**CHAPTER ONE**

**INTRODUCTION**

**1.1 BACKGROUND TO THE STUDY**

Organizations of all sizes use **time and attendance** systems to record when employees start and stop work, and the department where the work is performed. However, it’s also common to track meals and breaks, the type of work performed, and the number of items produced. In addition to tracking when employees work, organizations also need to keep tabs on when employees are not working. Vacation time, compensation time, Family and Medical Leave Act (FMLA) time, and jury duty must be recorded. Some organizations also keep detailed records of attendance issues such as who calls in sick and who comes in late.

A time and attendance system provides many benefits to organizations. It enables an employer to have full control of all employees working hours. It helps control labor costs by reducing over-payments, which are often caused by transcription error, interpretation error and intentional error. Manual processes are also eliminated as well as the staff needed to maintain them. It is often difficult to comply with labor regulation, but a time and attendance system is invaluable for ensuring compliance with labor regulations regarding proof of attendance.

Companies with large employee numbers might need to install several time clock stations in order to speed up the process of getting all employees to clock in or out quickly or to record activity in dispersed locations. In the business world of today we all know one simple truth…TIME IS MONEY! We work to keep the amount of time it takes to complete even the simplest tasks down to the minimum.

**1.2 PROBLEM STATEMENT**

Niger Insurance PLC makes use of a pen and notebook to check attendance of its employees. With this system, employees write their names, time of arrival and signature in ruled columns in the notebook. This data is used to process the salary of employees given account of times when they were late and absent. The issue with this system is that employees tend to falsify entries and indirectly

**1.3 PROJECT AIM**

The aim of this project is to develop an automated employee attendance system using face recognition system.

**1.4 PROJECT OBJECTIVES**

The objectives of this project are:

* Creating a database that contains attendance information of the employees
* Linking this system to an existing employee management system
* Capturing live feed from camera to record attendance times
* Linking the information captured by the feed to the database for accuracy of results.

## 1.5 SIGNIFICANCE OF STUDY

A lot more goes into keeping proper time and attendance records than just recording when employees clock in and out each day. From correctly tracking hours logged and paid time off to making sure certain insurance requirements are being met, there are a wide variety of advantages to using time and attendance systems. One of the largest benefits is the added financial insight into your business.

A time and attendance system is important for businesses because it provides advanced tracking for job-costing and labor distribution, It also provides enhanced views of the organization's key performance indicators, including hours and wages, employee counts and scheduled/working status.

Time and attendance systems can also be a huge cost-saving tool by providing a detailed look at labor costs and where changes can be made. Businesses can save as much as 5 percent of their payroll costs by automating their time and attendance system.

"You can see time and attendance trends that could be costing you money, so you can control Gthem now and into the future.

These systems also improve employee productivity because it frees some staffers of the time-consuming task of entering each employee's daily hours into the payroll programs.

The project will prevent data diddling by employees.

This project will form a project template that is replicable in other institutions of similar standing.

* 1. **ORGANIZATION OF THE STUDY**

Thisproject work is structured as follow:

Chapter one introduces the topic of Employee attendance and its relevance to any company and how it can reduce cost and increase productivity. It further discusses the benefits of implementing this project.

In chapter two, we will describe a variety of facial recognition algorithms and how they are interfaced. We will then move on to review recent works related to employee attendance and discuss some of their limitations.

Chapter three will start with the analysis of Niger Insurance PLC’s infrastructure. Based on the analysis we will be able to integrate employee attendance systems. Furthermore we will document the step by step procedures we used to fully implement this project.

Chapter four is about testing and observing our newly implemented system to verify if we met our stated objective.

Chapter five will draw conclusion on the work and make useful recommendations.

**CHAPTER 2: LITERATURE REVIEW**

**2.1 INTRODUCTION**

Attendance as defined by the Merriam-Webster dictionary:

: the number of people present at an event, meeting, etc.

: the act of being present at a place

: a record of how often a person goes to classes, meetings, etc.

Companies with large [employee](http://en.wikipedia.org/wiki/Employee) numbers might need to install several [time clock](http://en.wikipedia.org/wiki/Time_clock) stations in order to speed up the process of getting all employees to clock in or out quickly or to record activity in dispersed locations.

Depending on the supplier, identification method and number of clocking points required, prices vary widely. A time and attendance system protects a company from payroll fraud and provides both employer and employees with confidence in the accuracy of their wage payments all while improving productivity.

Many time and attendance software is now provided through cloud-based software as a service (SaaS) hosted solutions. Web-based solutions offer a number of advantages to organizations of all sizes, including reduced implementation costs, fewer maintenance and support concerns, as well as instant updates and upgrades.

## Manual systems

Manual systems rely on highly skilled people laboriously adding up paper cards which have times stamped onto them using a time stamping machine such as the Bundry Clock. Time stamping machines having been in use for over a century are still used by many organizations as a cheaper alternative to time and attendance software.

## Automated systems

Automated time and attendance systems can use electronic tags, barcode badges, magnetic stripe cards, biometrics (vein reader, hand geometry, fingerprint, or facial), and touch screens in place of paper cards which employees touch or swipe to identify themselves and record their working hours as they enter or leave the work area. The recorded information is then ideally automatically transferred to a computer for processing although some systems require an operator to physically transfer data from the clocking point to the computer using a portable memory device. The computer may then be employed to perform all the necessary calculations to generate employee timesheets which are used to calculate the employees' wages. An automated system reduces the risk of errors that are common in a manual system, and allows the workforce to be more productive instead of wasting time on tedious administrative tasks.

We know that different businesses have diverse time-and-attendance system needs. We have researched and reviewed dozens of time and attendance systems and came up with the ones we think are best for a variety of business types. Here is a roundup of our best picks and an explanation of how we chose them.

**Best time and attendance system for small businesses: stratustime**

Stratustime offers everything a small business needs in a time and attendance system. The cloud-based system has the most comprehensive time-tracking options we found, as employees can clock in and out via traditional time clocks, computers with Internet access, mobile devices and telephones. The system also manages paid time off, generates employee schedules, monitors overtime hours and integrates with a wide range of payroll services. We were also very impressed with the system's easy-to-use online portal, the company's customer service and the system's affordable monthly cost.

## ****Best time and attendance system for very small businesses: uAttend****

The uAttend time and attendance system works perfectly for a very small business. A Web-based system, it requires no special software to load or hardware to install, and is very flexible, letting employees clock in with time clocks, Web browsers, mobile devices and telephones. uAttend's time clocks are plug-and-play ready, send data in real time to the systrem, include lifetime guarantees, and are among the cheapest we found. Besides being able to log when employees come and go each day, small businesses will appreciate that the system also manages paid time off, can track how long employees work on specific projects and can create office-wide schedules.

## ****Best time and attendance system for businesses with a mobile workforce: TSheets****

TSheets provides just what businesses with a mobile workforce need from a time attendance system. Besides being able to clock employees in and out with laptop computers, smartphones, telephones, text messages and Twitter, the system also records exactly where workers start and end their shifts, as well as their locations throughout the day. These are critical tools that allow businesses to easily monitor and track their mobile employees. The pricing is extremely affordable, and the customer service is among the best we encountered.

## ****Our methodology we used to gather ideas****

To determine the best time and attendance systems, we started by listing all of the vendors that have a good reputation online (i.e., services that were favorably and consistently reviewed by other websites). Then, we interviewed small business owners to discover new ones to add to our list.

We ultimately settled on 11 time and attendance systems to consider as best picks. They were: Acrotime from Acroprint, ClockVIEW from Acumen, Kronos' Workforce Ready, stratustime from nettime solutions, TimeAttend from Replicon, TimeClick from Hawkeye Technology, TimeForce from Infinisource, TimeStar from Insperity, TSheets, uAttend from Processing Point and Webtimeclock.

Next, we researched each provider by investigating its services, watching tutorials and how-to videos, trying out the system when possible, and reading user comments. We also considered the pricing that was listed on these services' websites. In all, we analyzed each service based on the following factors:

* Cost
* Time-tracking capabilities
* Accrual options
* Overtime monitoring
* Scheduling capabilities
* Available reports
* Ease of use
* Customer service
* Integration options
* Mobile access

Here is a full list of time and attendance system providers and a summary of what each company claims to offer. This alphabetical list also includes our picks for best services.

**AcroTime** – Acroprint's AcroTime is a Web-based system offered in both standard and platinum versions, neither of which requires the installation of special software. With the system, employees can clock in and out via their computers, biometric time clocks and badge terminals. Features exclusive to the platinum version include the ability to track accruals and a system to schedule time-off requests.

**ADP Small Business Time and Attendance**  – ADP's Small Business Time and Attendance system is Web-based and offers multiple ways of collecting employee time-and-attendance information, including electronic or online time sheets, badge-card time clocks, biometric time clocks and wireless devices, or via an interactive voice response system. The software automatically adds up employee hours and feeds them directly to the business's payroll and tax solutions. ADP's system allows managers to create work schedules and then track absences, late clock-ins, long lunches and early departures. ADP's solution also allows employees to request and track their time off. The software integrates with a variety of payroll, tax-service and human-resources solutions.

**ClockVIEW** – Acumen Data Systems' ClockVIEW is a Web-based time-and-attendance system. The software gives businesses a complete overview of when their employees are working, taking vacation and out sick. Managers can set notifications for when employees take time off. ClockVIEW also allows employees to clock and in and out, submit time-off requests and bid for shifts when other employees call in sick. In addition, the software boasts an instant-messaging feature and the ability to limit when employees can clock in by time of day, employee schedule and location.

**Easy Clocking** – Easy Clocking is a Web-based time-and-attendance system that offers employees the opportunity to clock in and out via time clocks, fingerprint scanners and their computers. Each of the time-clock options works in conjunction with the Easy Clocking time-and-attendance software. The software automatically calculates total hours worked, including overtime, vacation, sick days and holidays. The software is also designed to help reduce employee time theft by applying rounding rules, restricting schedules and eliminating the ability for co-workers to clock other employees in. Additionally, the Easy Clocking time-and-attendance system is built to automate the payroll process by allowing management to transfer time data to all payroll providers.

**EmpCenter** – Designed to meet the needs of large and growing organizations, Workforce Software's EmpCenter Time & Attendance is an enterprise-grade, cloud-based solution that captures time and attendance, as well as labor data, and automates complex pay rules, entirely through configuration.

**Kronos Workforce Ready** – Kronos Workforce Ready Time Keeping is a cloud-based solution that allows small businesses to fully automate their time-and-attendance information. Employees can clock in and out via mobile devices, computers and time clocks. Workforce Ready can also manage a variety of employee types, including those who are hourly, salaried and contracted. The system has a GPS feature that automatically records mobile employees' locations when they clock in and out. Workforce Ready also gives employees the ability to check their work schedules, review benefit accruals and request time off. Additionally, the software integrates with a variety of other Kronos Workforce Ready applications, such as human resources, payroll and tax solutions.

**TimeAttend** – Replicon's TimeAttend is a cloud-based time-and-attendance system. With its configurable and intuitive time sheets, the software allows businesses to track time for any employee type. TimeAttend also automates the time-off management process with self-service time-off request capabilities and accrual calculations. To help streamline the payroll process, the system enables project managers to store information about employees' schedules, overtime and pay rates. Additional features include automatic emails and alerts that can be set up to ensure the requests for paid time-off are submitted and approved on time.

**TimeClock Plus** – TimeClock Plus offers a variety of time-attendance systems, including server-based, Web-based and software-as-a-service options. The TimeClock Plus Web Edition is a browser-based time-and-labor management application that provides employee time-and-attendance tracking on the Web. To use the system, employees open their Web browsers, log on to the designated Web page and perform clock transactions. Many of the TimeClock Plus options also include mobile features that allow employees to clock in and out via their smartphones. TimeClock Plus also offers a Small Business Edition that runs on the business's own network. With this version, employees can clock in and out from their own computers, while managers have access to a "Who's Here" feature that shows which employees are currently working.

**TimeForce II**– TimeForce II from Infinisource is a cloud-based system designed to manage employee time and attendance. It gives businesses of all sizes the ability to easily and accurately collect and organize their employee time data. The system keeps track of the time employees work, as well as their holiday pay, overtime, and other time-and-attendance records. Employees have the option of using a Web browser or a simple time card to clock in and out. The TimeForce system offers an array of features, including time-off request notifications, time-off denial notes, time-card error alerts and role-based security options, which control what each employee is able to see and access. There is also a mobile application that allows employees to punch in and out via their smartphones.

**TSheets** – TSheets is designed as a way for businesses to both track and manage employee time. With the software, employees can clock in and out on their desktop or laptop computers, as well as from their mobile phones. There is also a GPS option that lets businesses automatically see where its mobile employees are starting and ending their days. In addition, TSheets lets employees clock in and out in real time, enter their time manually, and allocate time to custom fields, projects or tasks. The system allows administrators to manage an employee's time off, sick days and holidays. Additional features include overtime alerts, time-sheet approvals and the ability to see which employees are working at any given time. TSheets accounts can be configured to be DCAA and DOL compliant.

**uAttend** – Created by Processing Point, uAttend is a Web-based time-and-attendance system that allows employees to clock in and out from time clocks, Web browsers, mobile devices or telephones. The software tracks vacation, sick days and holidays; prevents employees from clocking in or out early or late to earn overtime; and gives managers the ability to see which employees are working at any given time. The software also easily integrates with nearly all payroll-processing systems, such as QuickBooks, Paychex, ADP and Sure Payroll. uAttend offers subscription options for businesses with workforces ranging from fewer than nine employees to more than 100 employees.

# Choosing a Time and Attendance System

While you might see your employees every day, logging their hours each week can be a tremendous undertaking. You must know when workers clock in and out, add up the number of hours they work, determine how much paid time off they've taken and how much they have left.

Instead of spending countless hours performing all of those calculations manually, many businesses are moving to digital time and attendance systems that automatically track and compute the data.

"A true time and attendance system tracks employee hours worked in real time from virtually anywhere, and automatically manages paid time off, vacation and time sheet approvals," Matt Rissell, CEO of [TSheets Time Tracking](http://www.tsheets.com/), told Business News Daily.

Manual time-tracking methods like punch clocks and paper time sheets are often cumbersome and error-prone when it comes to making sure employees are paid the right amount each month. Today's digital systems rely on a variety of technologies to make the time-collection process much smoother.

"More sophisticated and innovative [time collection] methods have become available in recent years to capture time, including intelligent time clocks, mobile devices, GPS location tracking, magnetic swipe cards, biometrics and touch screens," said Frank Moreno, director of product marketing at [Kronos Inc](http://www.kronos.com/). "In an ideal configuration, information is [then] automatically transferred to a system that enables proper payroll processing."

In addition to logging employee hours each day, time and attendance systems can:

* Manage paid time off
* Administer employee schedules
* Monitor overtime expenses
* Provide detailed reports on labor expenses
* Integrate with payroll services
* Track mobile employees via GPS locators
* Help ensure compliance with federal, state and local labor laws

**Choosing a Time and Attendance System for small businesses:**

While time-tracking systems were previously used only by large corporations that could afford the expensive costs, the cloud has made these services much more affordable for small businesses. Since these systems use the Software-as-a-Service (SaaS) model, small businesses doesn't have to worry about the expensive costs of ensuring they have the right computer systems in place to run the software. The cloud model allows businesses to log in from any Internet browser and immediately start using the system.

Armando Ortega, director of product marketing for time-tracking provider Replicon, said many small businesses are attracted to this style because it can easily be expanded as their operation grows.

"You pay for as much as you use," Ortega said. "If they have more employees, they can just buy more user licenses, and if for some reason there is a need to downsize, you have the flexibility to do that, too."

Jim Wacek, president of the time and attendance division for Insperity, which offers the TimeStar time and attendance system, said cloud-based systems are an additional advantage for businesses that don't want to host the application on their own network.

"Having this system helps reduce costs because companies don't have to purchase and maintain their own servers and IT infrastructure," Wacek said.

While large companies can expect to pay thousands of dollars for an on-premise time and attendance system, cloud-based options are significantly cheaper. Monthly fees can range from as low as $20 a month, to as much as several hundred, depending on the provider you choose and how many employees you have. The only other costs are for any time clocks you purchase. For example, biometric time clocks, which automatically punch employees in and out, can cost anywhere from several hundred to several thousand dollars each.

## Why a company will need a time and attendance system

Any business with one or more hourly employees can benefit from a time and attendance system.

Questions one should ask when considering whether a time and attendance system is necessary for the organization include:

* Does your company have hourly employees?
* Do you have a mobile workforce?
* Do your employees work at multiple jobsites or in remote locations?
* Is employee accountability and verification of employee location critical?
* Do you need to know who's working, in real time?
* Do you need to ensure employees are paid accurately for the time they work?
* Have paper time sheets or punch timecards have become too difficult to manage?
* Do you need a way of measuring the productivity of your organization?
* Is accurately measuring the cost of goods or projects a challenge?
* Do you need a way to accurately calculate job-costing and forecasting numbers?
* Do you need a way to ensure overtime, PTO and vacations are calculated correctly, kept in check and paid out properly?

The more questions you answer yes to, the better fit a time and attendance system will be for you.

## The features to look for in a time and attendance system

With so many time and attendance systems on the market, it can be difficult to know which one is right for your business. When shopping for a system, experts agree there are a number of things you should look for.

* **Keeps accurate time:** The best time and attendance systems ensure employers are paying their employees for the actual time they work, not the hours they're scheduled to work. Even the best employees may be rounding off their times or making unintentional mistakes.
* **Tracks all time issues:**Make sure you choose a system that takes care of all of your business's scheduling needs, not one that just lets employees punch in and out. This includes managing employee schedules and handling all paid-time-off calculations.
* **Integrates with programs:**Choose a system that integrates with the programs, applications and software, such as HR and payroll programs, that you are already using.
* **Monitors overtime expenses:** Make sure the system you select can help you reduce overtime costs. setting up overtime alerts can notify you before incurring the cost of overtime pay, and minimize unnecessary or unprofitable overtime hours.
* **Cloud-based:**Time and attendance systems hosted in the cloud are the easiest to implement and manage and reduce capital and maintenance expenses.
* **Reduces administrative time:**Search for a system that's easy to use and allows human resources employees to focus their time on issues that aren't payroll-related, .
* **Provides real-time information:**You want a time and attendance system that provides detailed information in real time, because businesses need labor information when it can be impacted, not after the fact.
* **Mobile:**Since many employees and managers work outside the office, you want a system that has mobile capabilities so it can be accessed wherever they are.

**CHAPTER 3**

**SYSTEM SPECIFICATION AND DESIGN**

**3.1 INTRODUCTION**

This section introduces the system requirements specification for the face recognition attendance system for Niger Insurance PLC employees.

**3.2 SYSTEM SCOPE**

The software is to provide employers an easier and fool-proof way to mark attendance of the employees. The software will allow administrators enroll employees. The software will allow employees have their face captured for attendance marking purposes. Only administrators will have direct manipulation rights to the system. Employees will have no other interaction with the system except for having their faces captured whilst they are entering the office. It will be a console application and would require maximum uptime due to its sensitive purpose.

**3.3 METHODOLOGY**

In software engineering, a **software development methodology** (also known as a **system development methodology**, **software development life cycle**, **software development process**, **software process**) is a division of software development work into distinct phases (or stages) containing activities with the intent of better planning and management. It is often considered a subset of the systems development life cycle. The methodology may include the pre-definition of specific deliverables and artifacts that are created and completed by a project team to develop or maintain an application.

Common methodologies include waterfall, prototyping, iterative and incremental development, spiral development, rapid application development, extreme programming and various types of agile methodology. Some people consider a life-cycle "model" a more general term for a category of methodologies and a software development "process" a more specific term to refer to a specific process chosen by a specific organization. For example, there are many specific software development processes that fit the spiral life-cycle model.

**Examples of methodologies**:

**Prototyping**

[Software prototyping](http://en.wikipedia.org/wiki/Software_prototyping), is the development approach of activities during [software development](http://en.wikipedia.org/wiki/Software_development), the creation of prototypes, i.e., incomplete versions of the software program being developed.

The basic principles are:

* Not a standalone, complete development methodology, but rather an approach to handle selected parts of a larger, more traditional development methodology (i.e. incremental, spiral, or rapid application development (RAD)).
* Attempts to reduce inherent project risk by breaking a project into smaller segments and providing more ease-of-change during the development process.
* User is involved throughout the development process, which increases the likelihood of user acceptance of the final implementation.
* Small-scale mock-ups of the system are developed following an iterative modification process until the prototype evolves to meet the users’ requirements.
* While most prototypes are developed with the expectation that they will be discarded, it is possible in some cases to evolve from prototype to working system.
* A basic understanding of the fundamental business problem is necessary to avoid solving the wrong problems.

We decided to go with **Rapid application development** (**RAD**). This is because:

* Better Quality. By having users interact with evolving prototypes the business functionality from a RAD project can often be much higher than that achieved via a waterfall model. The software can be more usable and has a better chance to focus on business problems that are critical to end users rather than technical problems of interest to developers.
* Risk Control. Although much of the literature on RAD focuses on speed and user involvement a critical feature of RAD done correctly is risk mitigation. It's worth remembering that Boehm initially characterized the spiral model as a risk based approach. A RAD approach can focus in early on the key risk factors and adjust to them based on empirical evidence collected in the early part of the process. E.g., the complexity of prototyping some of the most complex parts of the system.
* More projects completed on time and within budget. By focusing on the development of incremental units the chances for catastrophic failures that have dogged large waterfall projects is reduced. In the Waterfall model it was common to come to a realization after six months or more of analysis and development that required a radical rethinking of the entire system. With RAD this kind of information can be discovered and acted upon earlier in the process.
* **Rapid application development** (**RAD**) is both a general term used to refer to alternatives to the conventional waterfall model of software development as well as the name for James Martin's approach to rapid development. In general, RAD approaches to software development put less emphasis on planning tasks and more emphasis on development. In contrast to the waterfall model, which emphasizes rigorous specification and planning, RAD approaches emphasize the necessity of adjusting requirements in reaction to knowledge gained as the project progresses. This causes RAD to use prototypes in addition to or even sometimes in place of design specifications. RAD approaches also emphasize a flexible process that can adapt as the project evolves rather than rigorously defining specifications and plans correctly from the start. In addition to James Martin's RAD methodology, other approaches to rapid development include Agile methods and the spiral model. RAD is especially well suited (although not limited to) developing software that is driven by user interface requirements. Graphical user interface builders are often called rapid application development tools.

## [http://upload.wikimedia.org/wikipedia/commons/thumb/5/5f/RADModel.JPG/320px-RADModel.JPG](http://en.wikipedia.org/wiki/File:RADModel.JPG)

## Fig 3.1 The James Martin RAD methodology

**The disadvantages of RAD include:**

* The risk of a new approach. For most IT shops RAD was a new approach that required experienced professionals to rethink the way they worked. Humans are virtually always averse to change and any project undertaken with new tools or methods will be more likely to fail the first time simply due to the requirement for the team to learn.
* Requires time of scarce resources. One thing virtually all approaches to RAD have in common is that there is much more interaction throughout the entire life-cycle between users and developers. In the waterfall model, users would define requirements and then mostly go away as developers created the system. In RAD users are involved from the beginning and through virtually the entire project. This requires that the business is willing to invest the time of application domain experts. The paradox is that the better the expert, the more they are familiar with their domain, the more they are required to actually run the business and it may be difficult to convince their supervisors to invest their time. Without such commitments RAD projects will not succeed.
* Less control. One of the advantages of RAD is that it provides a flexible adaptable process. The ideal is to be able to adapt quickly to both problems and opportunities. There is an inevitable trade-off between flexibility and control, more of one means less of the other. If a project (e.g. life-critical software) values control more than agility RAD is not appropriate.
* Poor design. The focus on prototypes can be taken too far in some cases resulting in a "hack and test" methodology where developers are constantly making minor changes to individual components and ignoring system architecture issues that could result in a better overall design. This can especially be an issue for methodologies such as Martin's that focus so heavily on the User Interface of the system.
* Very large systems. RAD typically focuses on small to medium-sized project teams. The other issues cited above (less design and control) present special challenges when using a RAD approach for very large scale systems.

The James Martin approach to RAD divides the process into four distinct phases:

1. **Requirements planning phase** – combines elements of the system planning and systems analysis phases of the Systems Development Life Cycle (SDLC). Users, managers, and IT staff members discuss and agree on business needs, project scope, constraints, and system requirements. It ends when the team agrees on the key issues and obtains management authorization to continue.

The main method used in gathering data for this system was online interview.

Personnel that were interviewed include:

* Assistant head of ICT department
* Some employees
* Network Administrator

**3.5.1 FUNCTIONAL REQUIREMENTS**

* System must capture faces
* System must store faces in a DB attached to the employee’s ID
* System must recognize the employee to enable attendance marking

**3.5.2 NON- FUNCTIONAL REQUIREMENTS**

* System shall be error-free
* System shall operate in real-time
* System should prevent data manipulation
* System should have a maximum uptime

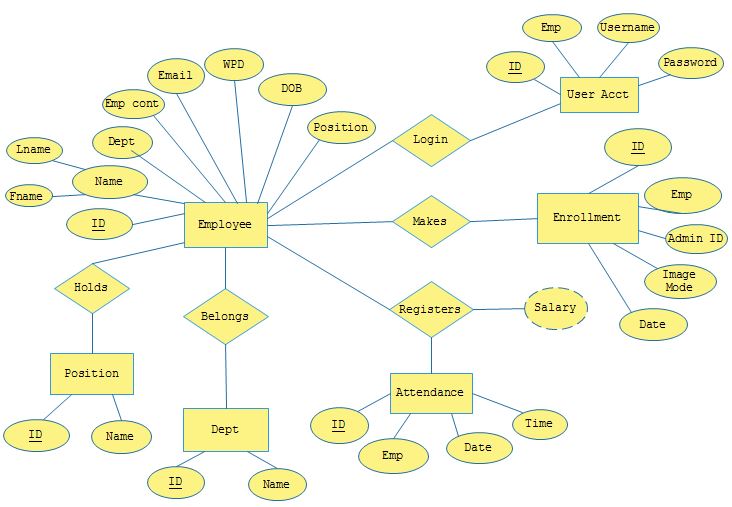
**3.5.3 HARDWARE REQUIREMENTS**

* Camera
* Computer System
* Screen

**3.5.4 SOFTWARE REQUIREMENTS**

* Windows OS 7 or later
* Minimum of 2GB RAM
* 40GB disk space
* Processor 1.8GHz or more\
  1. **USER DESIGN PHASE** – during this phase, users interact with systems analysts and develop models and prototypes that represent all system processes, inputs, and outputs. The RAD groups or subgroups typically use a combination of Joint Application Development (JAD) techniques and CASE tools to translate user needs into working models. *User Design* is a continuous interactive process that allows users to understand, modify, and eventually approve a working model of the system that meets their needs.

**3.6.1 ENTITY RELATION DIAGRAM**



**Fig 3.2** ENTITY RELATION DIAGRAM

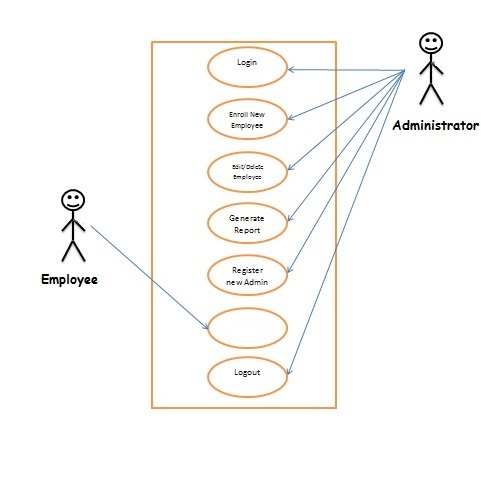
An entity-relationship diagram (ERD) is a type of data modeling that show a graphical representation of objects or concepts within an information system or organization and their relationship to one another. This entity relationship diagram describes an employee belonging to a department holding the admin position logging in to a user account. He can choose to make enrollment of a new employee. An employee registers attendance and his salary calculated based on attendance and Wage Per Day (WPD)

**3.6.2 USE CASE DIAGRAM**

The Use Case Model describes the proposed functionality of the new system. A Use Case represents a discrete unit of interaction between a user (human or machine) and the system. A Use Case is a single unit of meaningful work; for example login to system, register with system and create order are all Use Cases. Each Use Case has a description which describes the functionality that will be built in the proposed system. A Use Case may 'include'

another Use Case's functionality or 'extend' another Use Case with its own behavior.

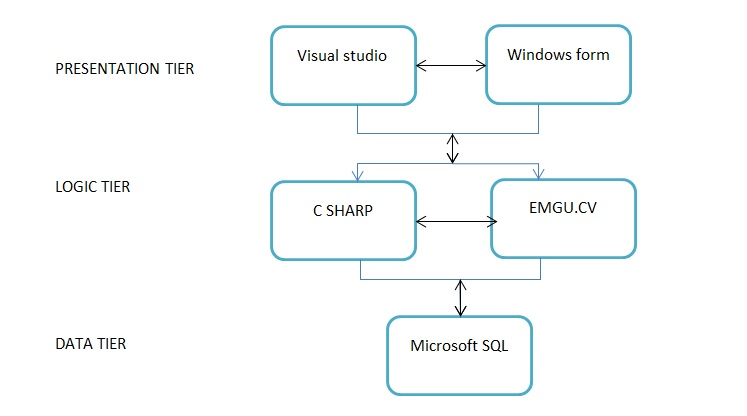
In the use case above, the administrator adds employees to the system by enrollment. The employee therefore login by the help of the camera to detect the face. The administrator is able to have access the checked attendance, recorded attendance generate report the he or she logs out



**Capture face for attendance**

**Fig 3.3** USE CASE DIAGRAM

**3.6.3 SYSTEM ARCHITECTURE**



**Server**

**Fig 3.4** SYSTEM ARCHITECTURE

System Architecture describes the various tiers of a system (Data, Logic and Presentation). The data tier describes the raw untouched unmodified data contained within the system which in our system was developed with Microsoft SQL Server.

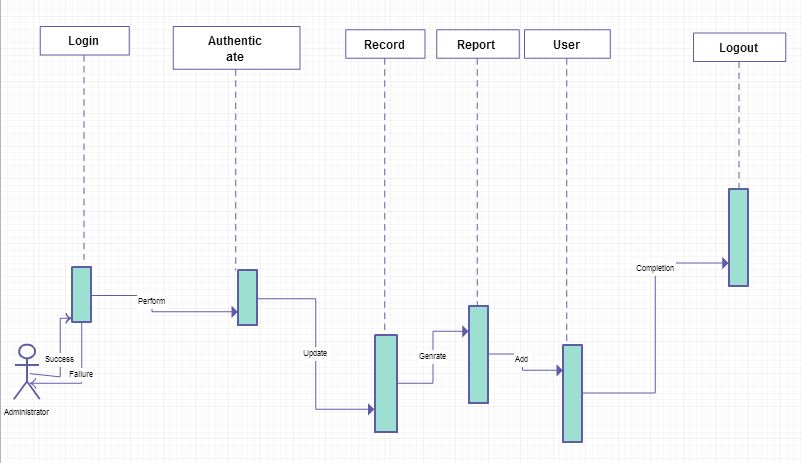
The Logic tier represents the use of the data to make the system work and feed the system with the necessary data needed to work. These include the attendance marking, payroll calculation and enrolling new employees. This was achieved using C# and an open source library Emgu.cv.

The presentation tier refers to the specific view a user wants to see. These include querying the database for reports and showing relationships between entities. Visual studio Windows Form Applications helps us to design these presentations.

**3.6.4.1 SEQUENCE DIAGRAM (ADMIN)**

A sequence diagram depicts a set of actions to be performed when a user uses a system.

Here, an administrator logs in, and is being authenticated. He can update records; generate reports on attendance and payroll. He can add a new user (Employee and Admin) and can choose to repeat all these processes. On completion, he logs out of the system.

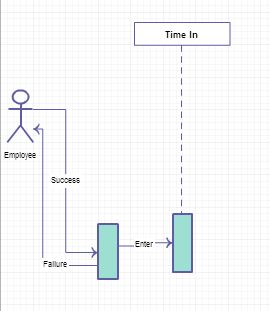


**Fig 3.5.1** SEQUENCE DIAGRAM (ADMIN)

**3.6.4.2 SEQUENCE DIAGRAM (EMPLOYEE)**

A sequence diagram depicts a set of actions to be performed when a user uses a system.

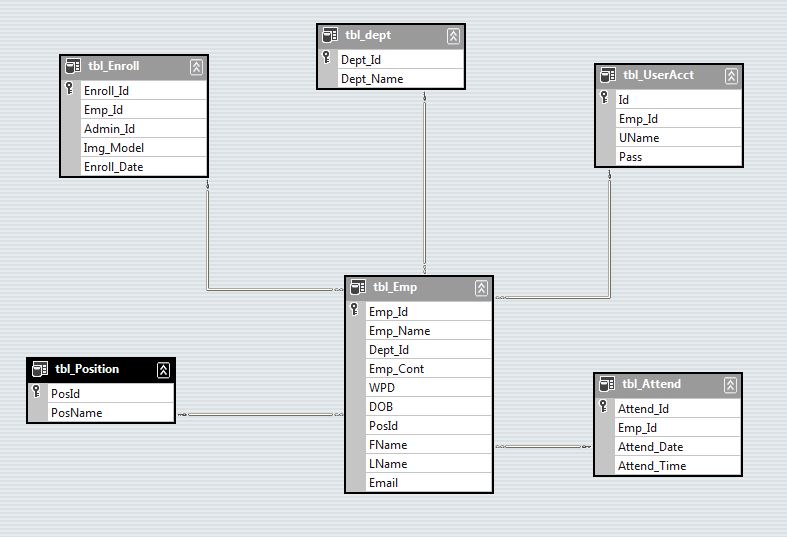
Here, an administrator logs in, and is being authenticated. He can update records; generate reports on attendance and payroll. He can add a new user (Employee and Admin) and can choose to repeat all these processes. On completion, he logs out of the system



**Fig 3.5.2** SEQUENCE DIAGRAM (EMPLOYEE)

≈

**3.6.5 DATABASE SCHEMA**



**Fig 3.6** DATABASE SCHEMA

**CHAPTER FOUR**

**SYSTEM IMPLEMENTATION AND TESTING**

**4.1 INTRODUCTION**

**4.1 CONSTRUCTION PHASE** (**RAD**)

Focuses on program and application development task similar to the SDLC. In RAD, however, users continue to participate and can still suggest changes or improvements as actual screens or reports are developed. Its tasks are programming and application development, coding, unit-integration and system testing.

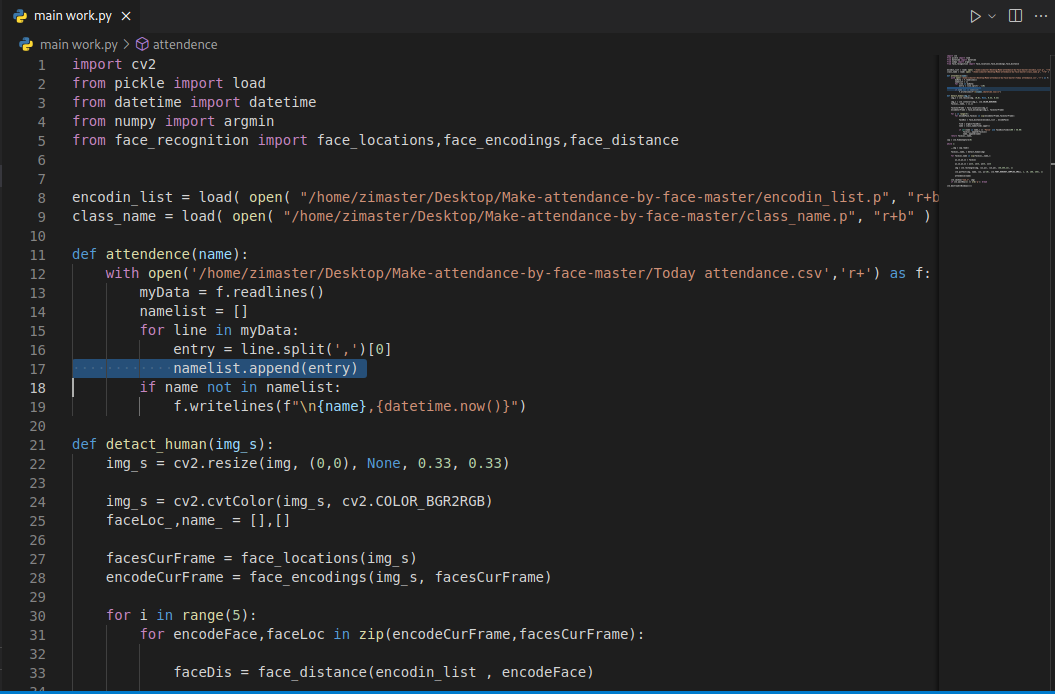
**4.2 IMPLEMENTATION STAGES**

The system is basically a windows form application running on any modern day operating system such as windows OS, Linux or Unix OS as well as Macintosh Operating system. It has two areas namely the administrator area which serves a platform for the administrator to use all functionalities of the system and a public screen which displays an incoming employee and the time and date. The backend includes the database which stores and supplies information to and from the system. Both front and backend of the system interact with each other in real-time basis.

**4.2.1 DEVELOPMENT TOOLS**

The development tools for this project include open source software and other software application. It will include the following:

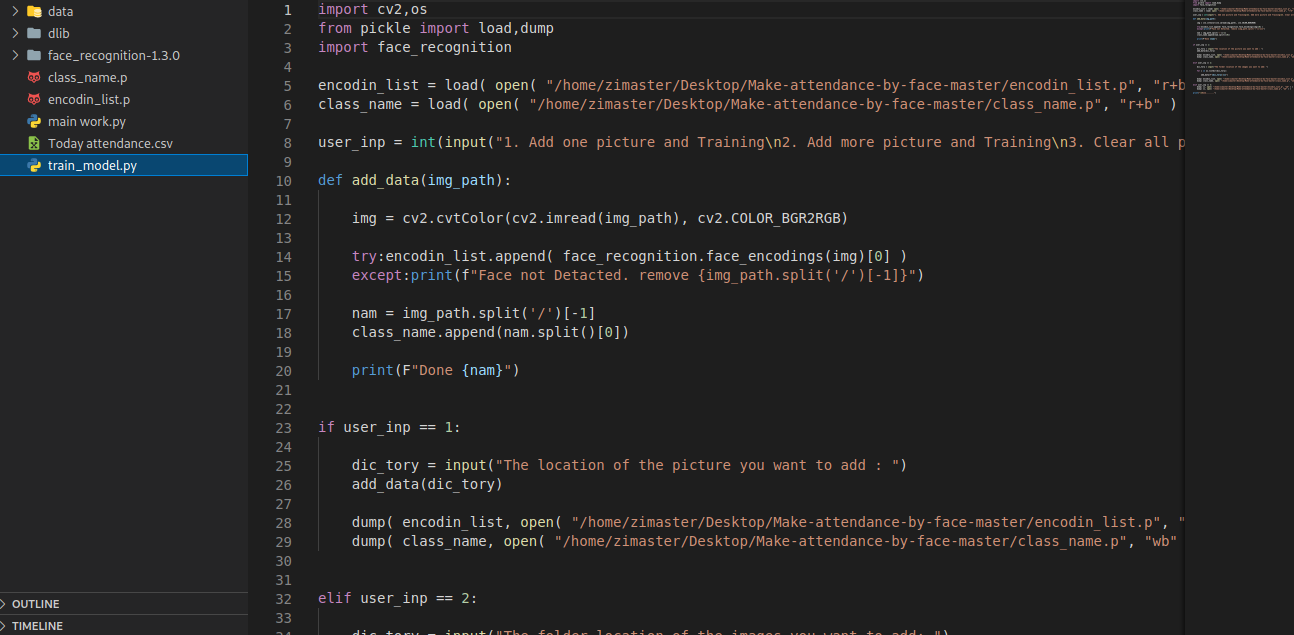
* **Visual Studio Code**: this is an IDE (Integrated Development Environment).
* **Python programming language**: this is the primary programming language used in development of the system
* **OpenCV:** OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products.
* **Tkinter Library:** Used in the designing of the interface
* **Spreadsheets:** Used to store the list of daily attendance
* **Face-recognition:** An open source Artificial intelligence library that is used to detected faces from a display source and mark them accordingly.
* **Date and Time library:** An open source libraries used within the python program to marked the date and time of the attendance

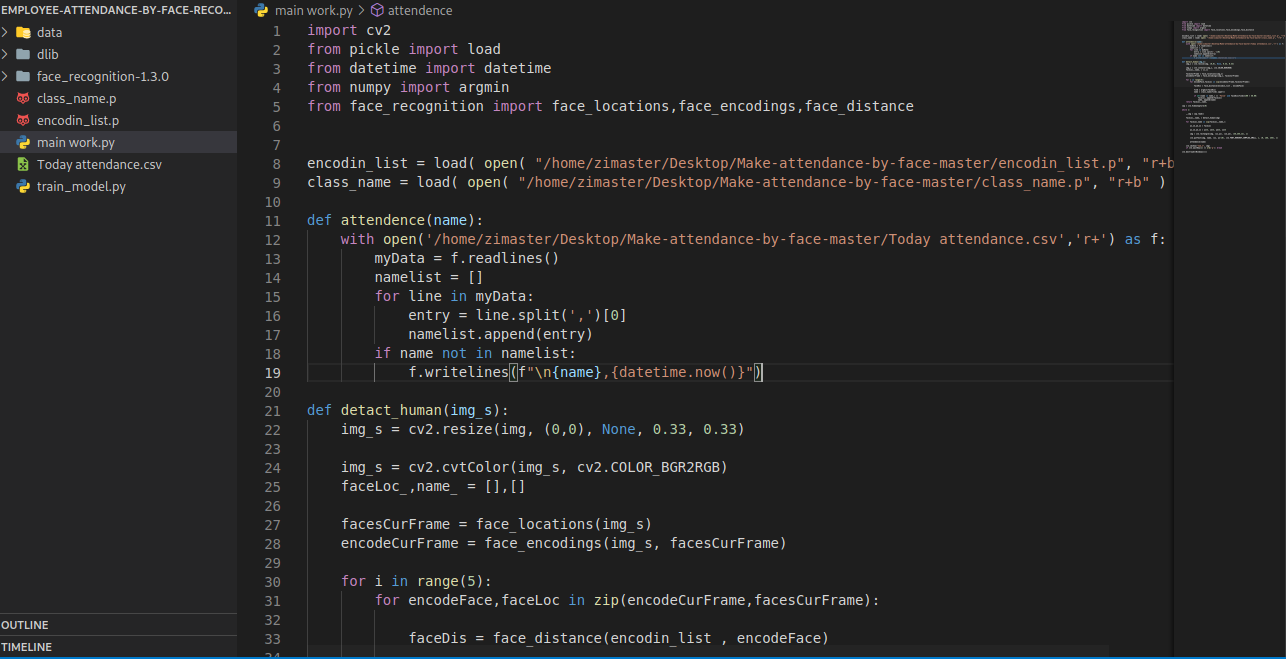
**FIG 4.1** Import libraries

**4.2.2 CREATION OF WINDOWS APPLICATION**

The creation of the windows application was mostly done in visual studio code. Resources such as Dlib, numpy cv2 and face recognition libraries were imported to assist in working with the images and the facial recognition and detection.

Microsoft SQL server was used in the creation of the database

**FIG 4.2** Train models python code-base

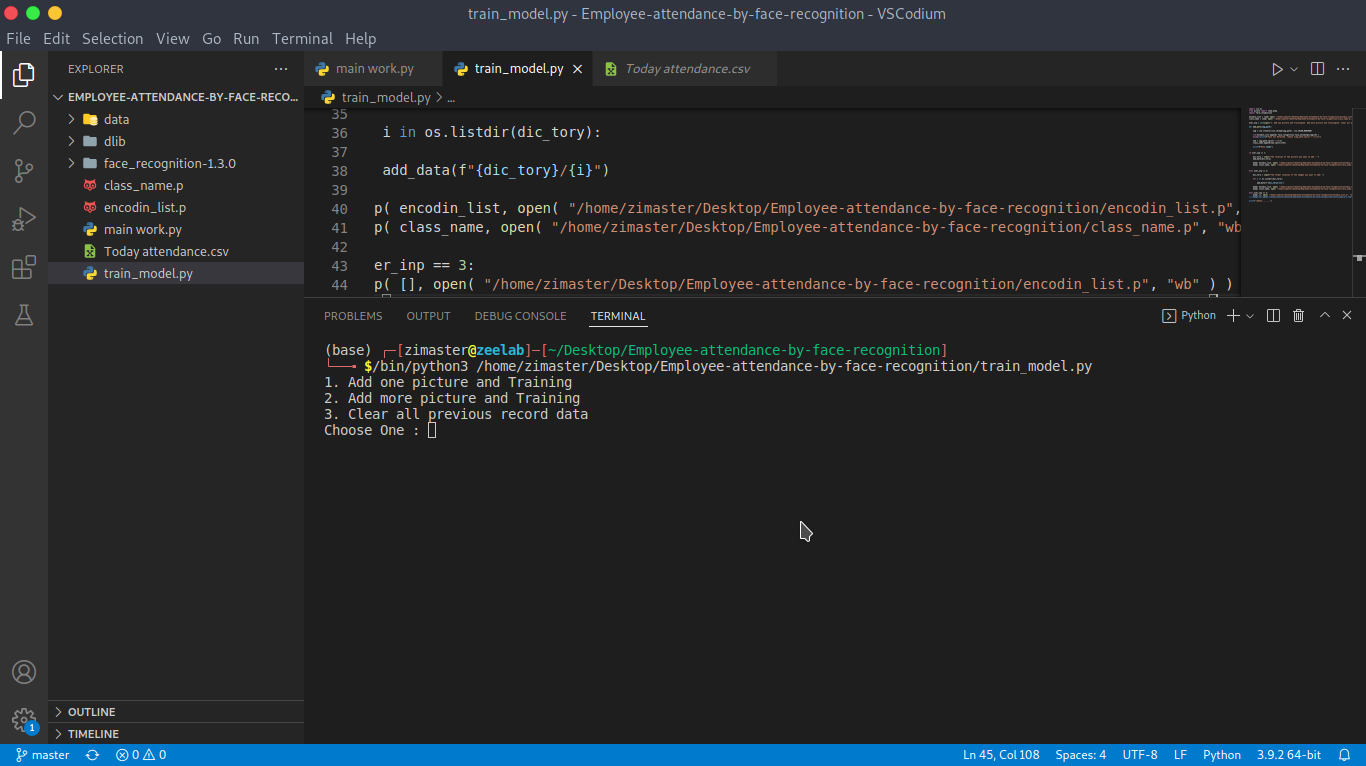
**FIG 4.3** Main work python code-base

**HOW IT WORKS**

**INTERFACES**

**4.3.1 TRAIN\_MODEL.PY**

The application has two major faces, The admin area for adding new images and the test phase which is capable of capturing a current face and compare it to a list of stored images. So when you execute this file, it allows the administrator to add a picture or pictures of employees to be recorded.

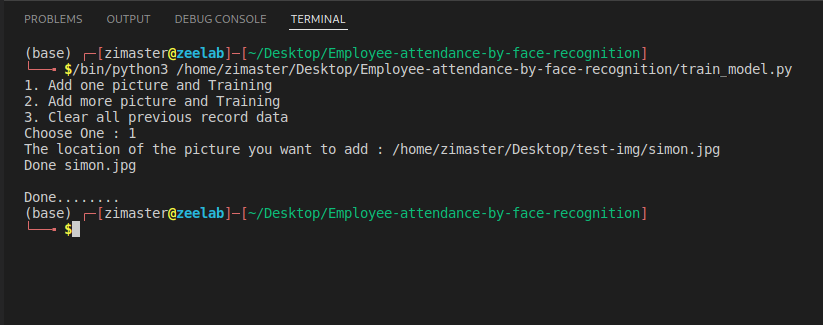


**FIG 4.4** ADD PICTURE (S)

The administrator is prompted to add the the employees picture location within the computer so that they are processed for comparison with newly coming pictures.

**4.3.2 ADMINISTRATOR PANEL**

At executing the train model file, the administrator panel is shown. Here the administrator has access to all system functionalities such as add users, enroll employee, edit or delete employee, print reports, check and mark attendance and many other functionalities captured in the system

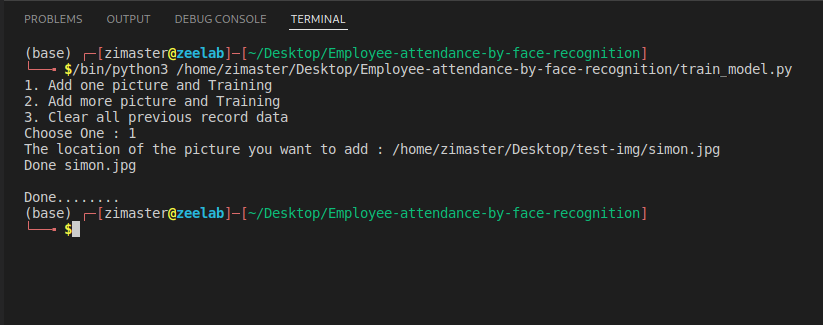


**FIG 4.2** ADMINISTRATOR AREA

**4.3.3 ENROLL EMPLOYEE**

An administrator can enroll a new employee supplying details such as NAME and IMAGE for recognition. The process is as follows:

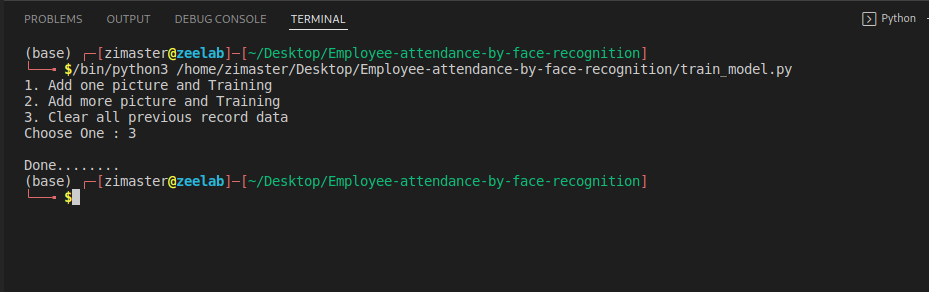
* execute **train\_model.py**
* in the choices select either choice (1) or (2). This is achieved by typing then number in the terminal window and pressing Enter as shown below.



**FIG 4.3** ENROLL EMPLOYEE

**4.3.4 EDIT EMPLOYEE**

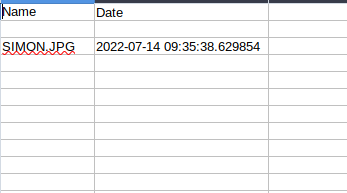
The administrator is able to edit and update information of the employee. This is achieved by choice number (3) which allows for deleting previous records.



**FIG4.4** EDIT EMPLOYEE

**4.3.5 ATTENDANCE REPORT**

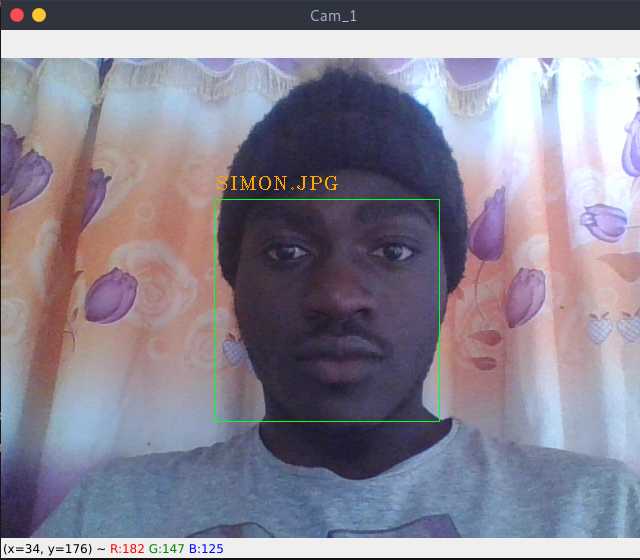
The administrator can generate a report on attendance based on date and time status of the employees

****

**FIG4.5** ATTENDANCE REPORT

**4.3.6 ATTENDANCE MARKING SCREEN**

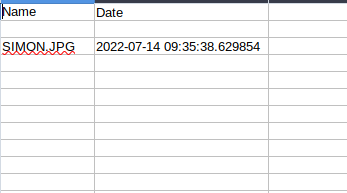
This screen is visible on a monitor close to the camera for employees to see if they have been captured and the time they came in. the form displays the date, time, employee detected and marks the attendance.



**FIG4.6** ATTENDANCE MARKING SCREEN

**4.3.7 PAYSLIP**

The administrator can generate a pay slip from the system. This captures the times employee was present, late, absent, employee’s specified WPD (Wage Per Day) and does the necessary computations to give the total salary for the month.



**FIG4.7** PAYSLIP

**4.4 CODING**

The project uses a simple interface and uses the necessary algorithm to help satisfy the objectives and also meet the requirements specified by stake holders. It comprises of a windows application and a database. Open source tools were used in its development and they are: face-recognition, numpy, Open computer vision and Dlib. These were used for both front end and back end development.

**4.5 TESTING**

**Integration and Testing:** All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.

**4.5.1 UNIT TESTING**

According to IEEE Standard, “Unit testing is the testing of individual hardware or software units or group of related units”. Thorough testing has gone into the project to ensure there are no lapses in the capturing of information. Various features such as: attendance marking, enrollment, adding and pay slip generation. This is done to ensure that each feature is performing correctly and working in real-time. Testing was also done to ensure that both functional and non-functional requirements are all met.

**4.5.2 INTEGRATION TESTING**

Integration testing is when all software components are combined and tested to check the level of interactivity between them. This testing determines that the applications involved are all functioning well with each other. This testing includes interface testing and usability testing.

We used the bottom-up approach where testing was done on lower levels before higher level components.

**4.6 CUTOVER PHASE (RAD)** – resembles the final tasks in the SDLC implementation phase, including data conversion, testing, changeover to the new system, and user training. Compared with traditional methods, the entire process is compressed. As a result, the new system is built, delivered, and placed in operation much sooner.

**4.7 DEPLOYMENT OF SYSTEM:** Once the functional and non-functional testing is done, the product is deployed in the customer environment or released into the market.

**4.8 MAINTENANCE:** There are some issues which come up in the client environment. To fix those issues patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

**CHAPTER FIVE**

**CONCLUSION AND RECOMMENDATION**

**5.1 SUMMARY**

Attendance is a key factor in employee salary processing for Niger Insurance PLC as well as other institutions. In this project, the potential benefits of implementing e-attendance environment were investigated. This is done in order to eliminate the challenges and limitations of the current e-attendance system. This project is centered on how to enhance attendance marking and generate pay slips from it.

First and foremost, the state-of-the-art report of e-attendance was studied. Furthermore, the essential benefit of e-attendance was highlighted. We went further by investigating how face recognition can solve the disturbing challenges manual attendance systems. Based on the knowledge derived from this project, it was concluded that implementing an e-attendance system in Niger Insurance PLC will not only eliminate the challenges that are faced by Niger Insurance PLC but also provide a rich, effective, and more flexible environment that will have a positive effect on attendance.

**5.2 CONCLUSION**

Automated attendance systems are more efficient than manual systems as it prevents employees from falsifying entries.

### 5.3 RECOMMENDATION

For the purpose of ICT growth that will lead to a rich, efficient, and result driven mode of attendance in major organizations, we recommend that the face recognition algorithm should be used in order to enhance the e-attendance system.

**5.4 PERSONAL EXPERIENCE**

The project really helped to sharpen my programming skills and understanding of the power of Artificial Intelligence. It served as a tough challenge as writing algorithms for face detection and recognition was a huge step from our previous knowledge. It also gave an insight to our short comings and abilities.

### 5.5 FUTURE RESEARCH DIRECTIONS

The project is just a blueprint for implementing a full functional e-attendance system. After this blueprint, a thorough research should be carried out on this system, laying more emphasis on the impact it has on attendance and how it has enhanced company’s employee-hour-output. The impact that this system have on the business sector should be studied and compared to that of the manual system

We successfully designed and implemented a e-attendance for an organization. The entire system should be studied properly so as to detect the faults of the system and ways of improving it. Research should also be carried out on how to integrate this system into a fully functional employee management system and payroll.

**5.6 FUTURE WORK**

Employee Management Systems should be implemented

A leave management system should also be implemented.

**REFERENCES**

* <http://www.projecttopics.info/Computer/Face_Recognition_in_e-attendance.php> [Accessed: June 2022]
* The Code Project, “EMGU Multiple Face Recognition using PCA and Parallel Optimisation", 05 October 2011. Available http://www.codeproject.com/Articles /261550/EMGU-Multiple-FaceRecognition-using-PCA-and-Paral [Accessed: July 2022]
* <https://developers.keylemon.com/welcome> [Accessed: June 2022]...
* OpenCV API reference. Available: <http://docs.opencv.org/modules/refman.html> [Accessed: July 2022]
* <http://www.fbi.gov/about-us/cjis/fingerprints_biometrics/biometric-center-of-excellence/modalities/facial-recognition>
* Computer vision based employee.pdf
* Available: <https://www.google.com.ng/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&cad=rja&uact=8&ved=0CCkQFjAB&url=http%3A%2F%2Fdspace.bracu.ac.bd%2Fbitstream%2Fhandle%2F10361%2F2747%2FComputer%2520vision%2520besed%2520employee.pdf%3Fsequence%3D1&ei=CYW-U5asMqzb7Aawv4HICQ&usg=AFQjCNEiGruMJtA9eFwlARO-yMtMjgTG2w&sig2=e3u7_DZufmm4NsYPe9JlMw>
* H. K. Ekenel, J. Stallkamp, H. Gao, M. Fischer, R. Stiefelhagen, “FACE RECOGNITION FOR SMART INTERACATINONS”, interACT Research, Computer Science Department, Universit¨atKarlsruhe (TH).
* Phillip Ian Wilson, and Dr. John Fernandez, “FACIAL FEATURE DETECTION USING HAAR CLASSIFIERS”, Texas A&M University.
* Viola, P. and M. Jones (2001). Rapid Object Detection Using a Boosted Cascade of Simple Features, in: Proc. IEEE Conf.
* Computer Vision and Pattern Recognition.
* JA.C. Weaver, “Biometric authentication”, Computer, 39(2), pp. 96-97 (2006).
* K. Kendall, J. Kendall (2011). system analysis and design 8th edition, Pearson Education, Inc.,
* "Time Attendance Device Aflak Electronics".[ http://aflak.com.sa/newweb/html/eng/timeAttendanceSoftware.html] Retrieved 8 December 2014.
* Jump up Protefil Group case study, “Touch screen terminals used to record staff attendance[http://www.sescoi.com/casestudies/pdf/wplan-protecmo.pdf] Retrieved 8 December 2014.
* <http://www.healthyworkinglives.com/advice/Legislation-and-policy/work-related-illness-injury/attendance-management/2definition> [Accessed: June 2022]
* <http://www.mitrefinch.co.uk/products/time_and_attendance/>[Accessed: June 2022]
* <http://humanresources.about.com/od/attendancepolicy/> [Accessed: June 2022]
* <http://www.acas.org.uk/media/pdf/n/m/G03_1.pdf> [Accessed: June 2022]
* <http://news.bbc.co.uk/1/hi/talking_point/4481317.stm> [Accessed: June 2022]
* <http://www.celayix.com/employee-time-and-attendance/monitoring/> [Accessed: June 2022]
* <http://www.acrotime.com/> [Accessed: June 2022]
* <http://www.adp.com/> [Accessed: July 2022]
* <http://www.apspayroll.com/> [Accessed: June 2022]
* <http://acumendatasystems.com/> [Accessed: June 2022]
* <http://easyclocking.com/> [Accessed: June 2022]
* <http://www.workforcesoftware.com/> [Accessed: June 2022]
* [http://www.acumendatasystems.com](http://www.acumendatasystems.com/) [Accessed: June 2022]
* [http://www.kronos.com](http://www.kronos.com/) [Accessed: June 2022]
* [http://www.replicon.com](http://www.replicon.com/) [Accessed: June 2022]
* [http://www.timeclockplus.com](http://www.timeclockplus.com/) [Accessed: June 2022]
* [http://www.mytimeforce.com](http://www.mytimeforce.com/) [Accessed: June 2022]
* [http://www.tsheets.com](http://www.tsheets.com/) [Accessed: June 2022]
* [http://www.uattend.com](http://www.uattend.com/) [Accessed: June 2022]
* <http://www.businessnewsdaily.com/6730-best-time-and-attendance-systems.html> [Accessed : June 2022]

**APPENDICES**

**REQUIRED LIBRARIES AND IMPORTS**

###

*Python 3.8.5*

*opencv-python==4.5.2.52*

*numpy==1.18.5*

*face\_recognition==1.2.3*

###

**IMPORTING THE LIBRARIES IN THE SCRIPT**

###

# main.py file

import cv2

from pickle import load

from datetime import datetime

from numpy import argmin

from face\_recognition import face\_locations,face\_encodings,face\_distance

# train\_model.py

import cv2,os

from pickle import load,dump

import face\_recognition

.....

**INITIALIZING CAMERA FOR FACE CAPTURE**

def detact\_human(img\_s):

img\_s = cv2.resize(img, (0,0), None, 0.33, 0.33)

img\_s = cv2.cvtColor(img\_s, cv2.COLOR\_BGR2RGB)

faceLoc\_,name\_ = [],[]

facesCurFrame = face\_locations(img\_s)

encodeCurFrame = face\_encodings(img\_s, facesCurFrame)

for i in range(5):

for encodeFace,faceLoc in zip(encodeCurFrame,facesCurFrame):

faceDis = face\_distance(encodin\_list , encodeFace)

find = argmin(faceDis)

name = class\_name[find].upper()

if str(name in name\_) == 'False' and faceDis[find]\*100 < 48.00:

faceLoc\_.append(faceLoc)

name\_.append(name)

return faceLoc\_,name\_

cap = cv2.VideoCapture(0)

........

**ATTENDANCE FORM**

def attendence(name):

with open('/home/zimaster/Desktop/Employee-attendance-by-face-recognition/Today attendance.csv','r+') as f:

myData = f.readlines()

namelist = []

for line in myData:

entry = line.split(',')[0]

namelist.append(entry)

if name not in namelist:

f.writelines(f"\n{name},{datetime.now()}")

……..

**ADD AN EMPLOYEE**

def add\_data(img\_path):

img = cv2.cvtColor(cv2.imread(img\_path), cv2.COLOR\_BGR2RGB)

try:encodin\_list.append( face\_recognition.face\_encodings(img)[0] )

except:print(f"Face not Detacted. remove {img\_path.split('/')[-1]}")

nam = img\_path.split('/')[-1]

class\_name.append(nam.split()[0])

print(F"Done {nam}")

if user\_inp == 1:

dic\_tory = input("The location of the picture you want to add : ")

add\_data(dic\_tory)

dump( encodin\_list, open( "/home/zimaster/Desktop/Employee-attendance-by-face-recognition/encodin\_list.p", "wb" ) )

dump( class\_name, open( "/home/zimaster/Desktop/Employee-attendance-by-face-recognition/class\_name.p", "wb" ) )

elif user\_inp == 2:

dic\_tory = input("The folder location of the images you want to add: ")

for i in os.listdir(dic\_tory):

add\_data(f"{dic\_tory}/{i}")

dump( encodin\_list, open( "/home/zimaster/Desktop/Employee-attendance-by-face-recognition/encodin\_list.p", "wb" ) )

dump( class\_name, open( "/home/zimaster/Desktop/Employee-attendance-by-face-recognition/class\_name.p", "wb" ) )

elif user\_inp == 3:

dump( [], open( "/home/zimaster/Desktop/Employee-attendance-by-face-recognition/encodin\_list.p", "wb" ) )

dump( [], open( "/home/zimaster/Desktop/Employee-attendance-by-face-recognition/class\_name.p", "wb" ) )

print("\nDone........")

……

**FRAMEGRABBER ACTION**

def detact\_human(img\_s):

img\_s = cv2.resize(img, (0,0), None, 0.33, 0.33)

img\_s = cv2.cvtColor(img\_s, cv2.COLOR\_BGR2RGB)

faceLoc\_,name\_ = [],[]

facesCurFrame = face\_locations(img\_s)

encodeCurFrame = face\_encodings(img\_s, facesCurFrame)

for i in range(5):

for encodeFace,faceLoc in zip(encodeCurFrame,facesCurFrame):

faceDis = face\_distance(encodin\_list , encodeFace)

find = argmin(faceDis)

name = class\_name[find].upper()

if str(name in name\_) == 'False' and faceDis[find]\*100 < 48.00:

faceLoc\_.append(faceLoc)

name\_.append(name)

return faceLoc\_,name\_

cap = cv2.VideoCapture(0)

while 1:

\_,img = cap.read()

faceLoc\_,name\_ = detact\_human(img)

for faceLoc,name in zip(faceLoc\_,name\_):

y1,x2,y2,x1 = faceLoc

y1,x2,y2,x1 = y1\*3, x2\*3, y2\*3, x1\*3

img = cv2.rectangle(img, (x1,y1), (x2,y2), (36,255,12), 1)

cv2.putText(img, name, (x1, y1-10), cv2.FONT\_HERSHEY\_COMPLEX\_SMALL, 1, (0, 166, 255), 1)

attendence(name)

cv2.imshow("Cam\_1", img)

if cv2.waitKey(1) == ord('q'): break

cv2.destroyAllWindows()()

……

**REMOVING EMPLOYEE**

encodin\_list = load( open( "/home/zimaster/Desktop/Employee-attendance-by-face-recognition/encodin\_list.p", "r+b" ) )

class\_name = load( open( "/home/zimaster/Desktop/Employee-attendance-by-face-recognition/class\_name.p", "r+b" ) )

def attendence(name):

with open('/home/zimaster/Desktop/Employee-attendance-by-face-recognition/Today attendance.csv','r+') as f:

myData = f.readlines()

namelist = []

for line in myData:

entry = line.split(',')[0]

namelist.append(entry)

if name not in namelist:

f.writelines(f"\n{name},{datetime.now()}")

def detact\_human(img\_s):

img\_s = cv2.resize(img, (0,0), None, 0.33, 0.33)

img\_s = cv2.cvtColor(img\_s, cv2.COLOR\_BGR2RGB)

faceLoc\_,name\_ = [],[]

facesCurFrame = face\_locations(img\_s)

encodeCurFrame = face\_encodings(img\_s, facesCurFrame)

for i in range(5):

for encodeFace,faceLoc in zip(encodeCurFrame,facesCurFrame):

faceDis = face\_distance(encodin\_list , encodeFace)

find = argmin(faceDis)

name = class\_name[find].upper()

if str(name in name\_) == 'False' and faceDis[find]\*100 < 48.00:

faceLoc\_.append(faceLoc)

name\_.append(name)

return faceLoc\_,name\_

cap = cv2.VideoCapture(0)

while 1:

\_,img = cap.read()

faceLoc\_,name\_ = detact\_human(img)

for faceLoc,name in zip(faceLoc\_,name\_):

y1,x2,y2,x1 = faceLoc

y1,x2,y2,x1 = y1\*3, x2\*3, y2\*3, x1\*3

img = cv2.rectangle(img, (x1,y1), (x2,y2), (36,255,12), 1)

cv2.putText(img, name, (x1, y1-10), cv2.FONT\_HERSHEY\_COMPLEX\_SMALL, 1, (0, 166, 255), 1)

attendence(name)

cv2.imshow("Cam\_1", img)

if cv2.waitKey(1) == ord('q'): break

cv2.destroyAllWindows()()