Journal of Engineering and Technology for Industrial Applications

ITEGAM-JETIA

Manaus, v.8 n.38, p. 24-36. Nov/Dec, 2022.

DOI: https://doi.org/10.5935/jetia.v8i38.841

ISSN ONLINE: 2447-0228

RESEARCH ARTICLE OPEN ACCESS

ELECTRONIC TIME CLOCK: DEVELOPMENT OF A FACIAL RECOGNITION SYSTEM TO CONTROL THE I.T. TRAINEES AT SEJUSC

Adison Praia[1], Jussara Branches[2], Sildoney Andrade[3], Eliton Smith[4] and Alexandra Amaro de Lima[5]

[1, 2, 3] Bachelor in Computer Science, Universidade Paulista - UNIP. Manaus-Amazonas, Brazil.
[4, 5] Research in Computer Science, Universidade Paulista - UNIP. Manaus-Amazonas, Brazil.
[4, 5] Research in Postgraduate Program in Engineering, Process Management, Systems and Environmental (PPEPMSE) - Institute of Technology and Education Galileo of the Amazon – ITEGAM.

[1] http://orcid.org/0000-0003-1103-0558, [2] http://orcid.org/0000-0002-9630-6826, [3] http://orcid.org/0000-0001-6560-7565, [4] http://orcid.org/0000-0002-8039-7532, [5] http://orcid.org/0000-0003-3918-0013

Email: adisonpraia@gmail.com, jussara.branches@gmail.com, sildoney.ipuchima@gmail.com, elitonsmith@gmail.com, *xanduca@gmail.com

ARTICLE INFO

Article History
Received: November 28th, 2022
Accepted: December 28th, 2022
Published: December 31th, 2022

Keywords:
Facial biometry,
Time Clock;
Identification.

ABSTRACT

The article reports the development of an electronic time clock with facial recognition aimed at Information Technology trainees of the State Department of Justice, Human Rights and Citizenship. This is a device in which it is possible to register the clock-in and clock-out schedules, and in addition, it is possible to check the overtime report where the sector supervisor can offer these hours as days off. Given the deficiencies found in the sector, one being the frequency control that was still being done manually and the other, the high demand for work with the need to work overtime, the theme was developed. The importance of the system is that it makes the dynamics of the working day more automated and optimized. In its creation, the methodology of research, interviews, and data collection was adopted, as well as technological tools such as Biometrics, Programming Language, and Database. After the development and tests, it was possible to implement the electronic time clock in the sector. It was found that with low cost technologies available and with the ease of access to information it was possible to solve the problems of the sector of an organization.

I. INTRODUCTION

The Department of State for Justice, Human Rights and Citizenship (SEJUSC) is composed of many sectors, among them the Information Technology Management (ITM) that is responsible for technological resources. And this sector is outdated regarding the frequency control since it is still done manually. Moreover, on certain days, there is a high demand for services with the need to work overtime.

Most employees of the ITM are trainees and according to current labor laws they cannot receive payment for overtime. Given these facts, we have the following question: would combining facial biometrics with the idea of an electronic timecard that sends the amount of days off make the sector's attendance system optimized and secure?

With the development of technology and ease of access to knowledge, it has become possible to develop electronic systems that meet certain needs. Therefore, using python language with facial biometrics it was possible, through a webcam, to recognize a person's face. This way, the goal of this work was to develop an electronic time clock with facial recognition for trainees of the ITM of SEJUSC.

This work aims to develop a low cost system, to help the sector, where the supervisor, with the overtime bank, can offer the correct amount of days off to the trainee and, thus, optimizing and automating the work in the sector.

Analyzing the sector's demand, as well as the trainees' work dynamics, was the first step taken as a methodology, then creating the layout of the electronic time clock system, to develop the frequency system and the information database, and testing the system by creating a library of the trainees' faces.

This work has the following structure: section II explains the theoretical framework in which facial biometrics, python language, clock-in and database are based; section III explains the

materials and methods for the development of the slack system; section IV shows the results and discussions; section V concludes with the results obtained with the development of the project.

II. THEORETICAL REFERENCE

To develop the proposed solution, it is necessary to study biometrics for facial recognition, python language for programming using its OpenCV library, database to store and organize the information, and the notion of time clocking and attendance control. These topics are discussed in the following sections.

II.1 BIOMETRY

Biometrics have the function of recognizing behavioral patterns or physical characteristics to assess a person's authenticity. In general, these systems are suitable for situations where you want to monitor logical or physical access to an environment, attest to attendance at appointments, establish location monitoring, or ensure the practice of law [1].

Biometrics is a field of study that uses statistical and quantitative analysis to identify human characteristics that are unique to each person. Currently, this area has been related to biometrics, which is the provision of materials through the recognition of measurements and structures of organs capable of distinguishing people [2].

According to [3], the most popular solutions that solve the problem of facial recognition are based on the location and analysis of facial attributes such as nose, eyes, or the overall analysis of the face. The basic flow of facial biometrics systems has four basic steps, they are:

1. Face detection in image or video;
2. Location of the face in the image;
3. Extraction of facial features (nose, mouth, eyes position) and comparison with knowledge base;
4. Return the stored pattern closest to the biometric signature pattern of the input image.

II.2 BIOMETRIC SYSTEM

According to [4], the role of a biometric system is to obtain biometric data from users, extract a specific set of features from that data, and then compare them with pre-registered templates in the database which can be characterized as a pattern recognition system.

According to [5], biometric systems are characterized by the integration of devices, programs, and algorithms in order to compare the characteristics of individuals so that it can be determined whether they belong to the same individual. They also add that in order to obtain results, some steps are important.

- Acquisition: The application digitally captures an individual's biometrics;
- Segmentation: the step of separating the information from biometrics only;
- Quality assessment: assurance that the information obtained is sufficient or correct;
- Enhancement: steps to improve sample quality;
- Feature extraction: salient features are then collected and stored in the model;
- Matching: The collected model is then compared with previously stored models to determine if the feature belongs to the individual being analyzed.

II.3 FACE RECOGNITION

Facial recognition, a skill we use every day in our lives, is the human cognitive trait that developers take inspiration from in the search for something that can process digital images automatically. Facial Recognition technology involves various automated processes in which digital facial representations are used to attempt to identify or verify the identity of individuals [6].

The face recognition process can encompass numerous parts [6]. Capture and detection are related to the acquisition of images or photographs. Captured situations can be: authentication to reach the tool or service; in a controlled environment; voluntary or not; and image collections available on social networks. Registration, on the other hand, is the method of collecting visual information from individuals to form a set or database [6].

The data collected in the previous steps is then processed using a face recognition algorithm. Traditionally, recognition algorithms can be divided into geometric methods relating to photometric or salient features, allowing classification between holistic algorithms - those that seek to fully recognize faces - and feature-based methods - methods that analyze local facial features and store parameters and metrics, e.g., angles and distances between facial features are used as descriptors for comparison in future face identification processes [7].

II.4 PYTHON LANGUAGE

Introduced in 1991 by Guido van Rossum, it is a free language (even suitable for commercial projects), and today can be programmed for desktop, web, and mobile devices [8].

The [9] also talks about its main features, which include: ease of use, through elegant syntax and easier prototype programming;

A wide variety of libraries that support most programming-related tasks; do not use special character types at the end of each line of related code, such as ";" (semicolon), which is common in other languages; also support many types of programming paradigms, such as object and aspect oriented, functional, imperative, and reflexive [10]. After all, it is an open-source language and not exclusive to a single operating system.

II.5 OPEN-SOURCE COMPUTER VISION LIBRARY (OPENCV)

Among all alternative methods of biometric identification, face biometric identification is the most prominent [11]. This is because face recognition requires less contact between the individual and the device receiving the biometric information, compared to other biometric features (iris, fingerprints, etc.), and has a low cost of installation and supervision, which basically requires a camera and a server running an application programming interface (API) for face recognition.

Among the APIs ready on the market, the Open-Source Computer Vision Library (OpenCV) stands out, a free and open-source library containing a large number of means for designing computer vision practices [12]. Developed by Intel, the OpenCV library has over 500 features covering multiple spheres of computer vision [13]. Containing interfaces in C++, Python, and Java languages, OpenCV supports operating systems such as Mac OS, Windows, Android, Linux, and iOS.

According to [14], one of the purposes of OpenCV is to provide easy-to-use computer vision support in order to help people quickly produce complex computer vision applications. The OpenCV library is used in different parts of the world and has a

pool of over 47,000 users with a download count of over 14 million [12]. From the moment it was released in 1999, the OpenCV library has been employed in many applications, products, and studies. These features cover automatic surveillance of security systems, distortion reduction of medical images, camera calibration, etc.

II.6 DATABASE

Databases can be defined as collections of integrated data. Data are facts that can be registered and that have an implicit meaning, for example: names, phone numbers and as addresses that are stored in phone books or on computers through spreadsheets. This information is a collection of data and consequently, a database [15].

A database relationship is an association between tables created using a join statement to retrieve data. In the relational model, there are three types of associations between different entities (tables). These relationships are used to map how tables interact and can be categorized in three ways: 1-1 (one to one), 1-N (one to many), and N-N (many to many) [16].

II.7 USING FACIAL BIOMETRICS TO CONTROL TIME CLOCKING RECORDS

By controlling the working hours of company employees, usually done manually through a time book, authorized by the Ministry of Labor, through Ordinance No. 373/2011, using computers to mark time, organizations are able to condition this procedure and suppress points of failure. According to [17], "This is marked by the use of technologies such as fingerprints (biometrics), as computer systems are used in many companies".

Time and attendance or time control is the act of recording employees' clock-in and clock-out schedules at their place of work. Time control occurred in several ways over the years, from the constant viewing and recording by an employee of other people's action times, to notes made by the worker himself, in a book, where he registered his entry and exit, clocks, watches, cards, which make room for technological innovations, allowing changes such as biometric clocks and time stamps made by cell phones or computers [17].

The use of biometrics is the prerogative of large companies, but this reality is changing very dynamically. Small businesses and homes are already using this technology in order to increase security and prevent theft, fraud and other forms of attack. The adoption of these systems helps to increase the authenticity of individuals and increase the security of computer networks [18].

III. MATERIALS AND METHODS

Applied research is concerned with the application, use and practical consequences of knowledge. It aims to apply scientific knowledge to solve a wide variety of individual or collective problems. It is accomplished by means of "applied sciences" and "technological sciences" [19].

Regarding the research methods used in this work, in search of a solution to the problem presented as a referential framework, it can be said that it was an applied research work in order to generate sufficient knowledge about the subject, with the interest of contributing with a practical application, capable of supplying the central question.

The materials used in making the facial recognition device project are Notebook, Webcam, Astah Community, PyCharm IDE (version 2020.2), MySQL DBMS.

The methodology used for this project in the first place was to collect the necessary field data, in this case, to capture the images, and also to verify how the frequency structure was until the moment of this research. In addition, the survey of theoretical material in websites, books, academic articles, etc., and consequently the choice of the necessary tools, that is, the use of programs and specific language for programming the software.

IV. RESULTS AND DISCUSSIONS

From this chapter, the results obtained in the development of the electronic time clocking system will be discussed, which shows the amount of overtime for the administrator, the manager of the ITM, the amount of overtime that the trainee has to be converted into days off, so called slack time.

IV.1 ANALYSIS OF THE DYNAMICS OF THE INTERNS' WORKING HOURS IN THE ITM SECTOR

In the requirements acquisition phase meetings and interviews were held with the help of the manager, where the working hours at the ITM were analyzed. The department consists of registered employees on business hours from Monday to Friday, i.e. 8 hours a day, with a lunch break between 12:00 and 13:00. However, for the trainees, the schedule is 6 hours a day, therefore allocated in two shifts: 8am - 2pm and 11am - 5pm, thus configuring two shifts, morning and afternoon, respectively.

The manager mentioned during the meetings that ITM serves several departments in different areas of the city of Manaus and also some cities in the interior of the state of Amazonas. On some days, the demand for work is quite high and overtime is needed. However, there is a problem, because according to the current labor laws, trainees cannot receive payment for overtime.

As a solution to the problem, he suggested to the trainees to work overtime under the following conditions: 6 hours of work plus 1 extra hour maximum. And as a bonus, the individual who agreed with the idea would be given 1 day off, that is, if you worked 1 hour overtime you would get a day off. However, there is a deficit in this solution, because the manager would be losing 5 hours of work for every 1 extra hour worked.

In addition to this problem, the attendance register is done manually. For example, a trainee in the morning, the registration form is filled out as follows: check in at 8am and sign; check out at 2pm and sign. Thus, configuring it as a British record. However, this does not happen in reality, because no one keeps this attendance with such accuracy and fidelity.

Below, in Figure 1, we have the monthly attendance register template to be filled in, for the month of August in the year 2022.

Figure 1: Attendance record sheet.
Source: [20].

In a second technical visit, it was necessary to collect the name, work registration, e-mail, date of birth, gender and shift of the trainees to put into practice the work proposal, however, as a form of security, the personal data were not disclosed in this monograph. The tests performed with the device created and presented in this document were tested with the authors' data, and also, each one received a pseudonym: user 1 (Sildoney), user 2 (Adison) and user 3 (Jussara).

IV.1.1 Specifying Software Requirements

Finalizing this first stage, the Software Requirements Specification document was created. This document contains product information such as the needs, mission, limits, benefits, definitions, and acronyms. These can be seen in Chart 2 below.

Table 1: Software Requirements Specification.

1 Objectives of this document		
The purpose of this document is to describe the specifications of the electronic time clocking registration needs for trainees of the ITM sector of SEJUSC, which will use the concept of Facial Recognition to perform the registration, as well as to present a report of bank of hours for use in time off. Target audience: customer, users, and developers.		
2 Product scope		
2.1 Product Mission		
The mission of the Electronic Time Clocking System with Facial recognition is to contribute to the improvement of the registration control of the exceeding hours worked in the trainees' day, in order to inform a monthly report of hours to be used for assisted leaves.		
2.2 Product limits		
1	The system will not be effective in low-light environments	
2	For efficiency, the System will require the user to position the face so that it is free of sunglasses, masks, and caps or hats.	
3	For each Trainee, several photos in different positions will be required to be registered in the facial recognition system (registration of 100 to 400 frames from various positions of the face).	
4	The photo registration environment should have a standard lighting for all photos.	
5	The System is totally dependent on the electrical energy of the applied environment.	
6	The good functioning of the system depends on the seriousness and responsibility of the people in charge of the sector.	
7	For each registration made by the user through facial recognition, a proof of entry or exit will be sent via *email.*	
2.3 Product Benefits		
Order Number	**Benefit**	**Customer Value**
1	User identification through Facial Recognition for entry and exit registration.	Essential
2	Sending a proof of registration via *email*, stating the exact date and time.	Essential
3	Allow the supervisor to have full access to the trainee's information and to the attendance register mirror.	Essential
4	Allow to inform monthly or in a certain period the amount of hours available to be used as time off.	Essential
5	Automatically unenroll the trainee if he is dismissed.	Desirable
2.4 Definitions and acronyms		
Order Number	**Term**	**Definition**
1	Face Registration	Registration of the trainee's face frames.
2	User Registration	Registration of user information.
3	Face Verifier	Identify the trainee by comparing the *Webcam* capture to the face library.
4	Proof via *e-mail*	Document via *email* with the data regarding the time clocking registration.
5	Query System	System for visualization of time clocking registers.

Source: Authors, (2022).

IV.2 DEVELOPING THE SYSTEM LAYOUT

In the development of the system layout was necessary the use cases that according to [21] represent "[...] a set of functions that the system must perform to meet customer requirements". Therefore, the use case diagrams should demonstrate the essential functions of the system. In this diagram model there are the actors, who interact directly with the system functions. The time clocking control use case modeling is presented in Figure 2.

In the diagram, the "trainee" actor has access to "Time clocking Registration" and thus receives a "proof via email", and can also "Consult the registration report", in other words, he can make the registration, check the registration report and the amount of bank hours. An observation to be made is that the "proof via email" is only sent when the "clock in/out " action is exclusively performed. The "Supervisor" actor has access to the "Registration/Report Control", "Trainees face library" and "Registration Control", and all this when "Query Registration Report", that is, he/she can insert, change, delete and update the user's (trainee's) registration. Thus, we classify the user types into administrator (supervisor), trainees and super-administrator (developers).

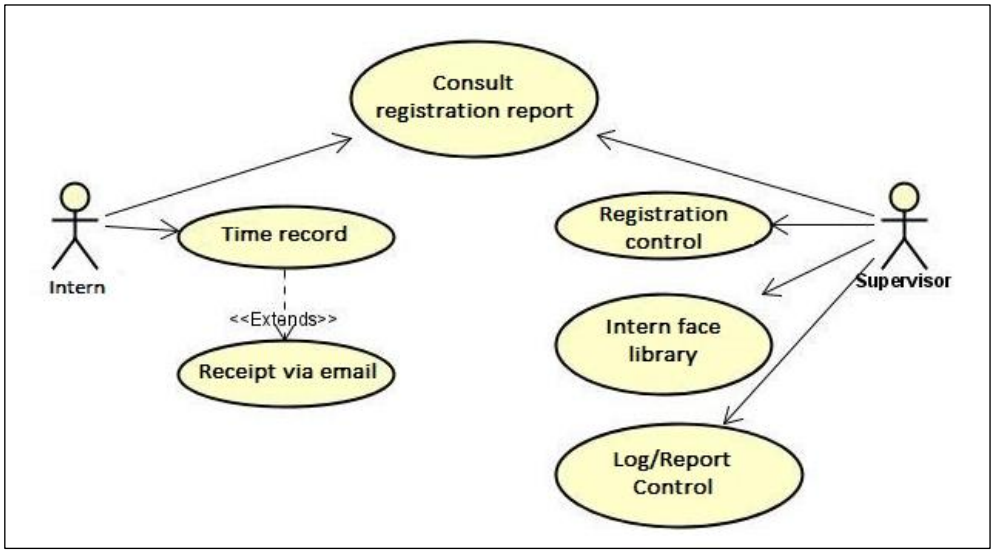


Figure 2: Diagram representing the use of the Time Clock.
Source: Authors, (2022).

In [21] the author explains that the activity diagram is a resource that "allows modeling the system behavior, denoting the logical paths that a process can follow ". In the system, the main activity diagram is the trainee's time clocking record, where the trainee performs facial recognition and has two conditions: if the face is not registered, it is requested to do so; if the face is registered, then the identification is done, consequently the record of entry/exit of the working day that, finally, sends a voucher via email. This process can be seen in Figure 3 below.

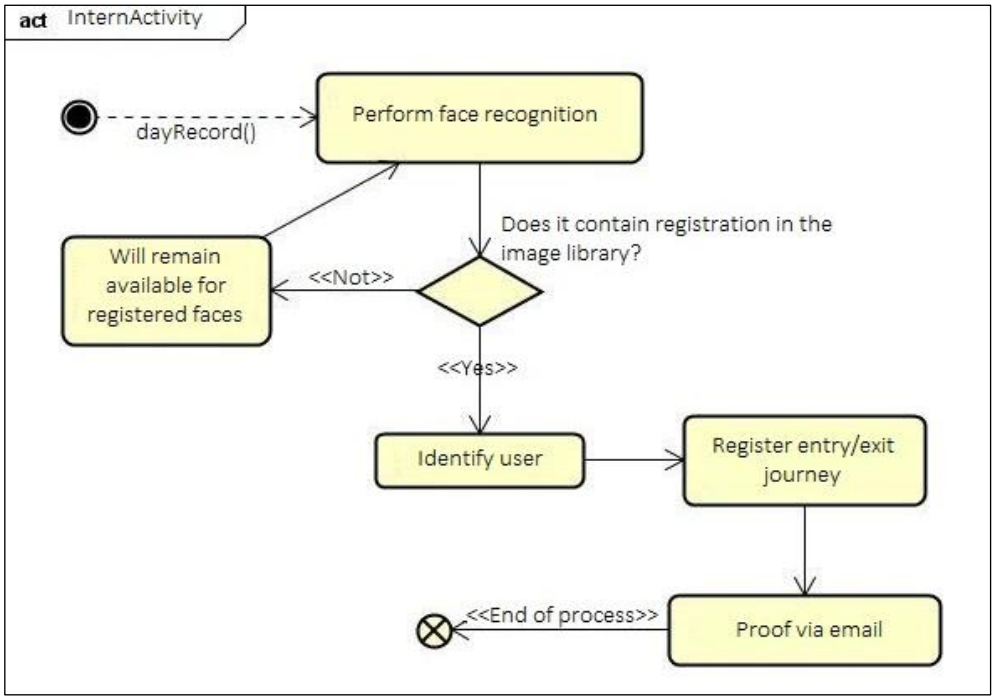


Figure 3: Intern's clocking time register diagram.
Source: Authors, (2022).

Prototyping is a technique used by developers to make an abstract idea, only registered in documents and diagrams, very close to the final product. In addition, prototypes "reduce project uncertainties, as they are an agile way to abandon alternatives that are not well received and, therefore, assist in the identification of a more assertive final solution" [22].

In this prototype we have a database developed in a web interface that stores the information for queries and registration procedures. Figure 4 shows the system's data storage process and is divided into the following entities: "trainee", "timesheet" and "supervisor", where each entity has its attributes which are information to be stored, that is, tables within the database. Thus, the attributes for the user will be fields to be filled in with the requested data.

The trainee's attributes are: name, e-mail, enrollment, shift (morning or afternoon), gender, date of birth and id. An observation about the id: it is an internal identification given by the database, this occurs uniquely for each attribute and is seen only by the developer. To continue, the attributes of the timesheet are: day, month, year, time of registration, overtime, extra minute, and id. The supervisor's attributes are: name, role, email and id.

The database has classifications of relationships, and these have been previously defined. So, in this system there are two relationships: the query relationship which is N-N (many to many) when it comes to the "supervisor" entity, that is, he can query any profile of the "clocking sheet" entity; and the record relationship which is 1-N (one to many), where the "trainee" can record several times in the "clocking sheet".

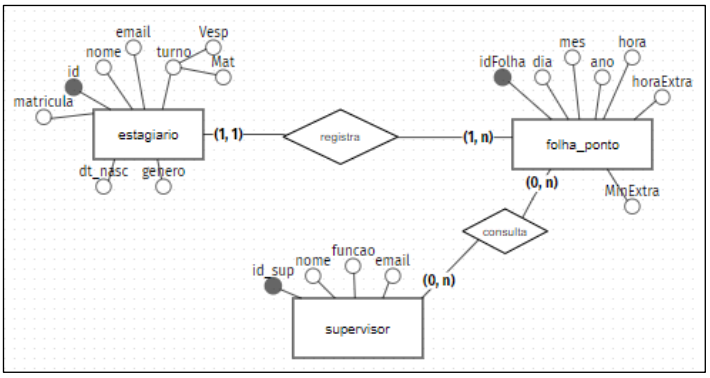


Figure 4: Process of storing the system data.
Source: Authors, (2022).

IV.3 DEVELOPMENT OF THE TIME CLOCK

The language chosen for the development of the system was Python, because using it in the PyCharm IDE there is a variety of packages that are fundamental in facial recognition and that assist in the system performance. This way the following packages were used: OpenCV, numpy, OS, Datetime, email.message, smtplib, tkinter and Mysql. And, for the dynamics of the process, two subsystems were developed, one that performs the function of registering faces in the image bank and the other to identify the trainee through facial recognition and record the time clocking.

IV.3.1 Registration of Trainees in the Database

Initially it was developed as the subsystem that has the objective of performing the registration of trainees through a form created with the help of the Tkinter library, obtaining all the relevant personal and professional information cataloged in the requirements survey highlighted in the topic 4.1. This data was used as parameters for the other processes to identify the faces in the image bank and the time clocking record. The trainee registration can also be performed by the web system, which is treated in topic 4.3.4.

Tkinter is a Python language library that comes with the standard installation and allows you to develop graphical user interfaces. This means that any computer that has the Python interpreter installed is capable of creating graphical interfaces using Tkinter, with the exception of some Linux distributions, requiring the module to be downloaded separately [23].

The form, shown in Figure 5, has the following fields: ID, Name, Date of Birth, Gender, E-mail, and Shift of the trainee.

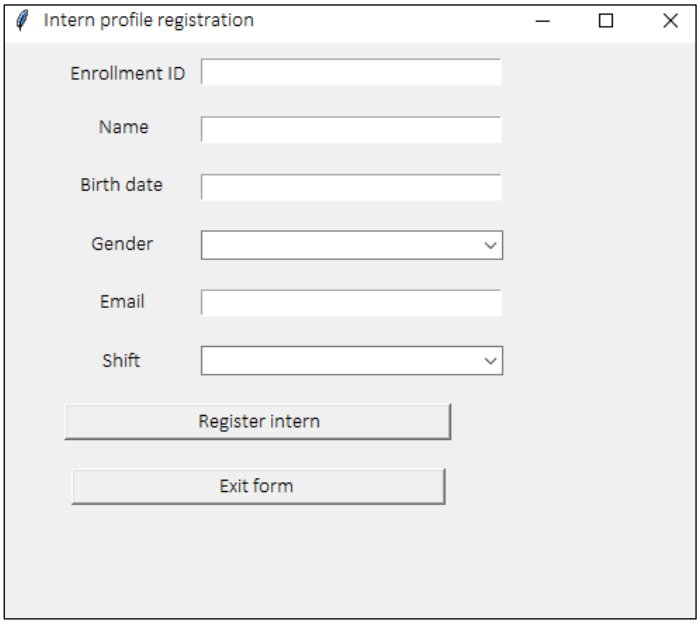


Figure 5: Intern registration form.
Source: Authors, (2022).

After filling out the form, the system needs to connect to the database to store the information, and for this the mysql API was used, which, according to [24], allows an application to connect and interact with a database system.

Figure 6 demonstrates connecting to the database and directly accessing the "bdhoras" database by passing the "host", "user" and "password" parameters to connect, which has all the trainee, user, supervisor and time clocking record information tables.

```
def conectandoBD():
    cnx = mysql.connector.connect(
        host='localhost',
        user='DBUSER',
        password='DBPASSWORD',
        database='bdhoras'
conexao = conectandoBD()
cursor = conexao.cursor()
```

Figure 6: Database connection function and instance.
Source: Authors, (2022).

Next, the information collected by the form in figure 5 is stored in the database through the function "registroEstagio", shown below in Figure 7.

```
def registroEstagio(idMatricula, Nome, genero, turno, dtnasc, email):
    dataBD = datetime.strptime(dtnasc, '%d/%m/%Y').date()
    cursor.execute(f"use {DATABASE}")
    comando = f"INSERT INTO estagiario " \
              f"(idMatricula, nome, genero, turno, dt_nasc, email) VALUES " \
              f"({idMatricula},'{Nome}','{genero}','{turno}','{dataBD}','{email}')"
    cursor.execute(comando)
    conexao.commit()
    print("Dados inseridos com sucesso")
```

Figure 7: Function for registering an Intern in the database.
Source: Authors, (2022).

The database management system is powered by MySQL Workbench. This holds all the data that the system can provide, such as information about registered trainees and supervisors, as well as time clocking registration data and user registration with a login and password. See Figure 8 below.

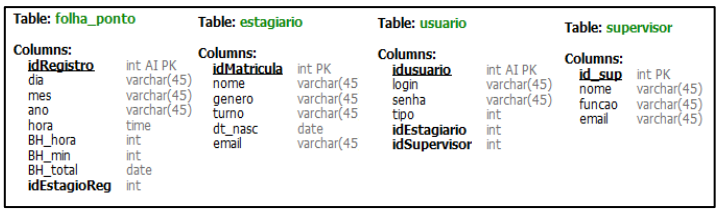


Figure 8: Registered users' information bank.
Source: Authors, (2022).

IV.3.2 Registering Faces to the Image Bank

In this process the functionality of the subsystem that performs the registration of frames in various positions of the trainees' faces that were captured by the computer's webcam for storage in the pre-defined directory created by the system was addressed.

Initially, to create the image bank storage directory, it was necessary to use the OS package which, according to Lima [25], provides functions to interact with the operating system. OS, comes in Python's standard utility modules. This module provides a portable way to use operating system dependent functionality.

Among the various functionalities, the OS was used mainly to create the function "creatDir" shown in Figure 9, whose execution is to manipulate the system to generate a folder, if nonexistent, in the project's root directory, and inside it, a new folder is created that receives the name of the variable "matrícula", which corresponds to the identifier of each trainee registered by the system for storing the person's face frames.

```
def creatDir(matricula, path=''):
    # verifica se tem um diretorio, se não tem ele cria
    if not os.path.exists(f'{path}/{matricula}'):
        os.makedirs(f'{path}/{matricula}')
```

Figure 9: Image bank directory creation function.
Source: Authors, (2022).

In the sequential structure of the source code, there is the creation of the activity of the function "cadastroDir", highlighted in Figure 10. This is responsible for naming the folder in the system matrix as "USER", where, inside it, a new folder is created, identified by the trainee's enrollment variable, to store the user's face frames.

```
def cadastroDir():
    global cadDir
    cadDir = True
    creatDir(matricula, 'USUARIO')
```

Figure 10: Function invoking "creatDir" for folder creation.
Source: Authors, (2022).

Next, the activity that registers the trainee's frames inside the "USER" folder was developed. And, for this, the library used was OpenCV. This was essential in the function of storing the faces in the system's image bank. The code in Figure 11 highlights the function called "registerImage" which is responsible for mapping the face, being defined as variable "img". And consequently, a folder identified by the trainee's registration number is generated with 100 frames.

```
def cadastroImagem(img):
    # salvando as fotos na pasta do usuário e nomeando elas
    global matricula
    qtd = os.listdir(f'USUARIO/{matricula}')
    cv2.imwrite(f'USUARIO/{matricula}/{str(len(qtd))}.jpg', img)
```

Figure 11: Function for registering faces in the image bank.
Source: Authors, (2022).

It should be noted that this amount of frames does not reach the capacity of 3 megabytes, as can be seen in Figure 12. Therefore, a large number of registered users can be added. And yet, this amount of 100 frames is considered the minimum effective for facial recognition. However, this value can be changed to more, thus being efficient in the identification.

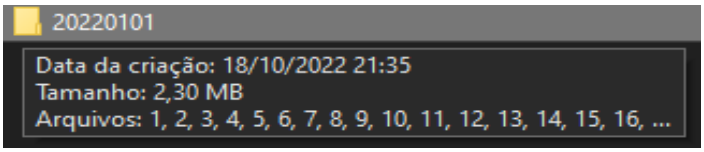


Figure 12: Size of a user's frame folder.
Source: Authors, (2022).

Next, after running the system, the code enters a loop and performs the decision command, in Figure 13. Here the callback execution of the "cadastroDir" function, previously seen in Figure 10, takes place.

After creating the directory, according to the source code in Figure 13, the system accesses the webcam and opens a window where you can keep track of all the images captured up to 100 times in jpg format and name them from 0 to 100. The system then uses the function "cadastroImagem", which uses the parameter "resize", responsible for cropping the trainee's face for storage in the image bank.

```
if cadDir:
    cv2.putText(frame, str(cadRostoCont))
    cadRostoCont += 1 # Incrementando a id
    cadastroImagem(resize) # Salvando os rostos
```

Figure 13: Decision Command for Registering Faces.
Source: Authors, (2022).

IV.3.3 Trainee Authenticator with Facial Recognition

In this topic, the functionality of the subsystem that performs face recognition through real-time capture of the webcam or video device configured by querying the image database of the registered faces detailed in topic IV.3.2 was addressed.

The function “verificador”, shown in Figure 14, performs the activity of capturing the face through the webcam and queries the image bank as a scan, through arrays and numeric arrays using the numpy library, if there is any image like it, performing facial recognition. Finally, it searches for the identification number of the folder and stores it in the variable "idents" for comparison purposes by the system.

The numpy library was created in 2005 by Travis Oliphant, the NumPy project was based on the Numeric and Numarray projects with the goal of gathering the community around a single framework for processing arrays. Therefore, the NumPy package, named so because of the abbreviation Numerical Python, is an open-source library designed to perform operations on multidimensional arrays, friendly referred to as ndarray in this library [26].

```
def verificador():
    global recognizer, verifica, pessoas
    pessoas = os.listdir('USUARIO')
    idents = []
    rostos = []
    for i, p in enumerate(pessoas):
        i += 1
        for f in os.listdir(f'USUARIO/{p}'):
            img = cv2.imread(f'USUARIO/{p}/{f}', 0)
            rostos.append(img)
            idents.append(i)
    recognizer.train(rostos, np.array(idents))
```

Figure 14: Face recognition function from the image bank.
Source: Authors, (2022).

Another important function, regarding the registration of time clocking in the database was modeling the function "registrandoHora" using the datetime library that stores the values of the exact moment of day, month, year, and time; and the system performs the calculation of the amount of extra hour and minute, being identified by the user's license plate variable. This process is done by Figure 15.

```
def registrandoHora(matriculaBD):
        tempo = dt.datetime.now()
        diaBD = int(tempo.day)
        anoBD = int(tempo.year)
        mesBD = int(tempo.month)
        horaBD = tempo.time()
        comando = f'INSERT INTO banco' \
                  f'(dia, mes, ano, hora, idEstagioReg, BH_horario, BH_hora, BH_minuto) VALUES' \
                  f' ({diaBD}, {mesBD}, {anoBD}, "{horaBD}", {matriculaBD}, "{horaFormula}", {qtdH}, {minuto_total})'
        cursor.execute(comando)
        conexao.commit()
```

Figure 15: Function for time clocking recording.
Source: Authors, (2022).

Another library used was datetime which for Santos [27] is an object type in Python to deal with dates and times. Whenever you have a variable with days, months, years, hours, you have the chance to transform it into a datetime object and have access to a series of methods to work with it and transform it.

After storing the clocking record, the system sends a voucher via email to the trainee identified by facial recognition. The function responsible for this activity is the "Sendcomprovante", shown in Figure 16. The text that will be contained in the email is structured by the "corpo_email" variable.

```
def SendComprovante(nomeBD, Email):
    corpo_email = f"""
    <p>{saudacao}, {nomeBD}.</p>
    <p>Segue seu comprovante de registro no sistema</p>
    <p>Dia: {diaBD}/{mesBD}/{anoBD}</p>
    <p>Hora: {horaBD}</p>
    """
```

Figure 16: Function to send voucher via email.
Source: Authors, (2022).

The text in Figure 16, the body of the email is sent to the recipient using the EmailMessage library which according to [28], provides the core functionality to set and query header fields, to access message bodies and to create or modify structured messages. Alongside the previous library is smtplib which [29] defines as used for sending email, dictates how the email will be formatted, encrypted and transmitted through online servers in addition to other details that the computer uses.

With the EmailMessage and smtplib library you can define which sender and recipient the system will send the voucher to. Figure 17 shows the process of defining the email subject, the sender, which will always be the address "sistema.sejusc@gmail.com", and the recipient, which will be the registered email address of the identified trainee.

```
msg = email.message.Message()
msg['Subject'] = f"Comprovante Registro ({tempo})"
msg['From'] = 'sistema.sejusc@gmail.com'
msg['To'] = Email
password = 'bkqxxzbbdkqaxpth'
msg.add_header('Content-Type', 'text/html')
msg.set_payload(corpo_email)
s = smtplib.SMTP('smtp.gmail.com: 587')
s.starttls()
s.login(msg['From'], password)
s.sendmail(msg['From'], [msg['To']], msg.as_string().encode('utf-8'))
```

Figure 17: Process of sending the e-mail.
Source: Authors, (2022).

In short, after all the creations of the functions discussed in this topic, the facial recognition subsystem, when in operation, performs the real-time capture of the webcam and checks the image bank to see if there are identical faces stored. If the system detects the trainee's registered frames, the trainee's information is searched in the database, the time clocking is registered through the function "registrandoHora" (Figure 15) and the proof of registration is sent through the function "sendComprovante" (Figure 16). The described process is structured and performed by the decision command depicted below in Figure 18.

```
if verifica:
    dados = userBanco()
    idf, conf = recognizer.predict(resize)
    idEstagio: int = pessoas[idf - 1]
    for dados in dados:
        if int(idEstagio) == int(dados[0]):
            nomeBD = dados[1]
            emailBD = dados[5]
            matriculaBD: int = dados[0]
            registrandoHora(matriculaBD)
            Email = f'{emailBD}'
            SendComprovante(nomeBD, Email)
```

Figure 18: Trainee identification for time clocking recording.
Source: Authors, (2022).

IV.3.4 Web development for querying time clocking registration

The web development was not the project's goal, but its rationale was commented on, because it serves as a consultation of the user's (trainee) time clocking record, being this a "bonus" function adhered to the system. This was based on HTML 5, CSS, PHP, JAVASCRIPT, Xampp.

The result of this development step can be seen in the following sequence of pictures, where user 2 was used as an example.

The layout of the login screen for supervisor and trainee access can be seen in Figure 19 below. It was designed to be objective and intuitive, so it has only two fields to be filled in: user and password; and a login button.

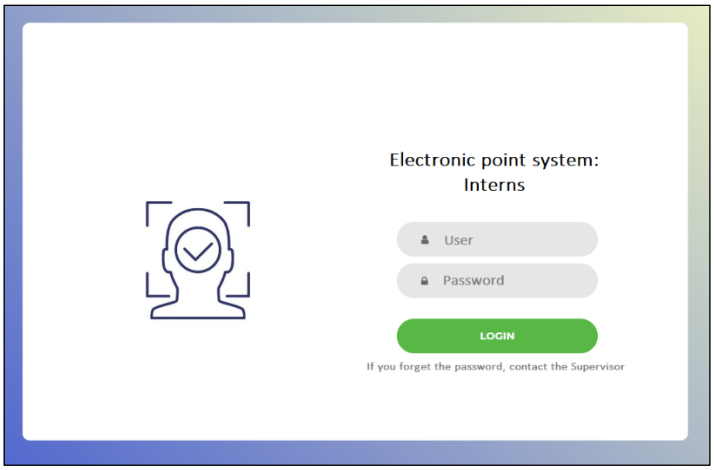


Figure 19: Login Screen.
Source: Authors, (2022).

After logging in, the home screen is shown, in Figure 20, and is composed of three buttons (dashboard) (located in the central part of the screen): "Trainees", "clocking time registered" and "Users". In addition, there is a yellow column with other buttons: "home", "users", "trainee", "timesheet", "about" and "logout".

It should be noted that the screens shown below are seen when the supervisor logs into his account, so he has access to all possible and necessary information.

Furthermore, in the bottom right corner of the button there is a variable that refers to the amount of existing records (registered) in the database for each field, i.e there are 2 users, there are 11 registered clocking times in total and there are 3 trainees in total. This is updated automatically.

Figure 20: Home Screen.
Source: Authors, (2022).

Accessing the "users" button (either through the yellow column or the dashboard) shows a list of the users that are registered in the system, i.e., trainee and supervisor. In addition, information such as registration, user type and user level are also shown. See Figure 1 below.

Figure 21: List of users.
Source: Authors, (2022).

Following the yellow column, clicking on the "trainee" button shows a list with information such as registration, name, gender, shift and e-mail regarding the registered trainee. You can also change the registration data, by clicking on the "pencil" icon, and delete a trainee, by clicking on the "trash can" icon. See Figure 22 below.

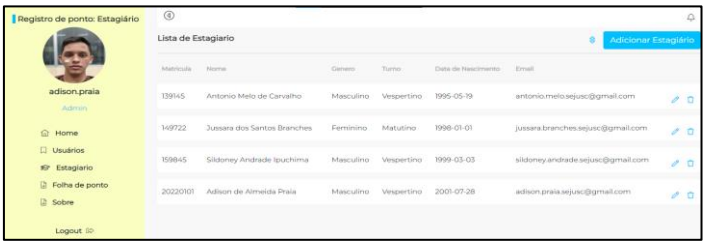


Figure 22: List of registered trainees.
Source: Authors, (2022).

The next screen, Figure 23, is visualized by clicking on the button "Time Sheet", on this page the supervisor can access not only the report of the record of all the trainees by clicking on the icon "eye" located in the View column, but also the information. However, this is the only option that appears for the trainee, i.e., this is where he can check his time clocking report individually.

Figure 23: Trainee's list for time clocking consult.
Source: Authors, (2022).

In the last screen, Figure 24, after clicking on the view icon, all the clocking records done by the trainee are shown. It has the following data: registration, name, day, month, year, hours, overtime, extra minutes and total in time format.

Figure 24: Trainee's time clocking register screen.
Source: Authors, (2022).

IV.4 TESTING THE SYSTEM USING THE USERS' FACE DATABASE

In this item, the process of the system's operation with the users' face database was presented, as well as the tests and errors that were essential to verify the product's limits and, finally, to implement it in the ITM.

IV.4.1 System operation process

The system, at first, trains the trainee's face and generates not only the folder with the user's name, but also 100 frames of his face. These have to be transformed to an easily identifiable shade, so they can't be colored, so all were converted to gray as can be seen on Figure 25. In this example, there is user 2.

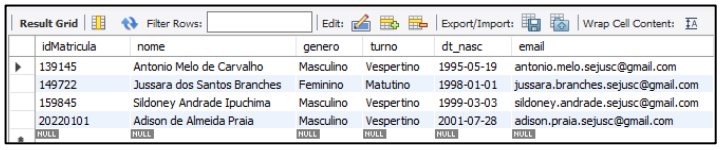
Figure 25: Trainee's clocking register screen.
Source: Authors, (2022).

The step-by-step process of registering a user (trainee) to the system was divided into 6 steps that can be seen below.

STEP 01 - New trainee registration form: in this window it is necessary to fill in the fields with the registration number, full name, date of birth, gender, e-mail and shift. Then, just click on the register customer button, and the new trainee is added to the database.

Figure 26: Registering the trainee profile.
Source: Authors, (2022).

STEP 02 - The data entered on the form are registered in the MySQL Workbench database in the table "estagiario". In which the "idMatricula" is mandatory, as it is a unique attribute, non-repetitive, used to identify each trainee.



Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

idMatricula	nome	genero	turno	dt_nasc	email
139145	Antonio Melo de Carvalho	Masculino	Vespertino	1995-05-19	antonio.melo.sejusc@gmail.com
149722	Jussara dos Santos Branches	Feminino	Matutino	1998-01-01	jussara.branches.sejusc@gmail.com
159845	Sildoney Andrade Ipuchima	Masculino	Vespertino	1999-03-03	sildoney.andrade.sejusc@gmail.com
20220101	Adison de Almeida Praia	Masculino	Vespertino	2001-07-28	adison.praia.sejusc@gmail.com

Figure 27: Trainee data.
Source: Authors, (2022).

STEP 03 - Registering faces for storage in the image library: in this window a blue frame is shown that is limited to the face of user 2. Besides, there are two options: press the 'C' key on the keyboard to register 100 frames of the face and the "e" key that closes the system.

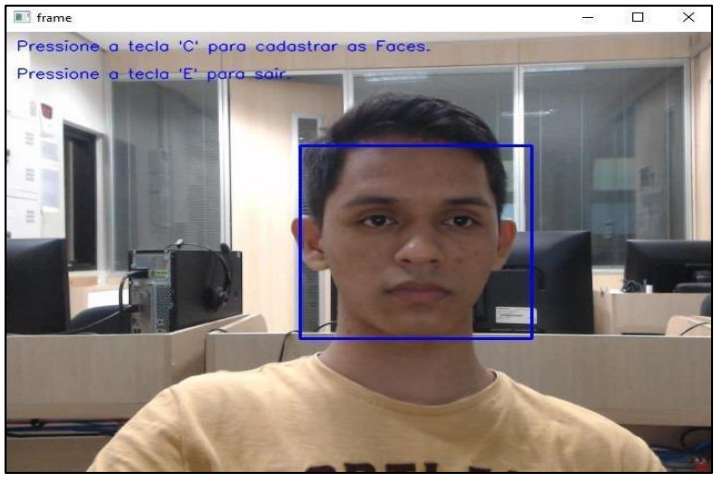


Figure 28: Face registration screen.
Source: Authors, (2022).

STEP 04 - Facial Recognition through Webcam: Checks to see if it contains an identical frame in the image library and fetches the information from the database.

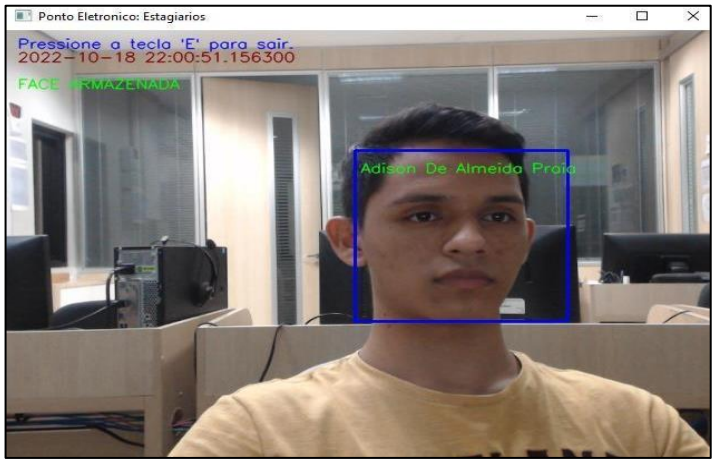


Figure 29: Face authenticator screen.
Source: Authors, (2022).

STEP 05 - Time clocking recording in the database. After performing facial recognition, the system records the time in the "timesheet_points" table. Each record is identified by the "idEstagioReg" column, which corresponds to the primary key of the "estagiario" table, making it possible to differentiate which record belongs to each trainee. The date was also segmented by day, month and year, giving the option to separate the data by a certain period of time.

idRegistro	dia	mes	ano	hora	data	idEstagioReg
21	18	10	2022	21:58:07	2022-10-18 21:58:07	20220101
22	18	10	2022	22:06:41	2022-10-18 22:06:40	18102022
23	18	10	2022	22:07:44	2022-10-18 22:07:43	20220101
24	18	10	2022	22:17:54	2022-10-18 22:17:54	20220101
25	18	10	2022	22:18:02	2022-10-18 22:18:01	20220101
26	18	10	2022	22:18:08	2022-10-18 22:18:07	20220101
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Figure 30: Table "time_sheet".
Source: Authors, (2022).

STEP 06 - Finally, after the registration of the time clocking record in the database, the system sends a clocking record voucher to the trainee's registered email address. This voucher is sent automatically by the system, and is always sent to the address: sistema.sejusc@gmail.com.

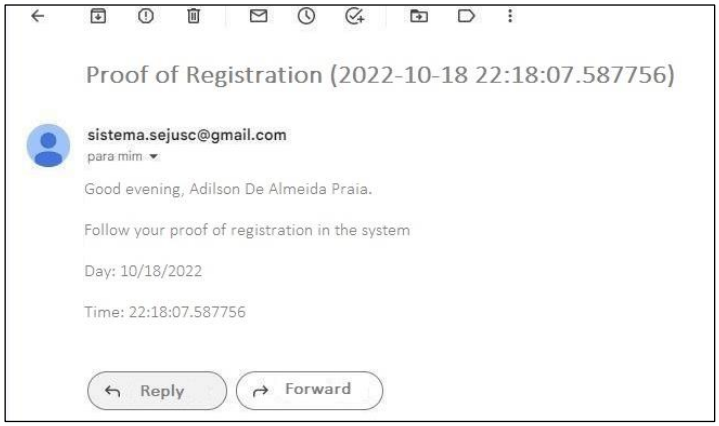


Figure 31: Proof of Registration via E-mail.
Source: Authors, (2022).

IV.4.1 Tests and Errors

Starting the tests, the registration of "user 1", "user 2" and "user 3" were done with a low quality webcam (VGA) and in a place with poor lighting. Because of this, an error was observed when performing facial recognition for each of the users. User 1" had the correct name, while "user 2" had the name of "user 1". See the following comparison of the error between Figure 32 and Figure 33.

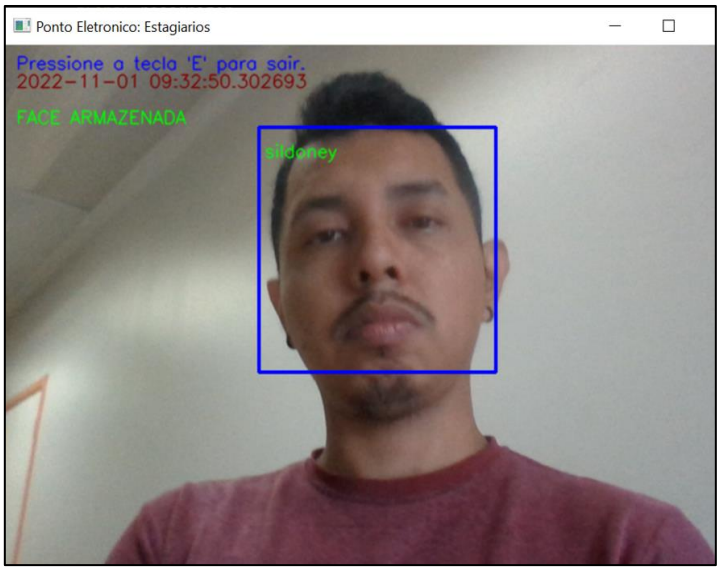


Figure 32: User 1 with correct name.
Source: Authors, (2022).

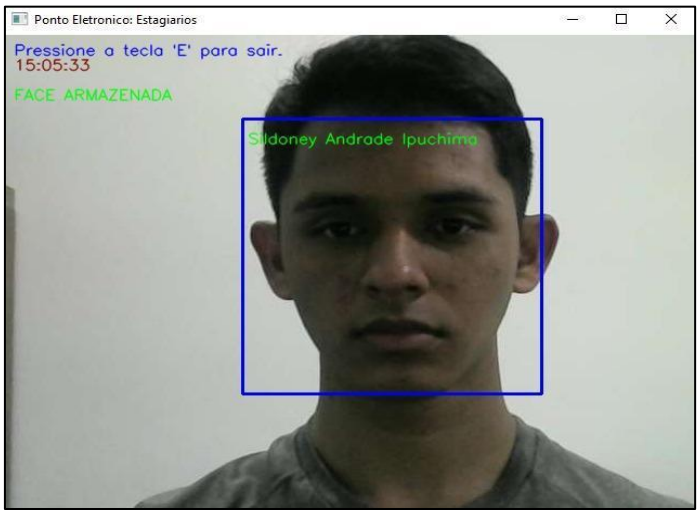


Figure 33: User 2 with user name 1.
Source: Authors, (2022).

The error occurred due to the mentioned conditions, and when registering, the frames were generated with low quality, causing the user's face traces to look similar. Take, for example, the "user 1" frames folder.

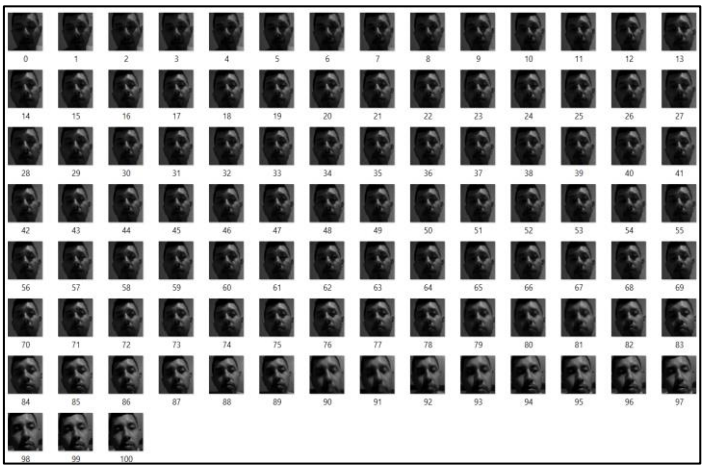


Figure 34: Folder with frames from user 1 with low quality.
Source: Authors, (2022).

IV.5 IMPLEMENTATION OF THE SYSTEM IN THE ITM OF SEJUSC

Joining Biometrics, Artificial Intelligence, Data Structuring, Programming Language, and Database it was possible to create the project, that is, the system was developed based on available technologies and easy access to information.

With the integration of the electronic time clock to ITM, it was possible to automate the frequency registration system that was

previously done manually, and to optimize the overtime accounting process.

The procedure was started by registering the trainees, adding their personal data. It was important to note that the institutional e-mail should be used, because the receipt is sent through there. And later, they did the face registration. So, every day, from Monday to Friday, when they arrived and left, each on their respective shift, they were able to register their attendance.

On the days when overtime was needed, the trainees logged 1 hour after their normal time off, this was the maximum amount of overtime agreed upon between the supervisor and the trainees. And their record report is seen on the web system, as well as being received by email.

Now, with the automated process, the supervisor has access to each trainee's attendance record report, where he can check the amount of overtime, where he can fairly offer the amount of days off.

V. CONCLUSIONS

The project was based on applied research, thus, it aimed to address the problems observed in the sector of Information Technology Management (ITM) located within the State Department of Justice, Human Rights and Citizenship (SEJUSC). One deficiency of the sector was related to the frequency control, while the other was the high demand of services having the need to work overtime and within this context were the interns.

As a result, along with the solution found by the ITM supervisor to offer time off to those trainees willing to work overtime, in a way to optimize and automate the working day, an electronic time off system was developed for trainees based on facial biometrics, Python language and database.

In addition, the methodology adopted was research of theoretical references, interviews, and data collection from the trainees, and also the materials used as low-cost technological tools were fundamental and essential to achieve the general objective. This was delimited in five specific objectives that were satisfactorily met.

The demand of the sector was analyzed, as well as the dynamics of the trainees' working hours, this being the first step taken within the methodology. In the sequence, the project's layout was created, thus following to the stage where the system and the database were developed. In view of this, arriving at the penultimate step, the system was tested through the trainees' library of faces. Finally, the implementation was done in the ITM sector of SEJUSC.

The development of the project was based on available technologies and the ease of access to information, and the following hypothesis was raised: by joining facial biometrics to the idea of an electronic time clock that emits the amount of days off would make the sector's attendance system optimized and safe?

In summary, it was possible to develop an electronic system that met the needs cited. This system, which is low cost, was implemented in the ITM and served to assist the supervisor who, now, with the report of the overtime bank, can offer optimally the correct amount of days off to the trainee. And, thus, making the time record automated.

In future research, it is recommended to improve the use of facial biometrics, where the recognition is limited to the face without the use of accessories, and also to look for better technologies regarding low light. In addition, it is suggested to switch or use voice biometrics together, so that the system can be activated without the need for contact and without using peripherals (mouse and keyboard), and perhaps making it faster.

The project developed was a great opportunity to complement and improve the academic training. In addition, it allowed students to put into practice the contents studied as a basis in the Computer Science course, such as Biometrics, Artificial Intelligence, Data Structures, Programming Languages and Databases. Applying in practice the concepts, fundamentals, definitions and software studied in the classes, and finally, understanding the importance of the elaboration of a scientific article and for the scientific community.

VI. AUTHOR'S CONTRIBUTION

Conceptualization: Adison Praia, Jussara Branches, Sildoney Andrade, Eliton Smith and Alexandra Amaro de Lima.
Methodology: Adison Praia, Jussara Branches, Sildoney Andrade, Eliton Smith and Alexandra Amaro de Lima.
Investigation: Adison Praia, Jussara Branches, Sildoney Andrade, Eliton Smith and Alexandra Amaro de Lima.
Discussion of results: Adison Praia, Jussara Branches, Sildoney Andrade, Eliton Smith and Alexandra Amaro de Lima.
Writing – Original Draft: Adison Praia, Jussara Branches, Sildoney Andrade, Eliton Smith and Alexandra Amaro de Lima.
Writing – Review and Editing: Eliton Smith and Alexandra Amaro de Lima.
Resources: Eliton Smith and Alexandra Amaro de Lima.
Supervision: Alexandra Amaro de Lima.
Approval of the final text: Adison Praia, Jussara Branches, Sildoney Andrade, Eliton Smith and Alexandra Amaro de Lima..

VII. REFERENCES

[1] J. de J. Lozoya-Santos, V. Sepúlveda-Arróniz, J. C. Tudon-Martinez, and R. A. Ramirez-Mendoza, "Survey on biometry for cognitive automotive systems," Cognitive Systems Research, vol. 55, pp. 175–191, Jun. 2019, doi: 10.1016/j.cogsys.2019.01.007.

[2] A. F. de Moraes, Segurança em redes: Fundamentos: Fundamentos, 1ª edição. Editora Érica, 2010.

[3] J. L. D. Moraes, "Controle de Acesso Baseado em Biometria Facial," p. 102, 2010. doi: http://repositorio.ufes.br/handle/10/4231.

[4] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," IEEE Transactions on Circuits and Systems for Video Technology, vol. 14, no. 1, pp. 4–20, Jan. 2004, doi: 10.1109/TCSVT.2003.818349.

[5] R. D. Labati, A. Genovese, E. Muñoz, V. Piuri, F. Scotti, and G. Sforza, "Computational Intelligence for Biometric Applications: a Survey," International Journal of Computing, pp. 40–49, Mar. 2016, doi: 10.47839/ijc.15.1.829.

[6] J. Buolamwini, V. Ordóñez, J. Morgenstern, and E. L. Miller, "Facial Recognition Technologies In The WILD," Ajl.org, 2022. [Online]. Available: https://www.ajl.org/federal-office-call (accessed Abr. 17, 2022).

[7] M. G. Galterio, S. A. Shavit, and T. Hayajneh, "A Review of Facial Biometrics Security for Smart Devices," Computers, vol. 7, no. 3, Art. no. 3, Sep. 2018, doi: 10.3390/computers7030037.

[8] P. M. P. Reis, "Identificação, análise e avaliação de linguagens de programação adequadas ao desenvolvimento de agentes móveis multi-plataforma," Ipleiria.pt, 2014, doi: http://hdl.handle.net/10400.8/1357.

[9] H. Özmen. "Python Lecture Notes". p.2, 2022. [Online]. Available: https://www.academia.edu/49678628/PYTHON_PROGRAMMING_LANGUAGE (accessed May 06, 2022).

[10] M. G. Rashed and R. Ahsan, "Python in Computational Science: Applications and Possibilities," International Journal of Computer Applications, vol. 46, no. 20, pp. 26–30, May 2012.

[11] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," ACM Comput. Surv., vol. 35, no. 4, pp. 399–458, Dec. 2003, doi: 10.1145/954339.954342.

[12] "Home," OpenCV. [Online]. Available: https://opencv.org/ (accessed May 06, 2022).

[13] M. Marengoni and S. Stringhini, "Tutorial: Introdução à Visão Computacional usando OpenCV," Revista de Informática Teórica e Aplicada, vol. 16, no. 1, Art. no. 1, 2009, doi: 10.22456/2175-2745.11477.

[14] G. Bradski and A. Kaehler, Learning OpenCV: Computer Vision with the OpenCV Library. O'Reilly Media, Inc., 2008.

[15] J. Medeiros, Y. Alencar Lima, e E. De Macedo Rodrigues, "Comparação De Ferramentas De Modelagem De Banco De Dados: Open Source", SIEPE, vol. 10, nº 2, mar. 2020.

[16] "Relacionamento 1-1, 1-N e N-N com Django," Treinaweb.com.br, 2022. [Online]. Available: https://www.treinaweb.com.br/blog/relacionamento-1-1-1-n-e-n-n-com-django (accessed Sep. 22, 2022).

[17] J. C. Cantú, "Sistema para controle de ponto de funcionários," Utfpr.edu.br, 2011, doi: http://repositorio.utfpr.edu.br/jspui/handle/1/15549.

[18] J. M. Pinheiro, Biometria nos Sistemas Computacionais: Você é a Senha. Rio de Janeiro: Editora Ciência Moderna Ltda, 2008.

[19] M. C. D. Assis. Metodologia do Trabalho Cientifico. In: Evangelina Maria B. de Faria: Ana Cristina 5. Aldrigue. (Org). Linguagens: usos e reflexões, 3. Ed. João Pessoa Editora Universitária UFPB. 2009.

[20] SEJUSC, Secretaria de Estado de Justiça, Direitos Humanos e Cidadania. 2022. Accessed in: June 2022. Available in: http://www.sejusc.am.gov.br/

[21] A. S. Lima, Uml 2, 0 do Requisito a Solução. Erica, 2008.

[22] M. T. and Inovation, "Design Thinking - Inovação em Negócios". [Online]. Available: https://www.livrodesignthinking.com.br (accessed Oct. 14, 2022).

[23] J. Otávio, "Tkinter: Interfaces gráficas em Python," DevMedia, 2016. [Online]. Available: https://www.devmedia.com.br/tkinter-interfaces-graficas-em-python/33956 (accessed Oct. 14, 2022).

[24] R. Python, "Python and MySQL Database: A Practical Introduction – Real Python.". [Online]. Available: https://realpython.com/python-mysql/ (accessed Aug. 07, 2022).

[25] "Módulo de sistema operacional em Python com exemplos – Acervo Lima". [Online]. Available: https://acervolima.com/modulo-de-sistema-operacional-em-python-com-exemplos/ (accessed Sep. 15, 2022).

[26] "Numpy Python: O que é, vantagens e tutorial inicial - Blog - Harve". [Online]. Available: https://harve.com.br/blog/programacao-python-blog/numpy-python-o-que-e-vantagens-e-tutorial-inicial/ (accessed Oct. 04, 2022).

[27] G. Santos, "Como Manipular Datetime no Python," Data Hackers, Oct. 08, 2021. [Online]. Available: https://medium.com/data-hackers/como-manipular-datetime-no-python-578f07b72920 (accessed Oct. 10, 2022).

[28] "email.message: Representing an email message — Python 3.11.0 documentation". [Online]. Available: Python.org, 2022. https://docs.python.org/3/library/email.message.html (accessed Sep. 24, 2022).

[29] AsttroDev, "TUTORIAL: Enviando e-mail utilizando a biblioteca SMTPLib do Python," Medium, Jun 17, 2018. [Online]. Available: https://medium.com/@asttrodev/tutorial-enviando-e-mail-utilizando-a-biblioteca-smtplib-do-python-2adfb3768a08 (accessed Oct. 14, 2022).