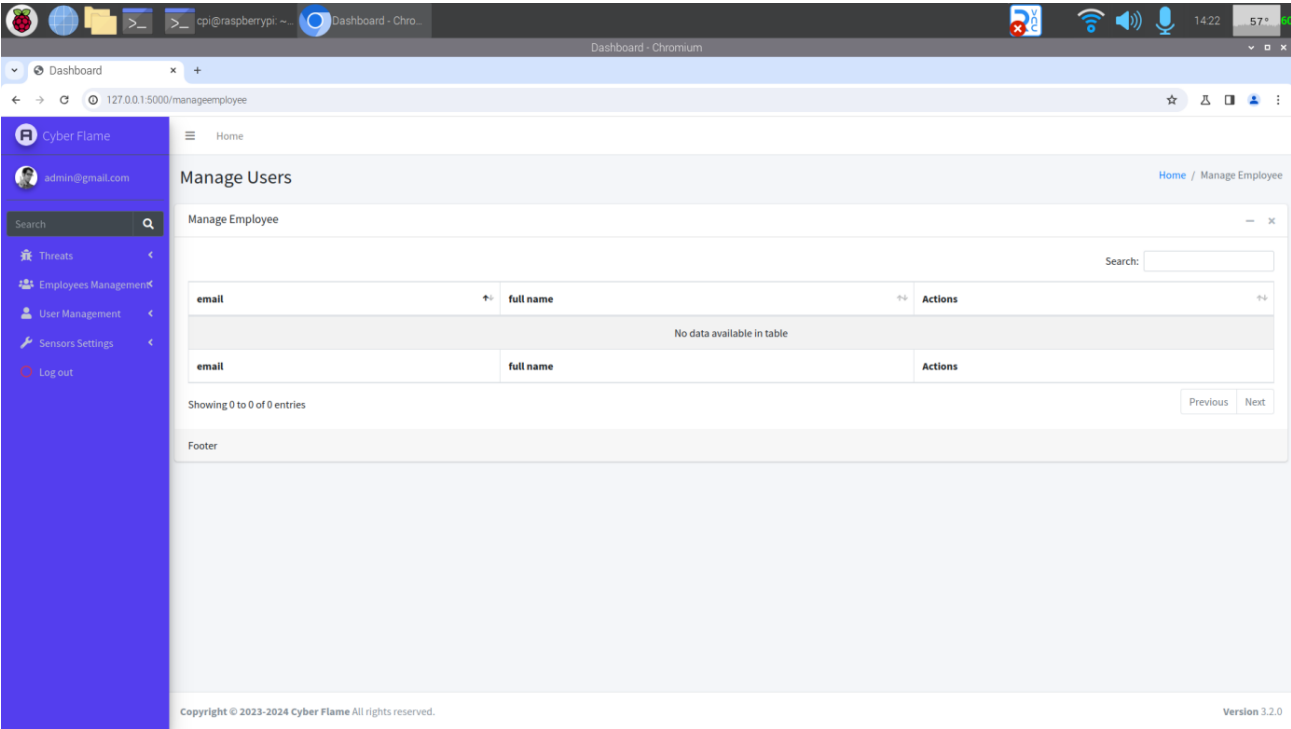


Figure 3.23:MangerUser

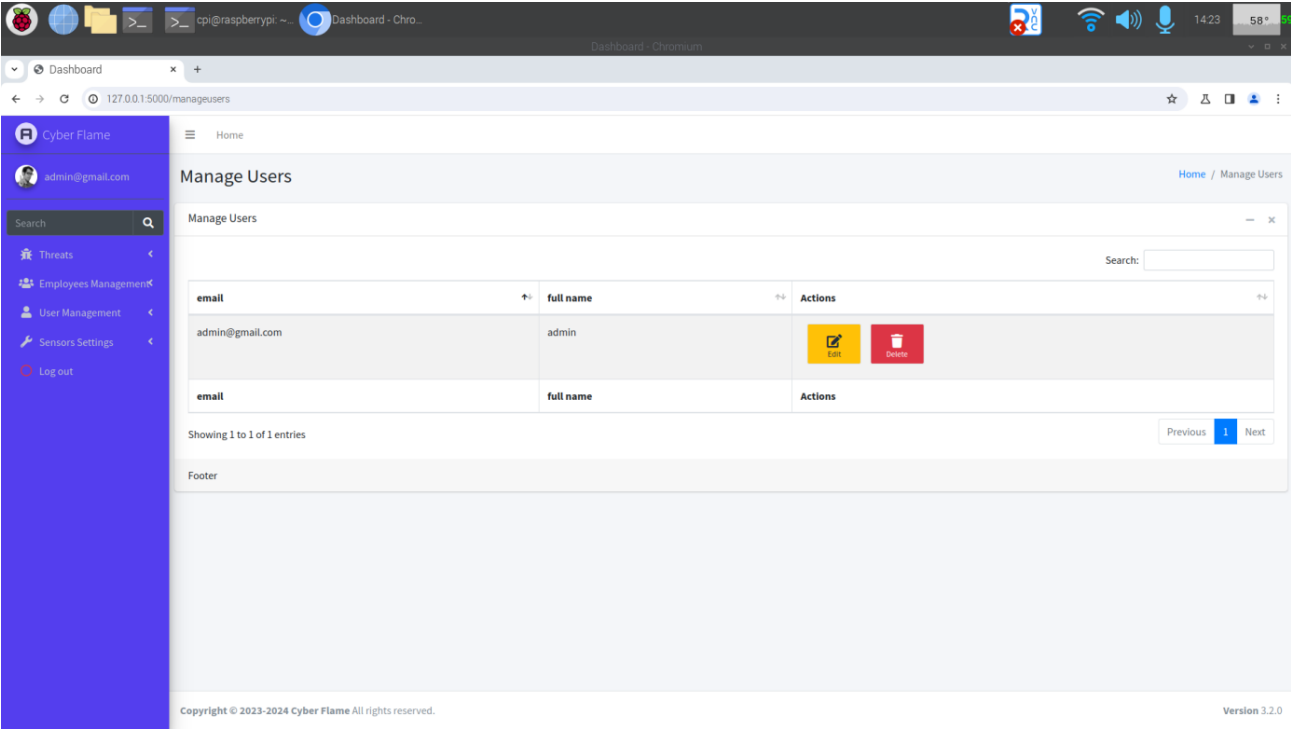


Figure 3.30: Mange Users

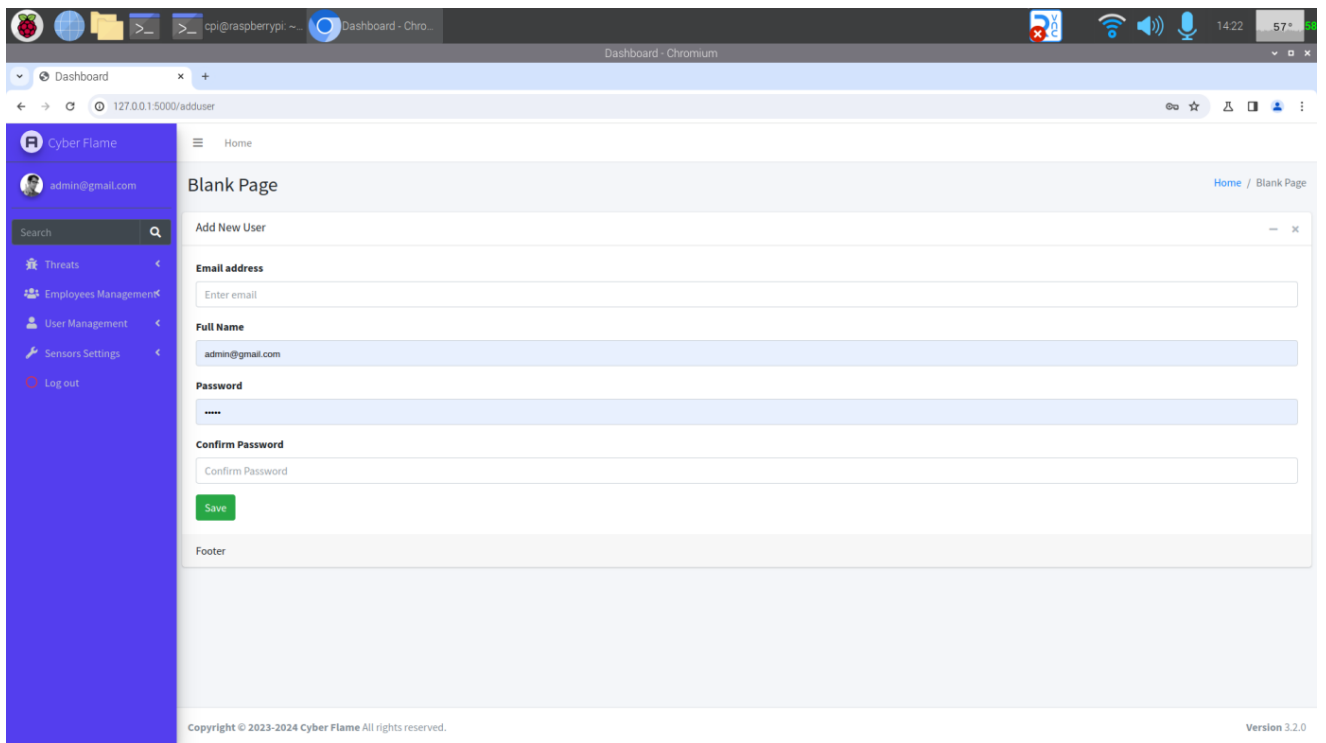


Figure 3.31: Add User

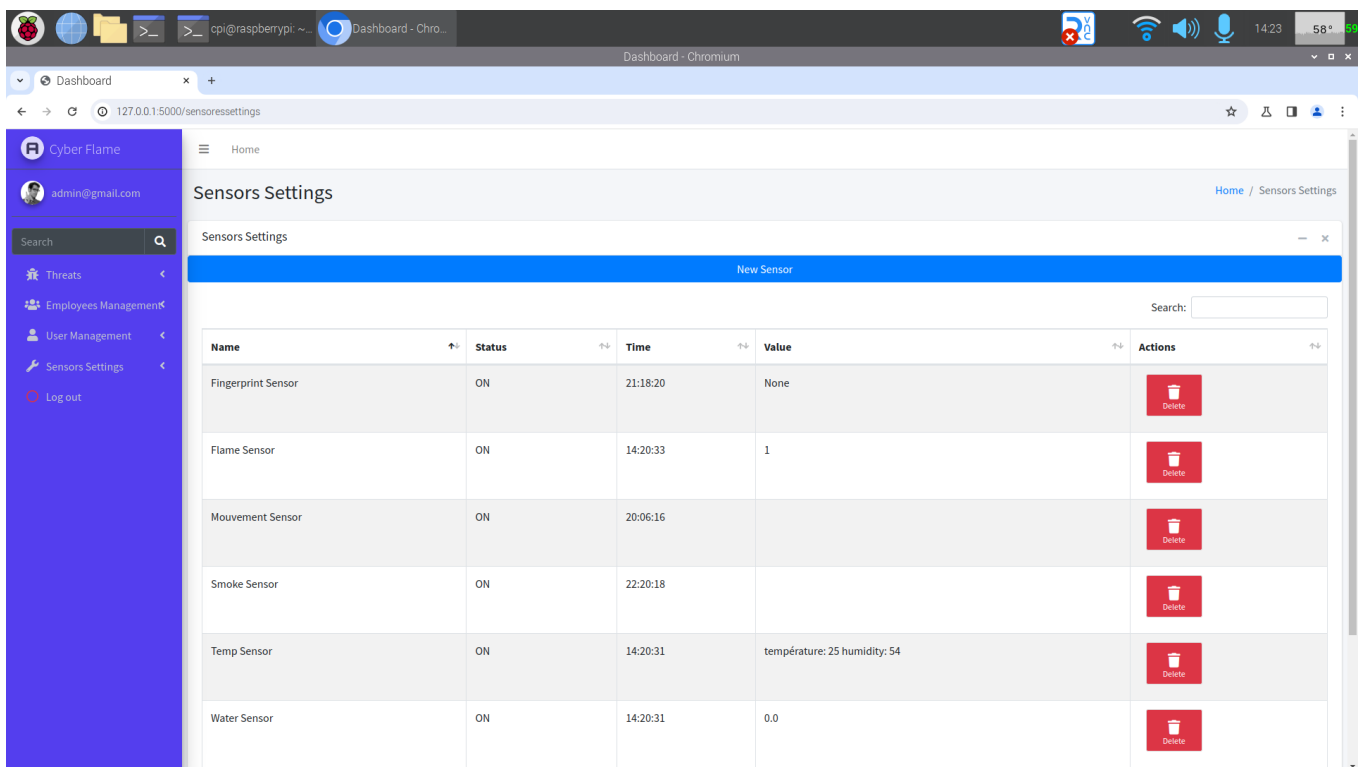


Figure 3.32: Sensors Settings

3.5 Integration of Composants

3.5.1 Process of Integration:

- Communication between composers: Ensure that the sensors, Raspberry Pi, and servers communicate using protocols such as MQTT, HTTP, or WebSocket.
- Configure MQTT for lightweight and efficient message exchange.

3.5.2 deployment in the datacenter

- Set up and configure Raspberry Pi and sensors in real-world environments. Ensure all components are properly configured and working as intended.
- Test communication and data collection in real-time.

3.6 Methods of Testing and Obtained Results

3.6.1 Methods of Test:

- Create unit tests for each system component (capteurs, collection scripts, etc.).
- Integration tests: Ensure all components work together harmoniously.
- Performance tests: Evaluate system performance under real-world settings.

3.6.2 Results obtained:

- Verify that collected data is precise and reliable.
- \tSystem Robustness: Ensure that the system can function without interruption for extended periods.
- Assess system responsiveness during data collection and analysis.

By following these steps, you may ensure an effective installation, configuration, and deployment of your Raspberry Pi and sensors for precise and reliable environmental observation.

3.7- Tests and Validation

3.7.1 Testing Methodology

Verification of Good Functionality:

1. Unitary Tests:

Objective: Ensure that all system components work as intended.

For example, test each capteur individually to ensure accurate data transmission.

Outil: Use Python testing libraries such as unittest or pytest.

3.7.2 Tests of integration:

- Objective: Ensure that all components work together cohesively.

- Ensure proper data transmission to the server, storage, and visualization.
- Create scripts to simulate the real environment and ensure complete integration.

3.7.3 Performance tests

- Objective: Evaluate the performance of the system under consideration.
- Check the system's response time when receiving data from many sensors simultaneously.
- Utilization of load testing tools such as JMeter and locust.io.

```
locust -f load_test.py --host=http://localhost:5000
```

3.7.4 Evaluation of Performance:

1. Accurate Data:

- Method: To ensure accuracy, compare sensor data to known reference values.
- Analyze: Calculate type and average differences between measured and reference values.

2. Rapidity of Alerts:

- Method: Determine the time it takes for a capteur to detect an event and receive the corresponding alarm.
- Analyze: Measure latency and ensure it meets project requirements.

3.7.5 Analyze the Results

Presentation and Interpretation of Test Results

1. Results of Unit Tests

Observation: Unit tests demonstrate that each capteur sends valid data.

Interpretation: Individual components function properly.

2. Results of Integration Tests:

Observation: Captured data is properly transmitted to the server and shown on the dashboard.

Interpretation: Successful component integration.

3. Performance Testing Results:

Observation: The system can handle a large number of sensors without significant delay.

Interprétation: The system can handle the planned fee.

3.7.6 Validation of Functionalities

Confirmation that the solution satisfies the defined requirements:

1. Precision:

Observation: Capteurs measure temperature and humidity with appropriate precision (kind of error < defined threshold).

Validation: Capture accuracy meets project requirements.

2. Rapidity: o Observation: Alerts are sent within X seconds after detection.

Validation: Rapid alerts meet operational needs.

3. Reliability: o Observation: The system operates without interruption during lengthy tests.

o Validation ensures system reliability.

3.7.7 Identifying Limitations and Proposals for Improvement

Limitations of the study:

1. Capacitor Precision:

Limitation: Some sensors may have error margins, affecting data precision.

Proposition: Use high-quality capteurs or calibrate them often to maintain precision.

2. Scalability:

Limitation: Without optimization, the system cannot sustain significant increases in the number of capteurs.

Proposition: Optimize code and use horizontal scaling approaches to manage a larger load.

3. Alert Timeline:

Limitation: Latency may increase with system load.

Proposition: Optimize data processing algorithms and use faster messaging systems such as Kafka.

3.7.8 Propositions for Improvement:

1. Improve data precision by incorporating data correction algorithms to compensate for sensor errors.

2. Performance Optimization: o Utilize caching methods and optimized databases for fast reading and writing.

3. Improving Robustness: Put in place recovery mechanisms for critical components to ensure the system's continuous availability.

By following these steps and implementing the suggested improvements, you can ensure that your solution is precise, quick, reliable, and adaptable to meet current and future requirements.

3.8 Comparison with Traditional Methods

The intelligent management and surveillance of the C.P.I. Spa datacenter using Raspberry Pi 4 provides several benefits over traditional methods, particularly in terms of cost, automation, flexibility, and data accessibility. However, these advantages are counterbalanced by technical challenges, security concerns, and the need for ongoing maintenance.

To maximize the benefits while minimizing the drawbacks, it is critical to properly plan the implementation, train the workers, and implement strong security measures. By taking a balanced approach, the C.P.I. Spa may benefit from technological advancements to improve the management and surveillance of its datacenter while ensuring the reliability and security of its systems

3.9 Conclusion:

our platform offers a state-of-the-art solution for enhancing the security and efficiency of the company's data center and sensitive areas. By leveraging the capabilities of Raspberry Pi and Arduino, the platform enables effective data collection and processing from various sensors, including temperature, humidity, pressure, and motion sensors. This system not only automates branch operations and improves safety but also provides real-time data and alerts, allowing for quick and informed decision-making. The robust frontend and backend

systems ensure seamless user interaction, comprehensive data management, and detailed analysis. Implementing this platform will result in a more autonomous, efficient, and sustainable data center, capable of detecting equipment anomalies and preventing incidents such as failures and fires. Additionally, it will enhance access management during working hours, preventing unauthorized or malicious intrusions. This intelligent management system represents a significant advancement over traditional methods, balancing technological benefits with necessary security measures and ongoing maintenance.

General Conclusion

General conclusion

Our proposed solution described in the document showcases a sophisticated platform designed for enhancing the management and security of data centers using Raspberry Pi and Arduino technologies. The platform's capabilities extend to efficient data collection, processing, and monitoring by integrating various sensors that measure temperature, humidity, pressure, and motion. This system not only automates operations and boosts safety through real-time data and alerts but also significantly improves decision-making processes. The robust design ensures seamless user interaction, comprehensive data management, and detailed analysis, making the data center more autonomous, efficient, and sustainable. Additionally, the platform excels in detecting equipment anomalies, preventing incidents like failures and fires, and enhancing access management to deter unauthorized intrusions. This intelligent management system represents a substantial improvement over traditional methods, balancing technological benefits with essential security measures and ongoing maintenance. Overall, the platform exemplifies a cutting-edge solution that significantly elevates the operational efficiency and security of modern data centers .

References

References

References

The life of Pi: Ten years of Raspberry Pi- university of Cambridge [1]

www.raspberrypi.org[2]

WHAT IS RASPBERRY PI - Opensource.com[3]

Getting started with Arduino - Yeshvanthmuniraj, 2020[4]

www.arduino.cc[5]

www.byjus.com[6]

www.criticalcase.com [7]

<https://www.raspberrypi.com/documentation/> [8]

https://wiki.seeedstudio.com/Grove-Water_Sensor/ [9]

https://gpiozero.readthedocs.io/en/latest/api_input.html [10]

<https://cdn-learn.adafruit.com/downloads/pdf/dht.pdf> [11]

<https://docs.circuitpython.org/projects/dht/en/latest/> [12]

<https://learn.adafruit.com/dht-humidity-sensing-on-raspberry-pi-with-gdocs-logging/python-setup> [13]

<https://roboticsbackend.com/introduction-to-wiringpi-for-raspberry-pi/> [14]

<https://github.com/WiringPi/WiringPi> [15]

<https://einstronic.com/product/infrared-line-tracking-sensor-module/> [16]

<https://pyserial.readthedocs.io/en/latest/> [17]

<https://www.sparkfun.com/products/retired/18929> [18]

<https://www.instructables.com/AIR-QUALITY-MONITOR-AND-ALERT-SYSTEM/> [19]

References

<https://randomnerdtutorials.com/fingerprint-sensor-module-with-arduino/> [20]

<https://github.com/adafruit/Adafruit-Fingerprint-Sensor-Library> [21]

<https://raspberrypi-lab.fr/Tutoriels-Arduino/Communication-serie-entre-Raspberry-Pi-et-Arduino-Uno/> [22]

<https://www.locoduino.org/spip.php?article279> [23]

<https://www.framboise314.fr/i-a-realisez-un-systeme-de-reconnaissance-dobjets-avec-raspberry-pi/> [24]

<https://opencv.org/> [25]

<https://www.geeksforgeeks.org/how-to-install-dlib-library-for-python-in-windows-10/> [26]

<https://www.width.ai/post/tensorflow-facial-recognition> et

<https://github.com/mjDelta/face-recognition-keras> [27]

<https://pypi.org/project/honeypots> [28]

<https://adlumin.com/post/honeypots-101-origin-services-and-types/> [29]

<https://www.techtarget.com/searchsecurity/definition/honey-pot>
[30]

<https://flask.palletsprojects.com/en/3.0.x/> [31]

<https://developer.mozilla.org/fr/docs/Web/HTML> ET

<https://developer.mozilla.org/en-US/docs/Web/CSS> [32]

<https://getbootstrap.com/docs/4.1/getting-started/introduction/> [33]

<https://developer.mozilla.org/fr/docs/Web/JavaScript> [34]

<https://docs.github.com/fr/rest?apiVersion=2022-11-28> [35]

<https://pypi.org/project/SQLAlchemy/> [36]

References

<https://docs.python.org/3/library/sqlite3.html> [37]

<https://www.cloudflare.com/fr-fr/learning/email-security/what-is-smtp/> [38]