Babasaheb Bhimrao Ambedkar Bihar University, Muzaffarpur, Bihar

A PROJECT REPORT ON

Attendance Management System Using Face Recognition

By

Aditya Aman

Registration No. : 19BRVVCS047/19

Roll No. : 2020847

Bachelor of Computer Application (B.C.A)

Semester – VI

Subject: BCA-601

Guided By:

Prof. Chandan Kumar Jha

Department of BCA

Laxmi Narayan College, Bhagwanpur (Vaishali)

ACKNOWLEDGEMENT

It is not until you undertake the project like this one that you realize how massive the effort it really is, or how much you must rely upon the Selfless efforts and goodwill of others. There are many who helped us with this project, and we want to thank them all from the core of our Hearts.

We owe special words of thanks to our Signature of Project Supervisor/Guide ***Mr. Chandan Kr. Jha*** *for their vision, thoughtful counseling and encouragement at every step of the project. We would also like thank* ***Mr. Mukesh Kr. Singh*** *and* ***Mrs. Aarti Singh*** *for their co-operation in the completion of the project.*

We are also thankful to our Project Team of the Development Department for giving us the best of knowledge and guidance throughout the project.

And last but not the least, we find no words to acknowledge the financial assistance & moral support rendered by our parents in making the effort a success.

All this has become reality because of their blessings and above all by the grace of God.

ABSTRACT

An organized and systematic office solution is essential for all universities and organizations. There are many departments of administration for the maintenance of information and databases in any organizations. All these departments provide various records regarding employees. Most of these track records need to maintain information about the employees. This information could be the general details like employee name, employee e-mail, performance, attendance etc. or specific information related to organizations like collection of data. For example, when an employee joins any organization needs to register his/her self-identity to generate his/her employee id for identity.

With that in mind, we overhauled the Attendance Management System Using Face Recognition and made necessary improvement to streamline the processes. This project involves building an attendance system which utilizes facial recognition to mark the presence, time-in and time-out of the employees. It covers areas such as facial detection, alignment and recognition along with the development of a web application to cater to various use cases of the system such as registration of new employees, addition of photos to the training dataset, viewing attendance reports, etc. This project intends to serve as an efficient substitute for traditional manual attendance systems. It can be used in corporate offices, schools and organizations where security is essential. In general, this project aims to enhance efficiency and at the same time maintain information accurateness. Later in this report, features and improvement that allow achievement to this goal will be demonstrated and highlighted.

Our work is useful for easy user interface. We are planning to utilize the powerful database management, data retrieval and data manipulation. We will provide more ease for managing the data than manually maintaining in the documents. Our work is useful for saving valuable time and reduces the huge paper work.

PREFACE

This project ***"Attendance Management System Using Face Recognition"*** *provides us a simple interface for maintenance of employee's attendance. It can be used by corporate offices, educational institutes or colleges to maintain the attendance of employees, students easily. The purpose of this document is to specify software requirements of the Attendance Management System Using Face Recognition. It is intended to be a complete specification of what functionality the Attendance Management System provides.*

Throughout the project the aims are to automate the traditional attendance system where the attendance is marked manually. It also enables an organization to maintain its records like in-time, out-time, break time and attendance digitally. Digitalization of the system would also help in better visualization of the data using graphs to display number of employees present today, total work hours of each employee and their break time. Its added features serve as an efficient upgrade and replacement over the traditional attendance system.

SCOPE

Facial recognition is becoming more prominent in our society. It has made major progress in the field of security. It is a very effective tool that can help low enforces to recognize criminals and software companies are leveraging the technology to help users access the technology. This technology can be further developed to be used in other avenues such as ATMs, accessing confidential files, or other sensitive materials.

This project serves as a foundation for future projects based on facial detection and recognition. This project also convers web development and database management with a user-friendly UI. Using this system any corporate offices, school and organization can replace their traditional way of maintaining attendance of the employees and can generate their availability(presence) report throughout the month.

DECLARTATION

I hereby declare that the above provided information is true and complete to the best of my knowledge and belief.

SINGNATURE

(Chandan Kumar Jha)

CERTIFICATE OF GUIDE

"This is to certify that the project ***Attendance Management System Using Face Recognition*** for wagons is a bonafide work done by Mr. Aditya Aman (BCA, Babasaheb Bhimrao Ambedkar Bihar University, Muzaffarpur, Bihar) in partial fulfillment of BCA Semester 6th project and has been carried out under my direct supervision and guidance.

SINGNATURE
(Chandan Kumar Jha)

CERTIFICATE OF ORIGINILITY

This is to certify that the project titled ***ATTENDANCE MANAGEMENT SYSTEM USING FACE RECOGINITION*** submitted to the award of degree of Bachelor of Computer Application in partial fulfillment of the requirement for Bachelor of Computer Application (BCA) degree is an original work carried out by **Mr. Aditya Aman** Reg. No. **19BRVVCS047/19** under the guidance of **Mr. Chandan Kumar Jha.**

The matter embodied in this project is a genuine work done by the student and has been submitted whether to this university or to any other university or institute for the fulfillment of the requirement of any course of study.

---------- ----------

SIGNATURE OF GUIDE SIGNATURE OF STUDENT

(Chandan Kumar Jha) (Aditya Aman)

Table of Contents

1. INTRODUCTION

Attendance Management System Using Face Recognition aims to automate the traditional attendance system where the attendance is marked manually. It also enables an organization to maintain its records like in-time, out-time, break time and attendance digitally. Digitalization of the system would also help in better visualization of the data using graphs to display the number of employees present today, total work hours of each employee and their break time. Its added features serve as an efficient upgrade and replacement over the traditional attendance system.

Our design can facilitate us to explore all the activities happening in the organization, even we can get to know which employee is present on the specific day, total number of employees present on that day, attendance percentage of an employee. It also facilitates the work hours of an employee. The attendance management system using face recognition is an automated version of manual Attendance Management System.

In case of manual system, they need a lot of time, manpower etc. Here almost all work is computerized. So, the accuracy is maintained. Maintaining backup is very easy. It can do within a few minutes. Our system has two types of accessing modes, administrator and user. Attendance Management System Using Face Recognition is managed by an administrator. It is the job of the administrator to insert update and monitor the whole process. When a user logs in to the system. He/she would only view details of the personal panel. He/she can't perform any changes.

2. PROJECT PREVIEW

Project Management

Project management skills are put to good use for this project. Having gone through project management modules in Time Series Analysis, Optimization and with two interns Project Management for Business and IT respectively, they enhanced my knowledge on managing a project. Project management focuses on achieving the objectives by applying five processes presented in Figure below.

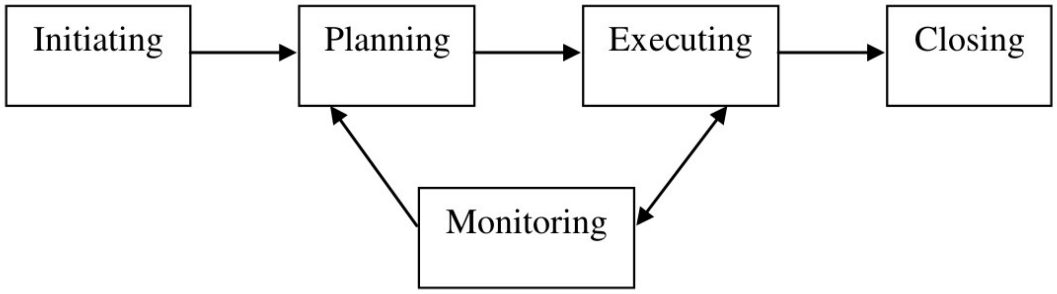


Figure 2.1: Project Development Phases

There are many software developments companies that offer attendance management system using face recognition for the organization and educational institutes in the market. There are records on the past years projects on attendance management system using face recognition is done by students. Through the researches, it is observed that there are features where this project can adopt and implement. It can maintain all the details of employee's records like in-time, out-time, break time and attendance digitally.

3. System Development Life Cycle

Systems Development Life Cycle (SDLC) is the most common process adopted to develop a project and not surprisingly, this project is following this model too. To be precise, waterfall model is being applied. Waterfall model is a sequential model process where the input of a phase actually results from the previous phase.

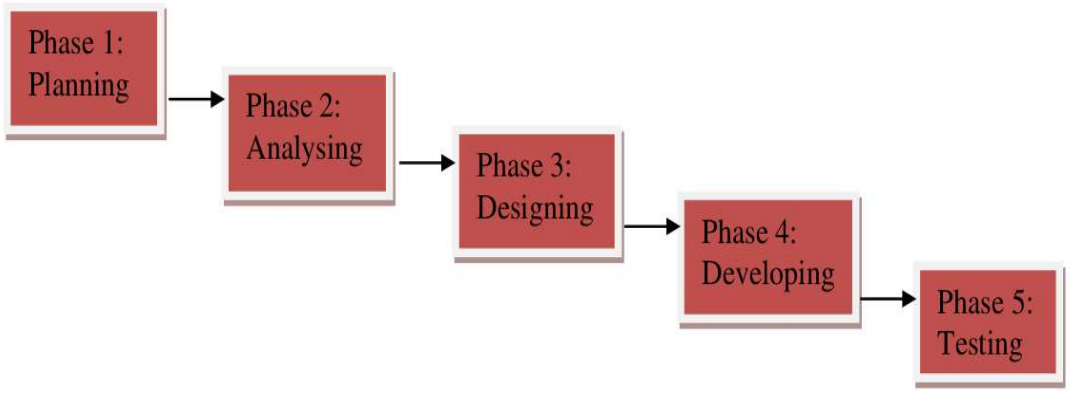


Figure 3.1: SDLC Phases

There are five phases in this model and the first phase is the planning stage. The planning stage determines the objectives of the project and whether the project should be given the green light to proceed. This is where the proposal submission comes into picture. After obtaining the approval, the next phase is analysis. Gathering and analyzing the system and user requirements is essential for entry to the design step.

With the user requirements gathering completed, there is a need to prepare the resources for the project. Be it software or hardware components, careful consideration and selection is to be taken care at this stage. The decision on the appropriate resources to be used is further elaborated under the subsections below. The next step is to design the system and database structure.

Results from the analysis and preparation that were concluded from the previous stage are put into action. With the user requirements in mind, the flow of the system is planned and the user interface is designed to suit their easy navigation needs. In addition, the number of tables, attributes, primary and unique keys of the database is listed.

After completing the design, actual coding begins. Database is created and codes are written. Some of the codes required amendments and improvement to it so these are being developed at this fourth stage of the waterfall model. With the development completed,

testing will begin. The codes and database are tested to ensure the results obtained are as intended. More time is spent on both development and testing stages because it is inevitable to have errors and issues and buffer time is allocated for troubleshooting.

4. Problem Analysis

Problem Analysis is a management technique, which helps in designing a new system or improving an existing system. System Analysis is the process of gathering and interpreting facts, diagnosing problems (if any), using information to recommend improvements to the system. There are four basic elements of system analysis: - Output, Input, Files Processes, and Database Management. For computerization of any system, the existing system must be thoroughly being understood to determine "how the computer can be best used to make its operation most effective". This is acquired by analyzing existing system.

1. FEASIBILITY STUDY

The feasibility of designing the system determined by evaluating alternative methods of converting available input data into require output. Each of these alternative methods that is termed as candidate system. The purpose of feasibility study is to identify various candidate systems and to evaluate system.

Feasibility study is the process of determination of whether or not a project is worth doing. Feasibility studies are undertaken within tight time constraints and normally culminate in a written and oral feasibility report. I have taken two weeks in feasibility study with my co-developer. The contents and recommendations of this feasibility study helped us as a sound basis for deciding how to precede the project. It helped in taking decisions such as which software to use, hardware combinations, etc.

- Technical Feasibility
- Economic Feasibility
- Operational Feasibility

A. TECHNICAL FEASIBILITY:

Evaluating the technical feasibility is the trickiest part of a feasibility study. This is because, at this point of time, not too many detailed design of the system is available, making it difficult to access issues like performance, costs on (on account of the kind of technology to be deployed) etc. A number of issues have to be considered while doing a technical analysis.

Understand the different technologies involved in the proposed system:

Before commencing the project, we have to be very clear about what are the technologies that are to be required for the development of the new system.

Find out whether the organization currently possesses the required technologies:

Is the required technology available with the organization?

If so is the capacity sufficient?
For instance: "Will the current printer be able to handle the new reports and forms required for the new system?"

B. OPERATIONAL FEASIBILITY:

Proposed projects are beneficial only if they can be turned into information systems that will meet the organizations operating requirements. Simply stated, this test of feasibility asks if the system will work when it is developed and installed. Are there major barriers to Implementation? Here are questions that will help test the operational feasibility of a project:

- Is there sufficient support for the project from management from users? If the current system is well liked and used to the extent that persons will not be able to see reasons for change, there may be resistance.
- Are the current business methods acceptable to the user? If they are not, Users may welcome a change that will bring about a more operational and useful systems.
- Have the user been involved in the planning and development of the project?
- Early involvement reduces the chances of resistance to the system and in
- General and increases the likelihood of successful project.

Since the proposed system was to help reduce the hardships encountered. In the existing manual system, the new system was considered to be operational feasible.

In examining technical feasibility, configuration of the system is given more importance than the actual makes of hardware. The configuration should give the complete picture about the system's requirement:
There are a number of technical issues which are generally raised during the feasibility stage of the investigation. They are as follows:

Q. Does the necessary technology exist to do what is suggested (and can it be acquired)?
Q. Does the proposed equipment have the technical system?
Q. Can the system be upgraded if developed?
Q. Are there technical guarantees of accuracy, reliability, ease of access and data security?
Q. Does the necessary technology exist to do what is suggested and can it?
Q. How many workstations are required?

Q. How these units are interconnected so that they could operate and communicate smoothly?

Q. What speeds of inputs and output should be achieved at particular quality of printing

Specific hardware and software products can then be evaluated keeping in viewing with the logical needs.

C. ECONOMIC FEASIBILITY:

Economic feasibility attempts 2 weigh the costs of developing and implementing a new system, against the benefits that would accrue from having the new system in place. This feasibility study gives the top management the economic justification for the new system.

A simple economic analysis which gives the actual comparison of costs and benefits are much more meaningful in this case. In addition, this proves to be a useful point of reference to compare actual costs as the project progresses. There could be various types of intangible benefits on account of automation. These could include increased customer satisfaction, improvement in product quality better decision making timeliness of information, expediting activities, improved accuracy of operations, better documentation and record keeping, faster retrieval of information, better employee morale.

It is very most important. Because software is the most expensive element for virtually all computer-based system. A large cost estimation error can make the difference between profit and loss. cost overrun can be desirous for the developer. Here I used economic analysis to study economic feasibility. Economic analysis is the most frequently used technique for evaluating the effectiveness of a proposed system. More commonly known as **cost/benefit** analysis: the procedure is to determine the benefits and saving that are expected from a proposed system and compare them with cost. If benefits outweigh costs, a decision is taken to design and implement the system. Otherwise, further justification or alternative in the proposed system will have to be made if it is to have chance of begin approved. This is an ongoing effort that improves in accuracy at each phase of the system life cycle.

A system that can be developed technically and that will be used if installed must still be profitable for the organization. Financial benefit must equal or exceed the costs. The analysts arise various financial and economic questions that during the primary investigation to estimate the following:

- The cost to conduct a full systems investigation.

- The cost of hardware and software for the class of application being considered.
- The benefit in the form of reduced costs or fewer costly errors
- The cost if nothing changes

Cost/Benefit Analysis:

To carry out an economic feasibility study, it is necessary to place actual money values against any purchases or activities needs to implement this project. it is also necessary to place money values against any benefits that will accurate from a new system caused by the place money values against any benefits that will accurate from a new system created by the project. Such calculations are often described as cost-benefit analysis.

Cost–benefit analysis usually includes two steps:

1. Producing the estimates of cost and benefits,

2. And determining whether the project is worthwhile once these costs are ascertained.

Producing Costs and Benefits:

The goal is to reduce a list of what is required to implement the system and a list of the new system benefits.

Cost–benefit analysis is always clouded by both tangible and intangible items. Tangible items are those to which direct values can be attached (e.g., the purchase of equipment, time spent by people writing programs or items such as tires costs or the vat). Some tangible costs often associated with computer system development are: -

- **Equipment costs for the new systems:** various items of computing equipment, as well as items such accommodation costs and furniture, are included here
- **Personal costs:** These include personnel needed to develop the new system and those will subsequently run the system when it is established. Analysis, designers and programmers will need to build the system. Also include the personnel to train the system users.
- **Material costs:** These include stationary, manual production and other documentation costs.
- **Conversion costs:** The costs of designing new forms and producers and of the possible parallel running of the existing and new systems are included here.
- **Training costs:** These include the cost of training users of the new systems, as well as developers who may be required to use new technologies.

- **Other costs:** Sometimes consultant's costs are included here together with the management overheated, secretarial support, travel budgets, and so on.

Intangible items, on the other hand, are those whose values cannot be precisely determined and are the result of subjective judgment. For example, how much is saved by completing a project earlier or providing new information to decision makers? Considerable argument can take place before agreement is reached on such intangible costs.

The sum value of costs of items needed to implement the system is known as the system cost. The sum value of the saving made is known as the benefit of the new system. Once we agree on the costs and benefits, we can evaluate whether the project is economically feasible.

The cost estimates are usually used to set the project budget. Often it is convenient to divide these costs into project phases to give management an idea of when funds and personnel will be needed. The cost estimate needs to be worked out very carefully. We will avoid omitting anything from estimates, as this will necessitate requests for more funds because something was forgotten.

On the other side of the evaluation are benefits of the project, which may also tangible or intangible. Tangible benefit includes those benefits that can be measured in actual rupees terms. Such benefit can include reduced production costs through the introduction of new technologies or reduced recessing costs through the use of computers.

Intangible benefits such as more satisfied customer or improved cop rate image because of using new system are not easily quantified.

***Determining Whether This Project is Worthwhile*:**

The costs and benefit are used to determine whether this project is economically feasible or not. There are two ways to do this: the payback and the present value method. Here we use the **payback** method.

The payback method:

The payback method defines the time required to recover the money spent on the project. The concept is quite simple. We know when this project will start. We also know the costs and benefits for each succeeding year. The difference between the cost and the benefit for each year will be saving or net befit for the year. The computation is used to determine the number of years. The computation to determine the number of years needs to recover the cost is quite simple.

5. SYSTEM ANALYSIS

EXISTING SYSTEM

The Existing system is a web based computerized application system but that's not good looking and it is maintained at individual databases i.e., in excels sheets, it's a time delay process. And maintaining all the records in Excel sheets is difficult. If they want any record, they have to search all the records. It doesn't provide multiple user accessibility and also doesn't have different user privileges. So, the system is not accessible for all the members of the organization.

LIMITATIONS IN EXISTING SYSTEM

- There is no centralized database maintenance.
- There is no easy access to the particular employees/student's record.
- The administrator and employee cannot easily navigate through the database.

PROBLEMS IN EXISTING SYSTEM

- SLOW PROCESS: Manual working of managing all the details of employees and generating reports, searching records etc. is very tedious and slow.
- DIFFICULTY IN SEARCHING RECORDS: It is difficult for administrator to manually find the records of different employees.
- DATA INCONSISTENCY: It is difficult to maintain the employees personal account, i.e., the updating employees record and profile.

PROPOSED SYSTEM:

The initial development of the project was started with the visit of ***Laxmi Narayan College in Vaishali District of Bihar***. In order to study the existing system details, we used the technique such as data collection, personal interview and onsite observation etc. After that we work on that proposed system. The Proposed system is a fully squared computerized system but which is maintained at Centralized databases i.e., in automated forms it's a very fast process. And maintaining all the records in online systems database which makes it very easy to access and retrieve data from the database. If they want any employee record, they can easily search all the records. It provides multiple user accessibility and also has different user privileges. So, the system is accessible for all the members of the organization. And also, we improve its interface from existing system. That is done with help of HTML5, CSS and also uses Bootstrap, JavaScript, Django.

ADVANTAGES OVER EXISTING SYSTEM

- It is completely automated system in maintaining employees/student's attendance.
- This system provides centralized database maintenance.
- This system provides easy access to the particular student's/employee attendance.
- This system provides administrator to upload and register the new employee/student to the database with their unique identity as user id and unique password.
- Secure registration and profile management facilities for employee/student's Account.
- Secured mechanism for checking account login for Admin and employee.

Sr. No	Features	Description
1	An effective and an automated process	To overcome the problems of Manual System for going into the shop to purchase items.
2	User friendly interface	Admin can easily insert and update record in the system for future mark of attendance.
3	Reporting	Easy to generate attendance report for particular and all employees/student's records.
4	Searching	Easy to search records for particular and all employees/student's records.
5	Updating	Easy to update records for particular and all employees/student's records.
6	Notifications	Display message for the user and admin to checked data you have to enter in field.

6. SYSTEM REQUIREMENT SPECIFICATION

A software requirements specification (SRS) is a comprehensive description of the intended purpose and environment for software under development. The SRS fully describes what the software will do and how it will be expected to perform.

An SRS minimizes the time and effort required by developers to achieve desired goals and also minimizes the development cost. A good SRS defines how an application will interact with system hardware, other programs and human users in a wide variety of real-world situations. Parameters such as operating speed, response time, availability, portability, maintainability, footprint, security and speed of recovery from adverse events are evaluated. Methods of defining an SRS are described by the IEEE (Institute of Electrical and Electronics Engineers) specification 830-1998.

STUDY OF THE SYSTEM

In the flexibility of uses the interface has been developed a graphics concept in mind, associated through a browser interface. The GUIs at the top level has been categorized as follows.

1. Administrative User Interface Design.
2. Employee User Interface.
3. Security Authentication.
4. Validation User Interface.

The administrative user interface will maintain the records like in-time, out-time, break time and attendance digitally.
The Employee user interface helps the employees to view his profile and he/she is able to get attendance sheet in a tabular format.

MODULES DESCRIPTION

In our system we use the various numbers of modules that are following:

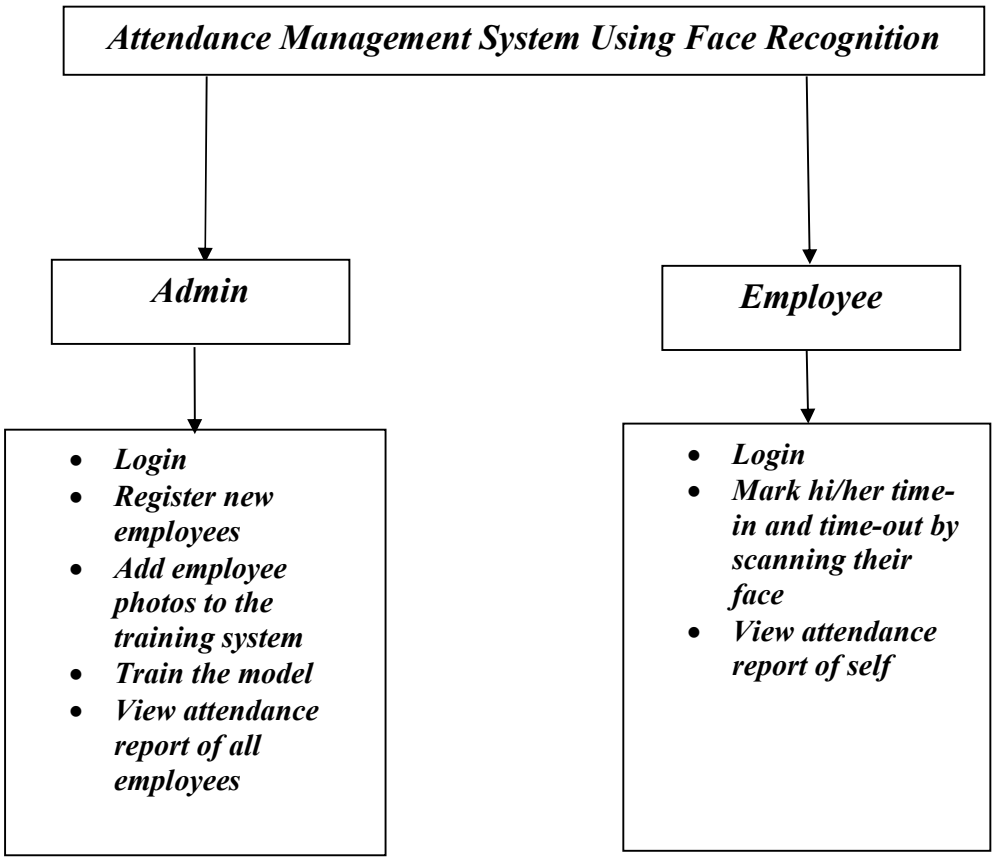
- Administrator
- Employee

ADMINISTRATOR MODULE

In this module of the system, the Administrator mainly deals with the functionalities related to the registration of new employee to the organization, log into the system and managing employee's profile details. After successful registration of employee's, the image of employee's is trained in the training section by entering username of the valid employee by the administrator. In this module administrator have privileged to view total number of employee's present on the specific day and also admin can view total number of work hours of specific or all employees.

EMPLOYEE MODULE

In this module of the system, employee can do the various type of jobs performs like as VIEW PROFILE own profile on any time by login to their personal account using valid login credential. In this module employee have privileged to view its total present day, total absent day, total number of work hours, etc. All the details of employees are view in tabular format and also in graphical format.



7. SOFTWARE & HARDWARE REQUIREMENT

SOFTWARE REQUIREMENTS

Operating System	:	Windows 10/11 or Linux
User Interface	:	HTML, CSS, Bootstrap
Client-side Scripting	:	JavaScript
Web Applications	:	Django
Database	:	SQLITE Database
Communication standard	:	HTTPS
Network server	:	Localhost

HARDWARE REQUIREMENTS

Processor	:	I3 processor-based computer or higher
Hard Disk	:	256GB or more
RAM	:	3GB or more
Camera	:	Working Web Camera with clear image

SUPPORTING BROWSERS

Google chrome/Mozilla

TECHNOLOGY

- Django
- OpenCV
- Dlib
- Open-Source Face Recognition Library
- SQLITE Database
- JavaScript
- Bootstrap

REQUIRED PYTHON MODULES USED

- ***Required python version for running this project: Python 3.7 or Above***

appdirs==1.4.4	asgiref==3.6.0	attrs==22.2.0
backports.zoneinfo==0.2.1	beautifulsoup==4.11.1	black==22.12.0
certifi==2022.12.7	chardet==5.1.0	charset-normalizer==3.0.1
click==8.1.3	cmake==3.25.0	colorama==0.4.6
contourpy==1.0.6	cycler==0.11.0	Cython==0.29.33
Django==4.1.5	django-bootstrap-v5==1.0.11	django-crispy-forms==1.14.0
django-pandas==0.6.6	dlib==19.19.0	face-recognition==1.3.0
face-recognition-models==0.3.0	fonttools==4.38.0	idna==3.4
importlib-resources==5.10.2	imutils==0.5.3	joblib==1.2.0
jsonschema==4.17.3	kiwisolver==1.4.4	matplotlib==3.6.3
mypy-extensions==0.4.3	ninja==1.11.1	numpy==1.24.1
opencv-python==4.4.0.46	pandas==1.5.2	Pillow==9.4.0
regex==2020.11.13	requests==2.28.2	scikit-learn==1.2.0
scipy==1.10.0	seaborn==0.12.2	six==1.16.0
sqlparse==0.4.3	threadpoolctl==3.1.0	toml==0.10.2
tomli==2.0.1	typed-ast==1.4.2	typing_extensions==4.4.0
tzdata==2022.7	urllib3==1.26.14	zipp==3.11.0
pathspec==0.10.3	packaging==23.0	pyproject-toml==0.0.10

8. SDLC METHODOLOGIES

This document plays a vital role in the development of life cycle (SDLC) as it describes the complete requirement of the system. It means for use by developers and will be the basic during testing phase. Any changes made to the requirements in the future will have to go through formal change approval process.

SPIRAL MODEL was defined by Barry Boehm in his 1988 article, "A spiral Model of Software Development and Enhancement. This model was not the first model to discuss iterative development, but it was the first model to explain why the iteration models.

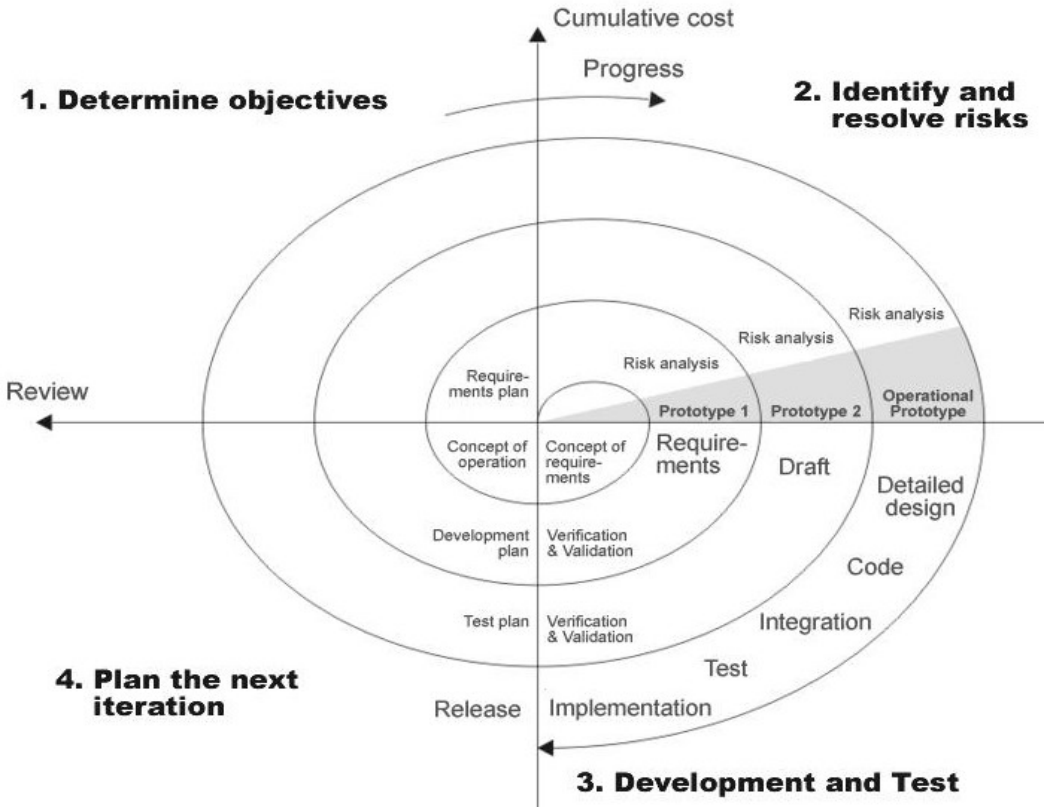
As originally envisioned, the iterations were typically 6 months to 2 years long. Each phase starts with a design goal and ends with a client reviewing the progress thus far. Analysis and engineering efforts are applied at each phase of the project, with an eye toward the end goal of the project.

The steps for Spiral Model can be generalized as follows:

- The new system requirements are defined in as much details as possible. This usually involves interviewing a number of users representing all the external or internal users and other aspects of the existing system.
- A preliminary design is created for the new system.
- A first prototype of the new system is constructed from the preliminary design. This is usually a scaled-down system, and represents an approximation of the characteristics of the final product.
- A second prototype is evolved by a fourfold procedure:
    1. Evaluating the first prototype in terms of its strengths, weakness, and risks.
    2. Defining the requirements of the second prototype.
    3. Planning a designing the second prototype.
    4. Constructing and testing the second prototype.
- At the customer option, the entire project can be aborted if the risk is deemed too great. Risk factors might involve development cost overruns, operating-cost miscalculation, or any other factor that could, in the customer's judgment, result in a less-than-satisfactory final product.

- The existing prototype is evaluated in the same manner as was the previous prototype, and if necessary, another prototype is developed from it according to the fourfold procedure outlined above.
- The preceding steps are iterated until the customer is satisfied that the refined prototype represents the final product desired.
- The final system is constructed, based on the refined prototype.

The final system is thoroughly evaluated and tested. Routine maintenance is carried on a continuing basis to prevent large scale failures and to minimize down time.



9. APPLICATION DEVELOPMENT

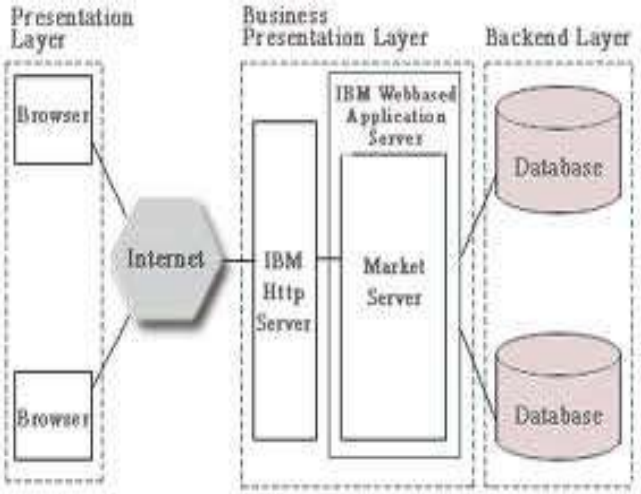
N-TIER APPLICATION

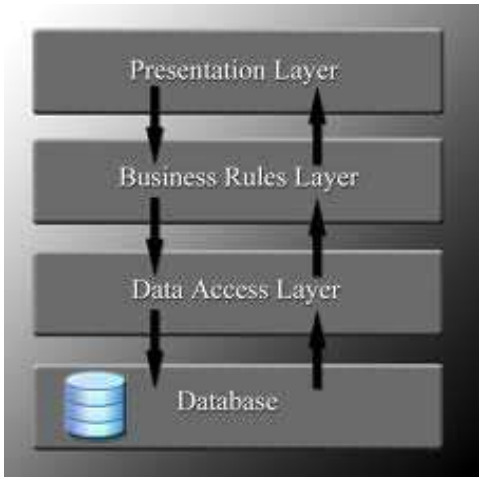
N-Tier Applications can easily implement the concepts of Distributed Application Design and Architecture. The N-Tier Applications provide strategic benefits to Enterprise Solutions. While 2-tier, client-server can help us create quick and easy solutions and may be used for Rapid Prototyping, they can easily become a maintenance and security night mare

The N-tier Applications provide specific advantages that are vital to the business continuity of the enterprise. Typical features of a real-life n-tier may include the following:

- Security
- Availability and Scalability
- Data Abstraction
- Manageability
- Easy Maintenance

The above-mentioned points are some of the key design goals of a successful n-tier application that intends to provide a good Business Solution.





DEFINITION: Simply stated, an n-tier application helps us distribute the overall functionality into various tiers or layers:

- Presentation Layer
- Business Rules Layer
- Data Access Layer

- Database/Data Store

Each layer can be developed independently of the other provided that it adheres to the standards and communicates with the other layers as per the specifications.

This is the one of the biggest advantages of the n-tier application. Each layer can potentially treat the other layer as a 'Block-Box'.

In other words, each layer does not care how other layer processes the data as long as it sends the right data in a correct format.

10. PERFORMANCE REQUIREMENTS

Performance is measured in terms of the output provided by the application. Requirement specification plays an important part in the analysis of a system. Only when the requirement specifications are properly given, it is possible to design a system, which will fit into required environment. It rests largely in the part of the users of the existing system to give the requirement specifications because they are the people who finally use the system. This is because the requirements have to be known during the initial stages so that the system can be designed according to those requirements. It is very difficult to change the system once it has been designed and on the other hand designing a system, which does not cater to the requirements of the user, is of no use.

The requirement specification for any system can be broadly stated as given below:

- The system should be able to interface with the existing system
- The system should be accurate
- The system should be better than the existing system

The existing system is completely dependent on the user to perform all the duties.

11. SYSTEM DESIGN

Design phase is the second phase of the system development life cycle. In this phase computer information system is designed in detailed from system specifications, generated during study phase. A data flow diagram that has been expanded in detail until each of the processing unit, processing function can be identified. A system flow chart that has been expanded in detail. Design phase report is a report prepared at the end of the design phase. Design specification is a baseline specification that serves as a blue print for the construction of a computer-based business information system. Design phase review held with the user organization at the conclusion of the design phase.

In the system design phase, we use the various techniques for system designing phase.

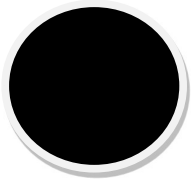
DATA FLOW DIAGRAM

A graphical tool used to describe and analyze the moment of data through a system manual or automated including the process, stores of data, and delays in the system. Data Flow Diagrams are the central tool and the basis from which other components are developed. The transformation of data from input to output, through processes, may be described logically and independently of the physical components associated with the system. The DFD is also known as a data flow graph or a bubble chart.

DFDs are the model of the proposed system. They clearly should show the requirements on which the new system should be built. Later during design activity this is taken as the basis for drawing the system's structure charts. The Basic Notation used to create a DFD's are as follows:

1. **Dataflow:** Data move in a specific direction from an origin to a destination.

2. **Process:** People, procedures, or devices that use or produce (Transform) Data. The physical component is not identified.

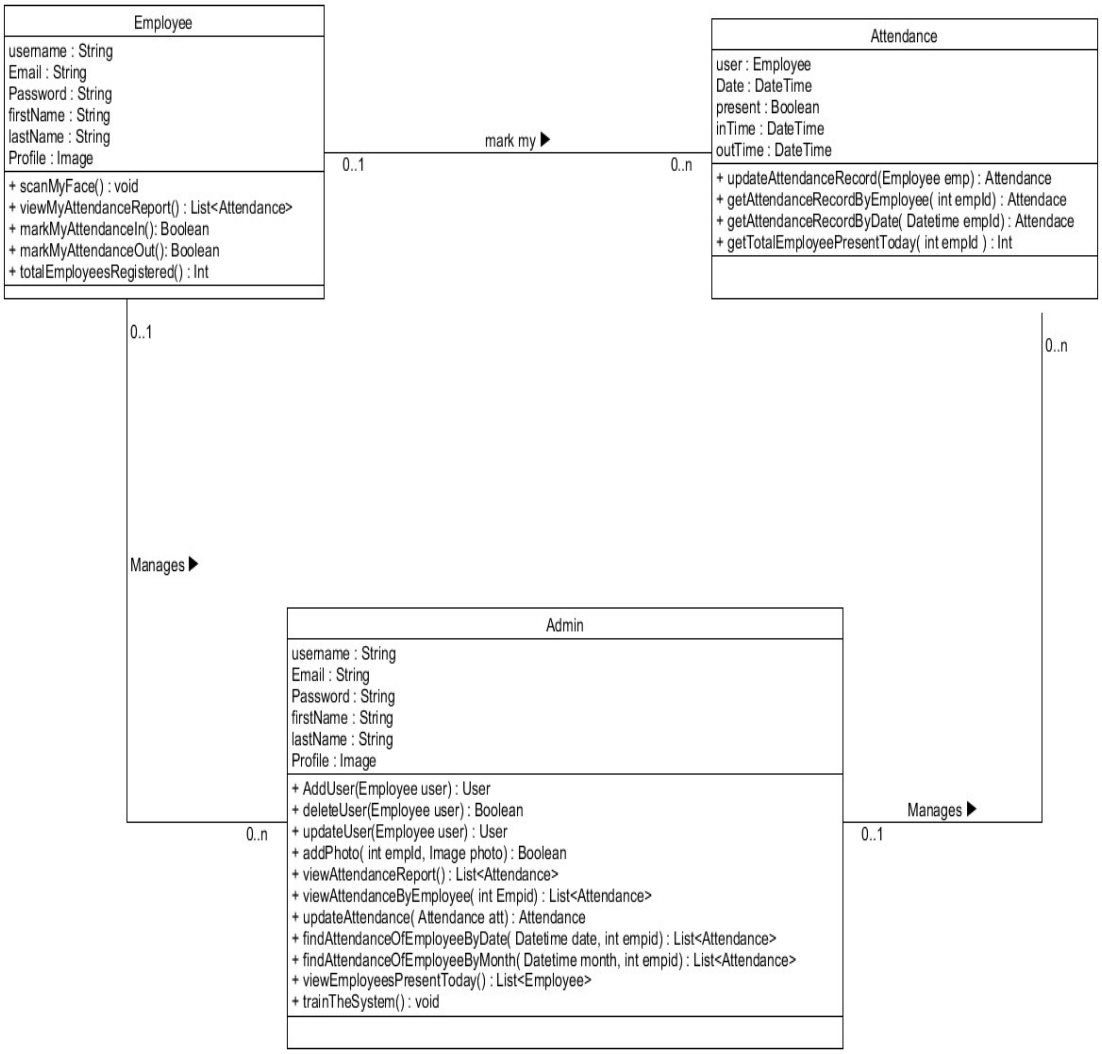


3. **Source:** External sources or destination of data, which may be People, programs, organizations or other entities.

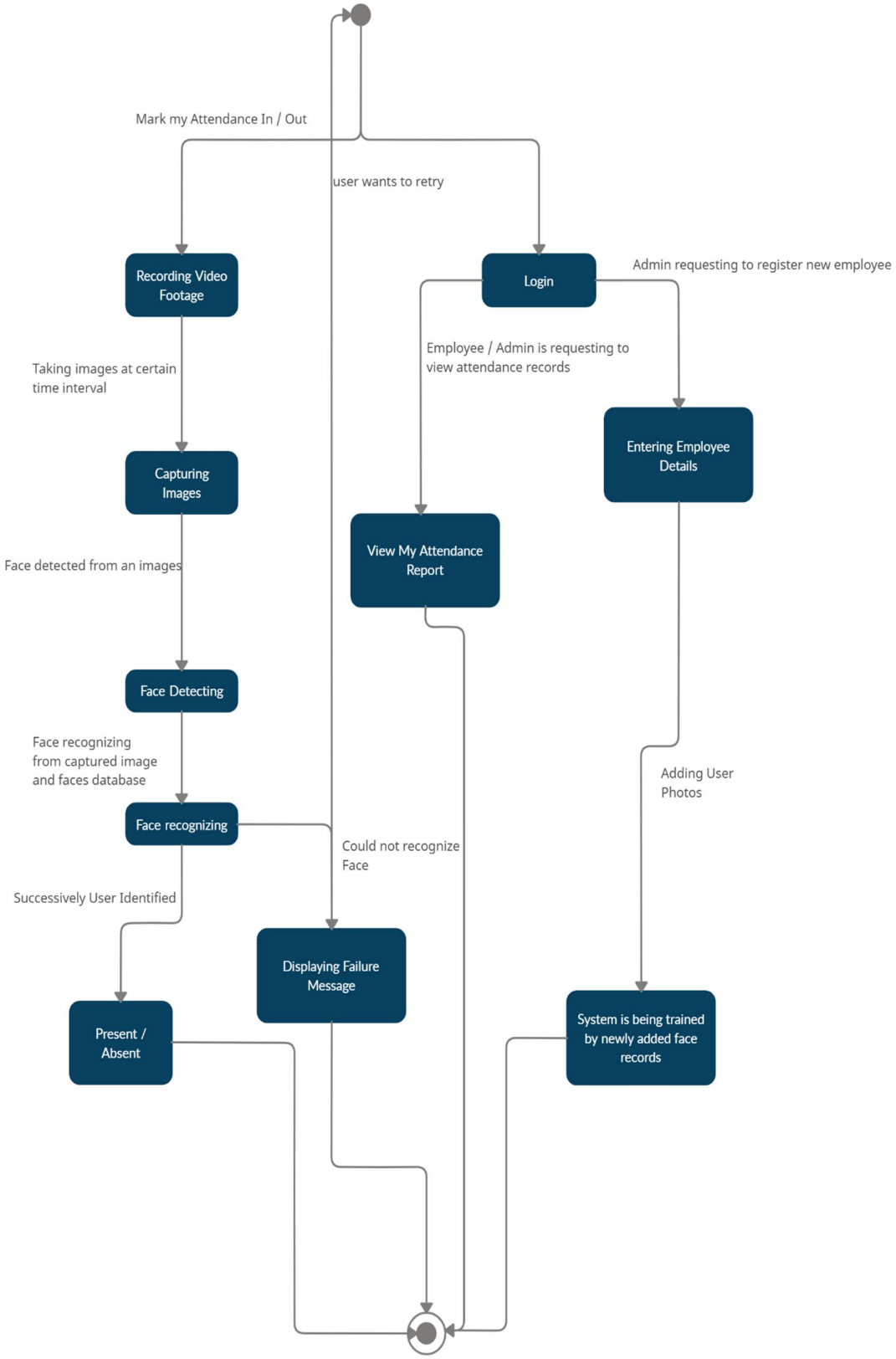
4. **Data Store:** Here data are stored or referenced by a process in the System.

SYSTEM DESIGN

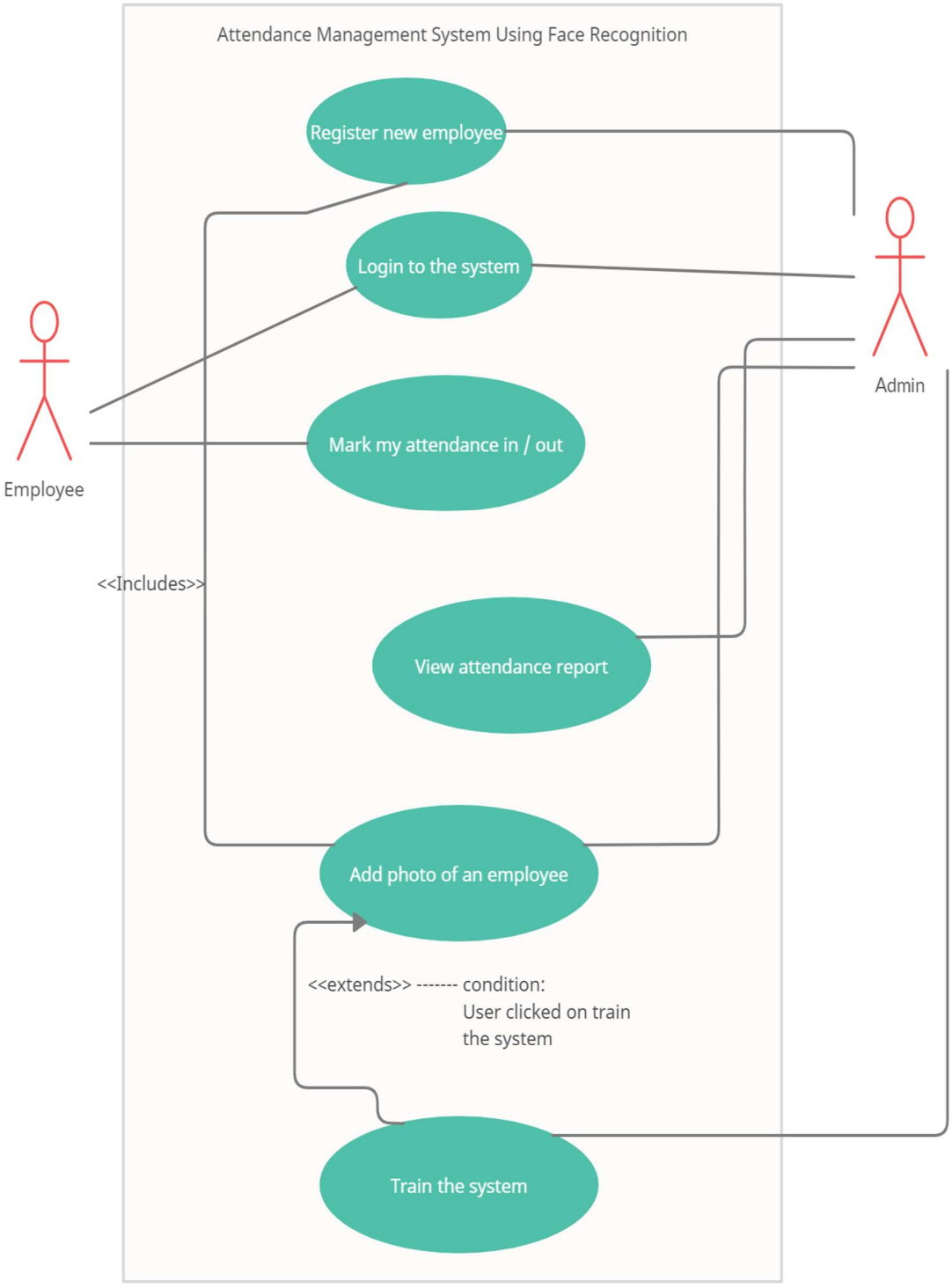
CLASS DIAGRAM



STATE DIAGRAM

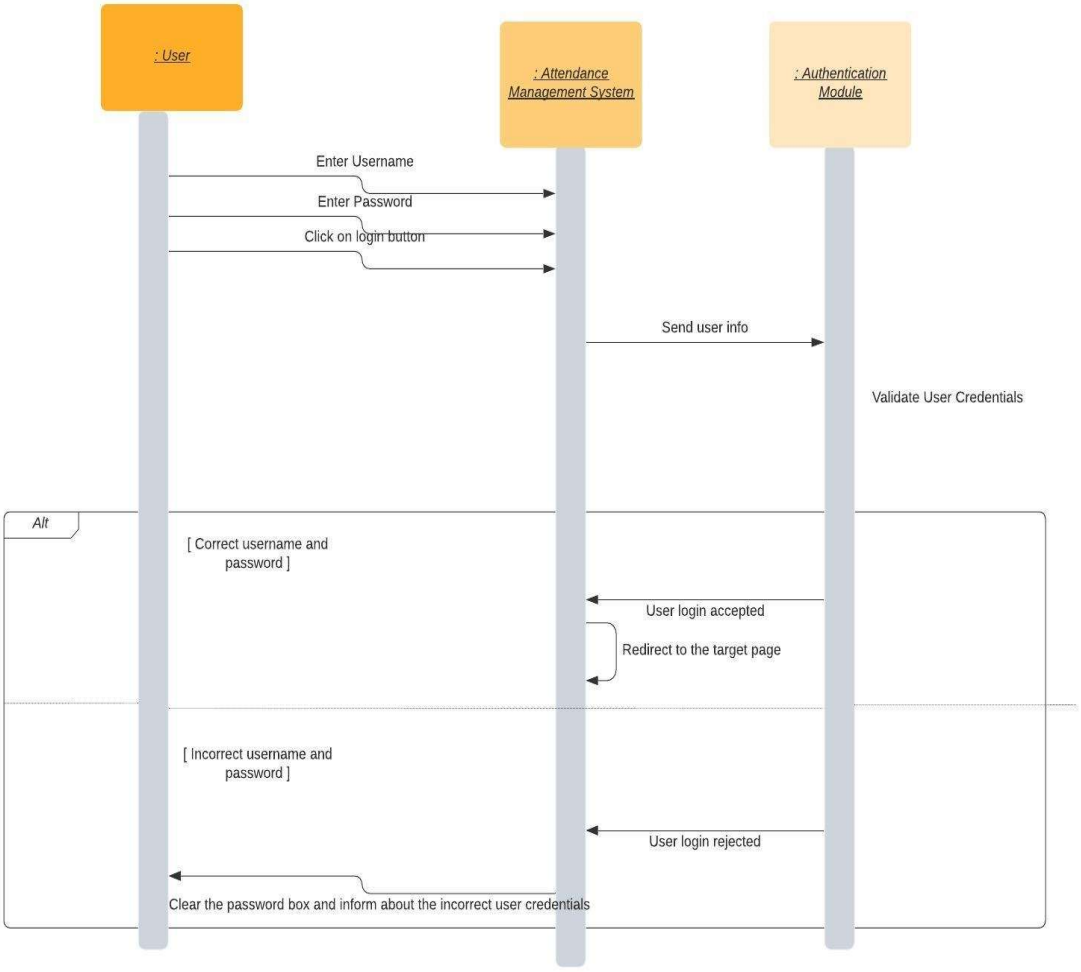


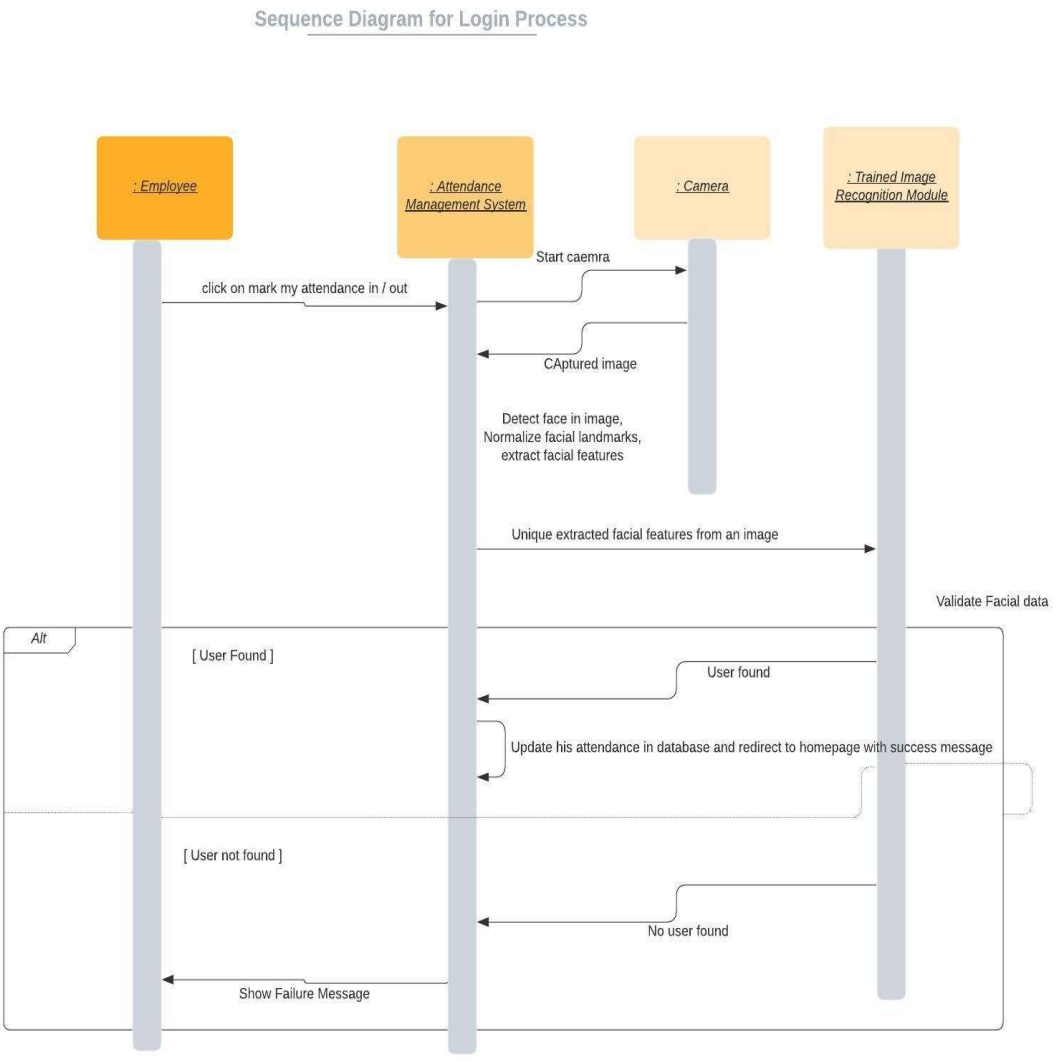
USE CASE DIAGRAM



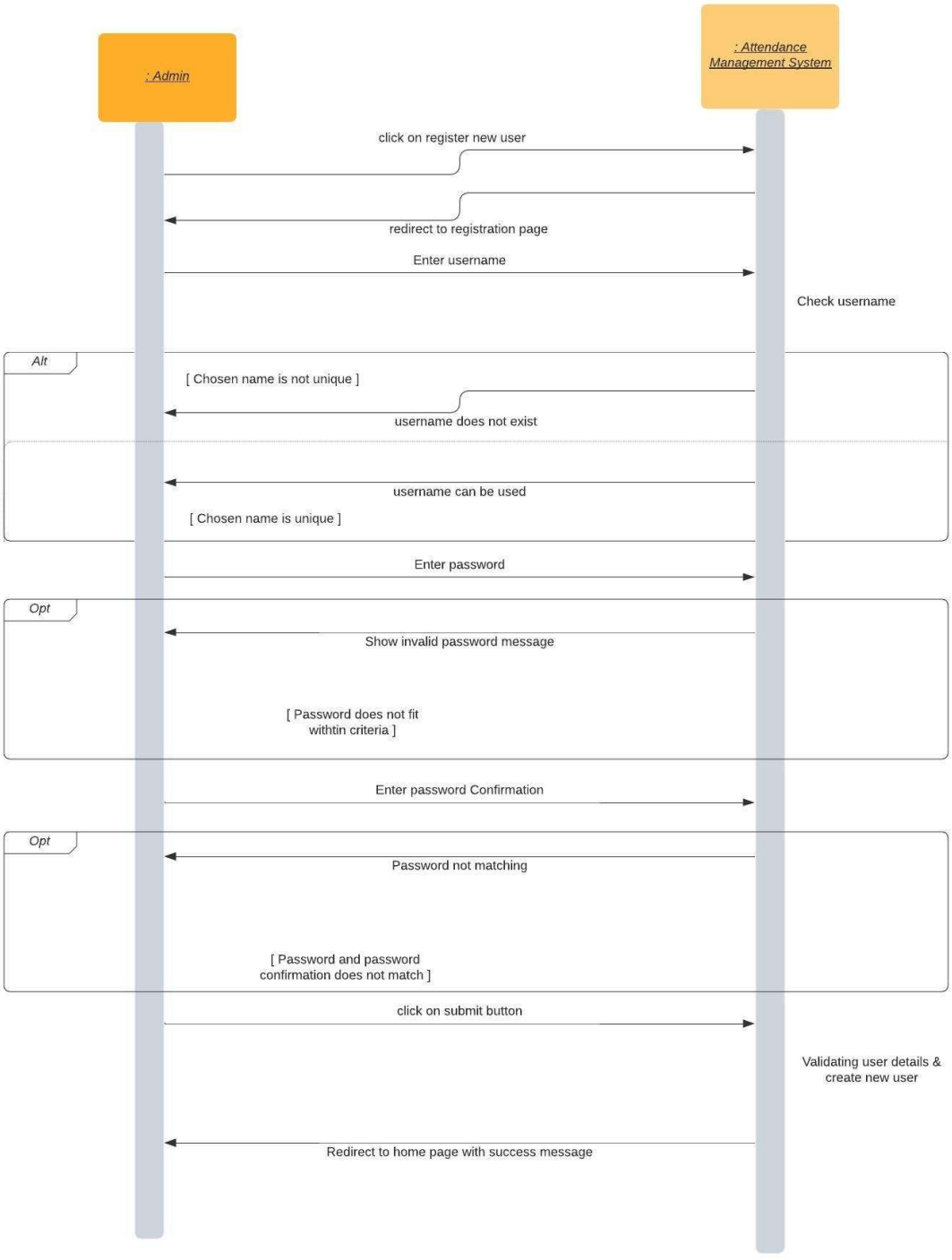
SEQUENCE DIAGRAM

SEQUENCE DIAGRAM FOR LOGIN PROCESS



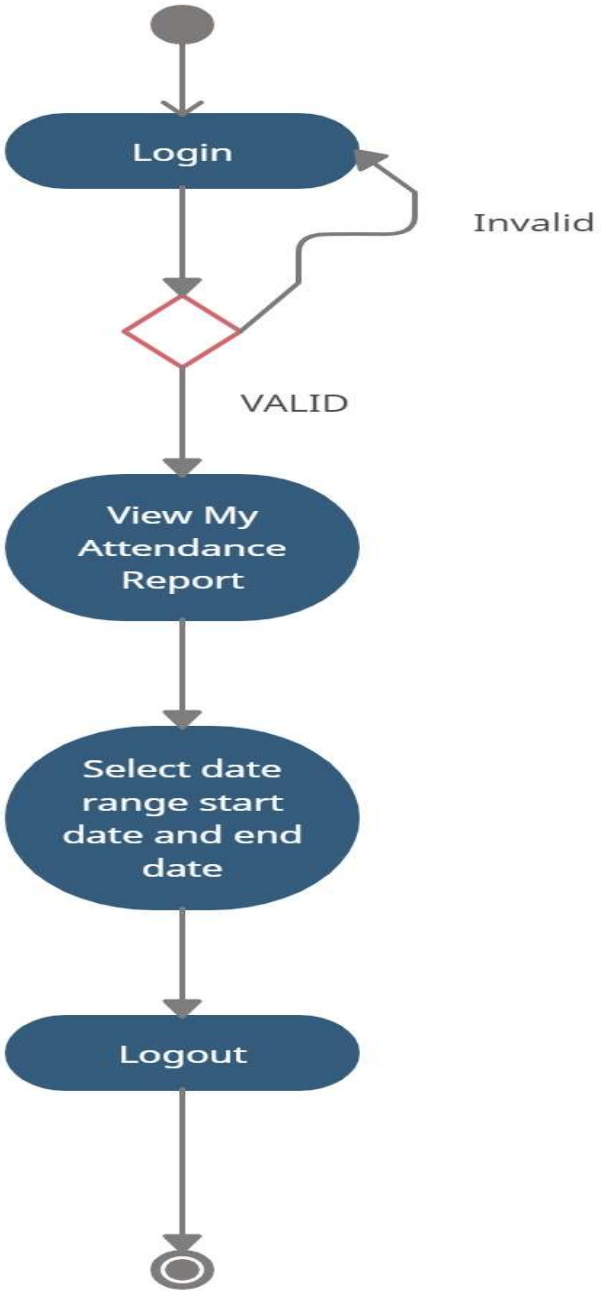

Sequence Diagram for Login Process
: Employee
: Attendance Management System
: Camera
: Trained Image Recognition Module
Start caemra
click on mark my attendance in / out
CAptured image
Detect face in image, Normalize facial landmarks, extract facial features
Unique extracted facial features from an image
Validate Facial data
Alt
[User Found]
User found
Update his attendance in database and redirect to homepage with success message
[User not found]
No user found
Show Failure Message

SEQUENCE DIAGRAM FOR REGISTER USER PROCESS

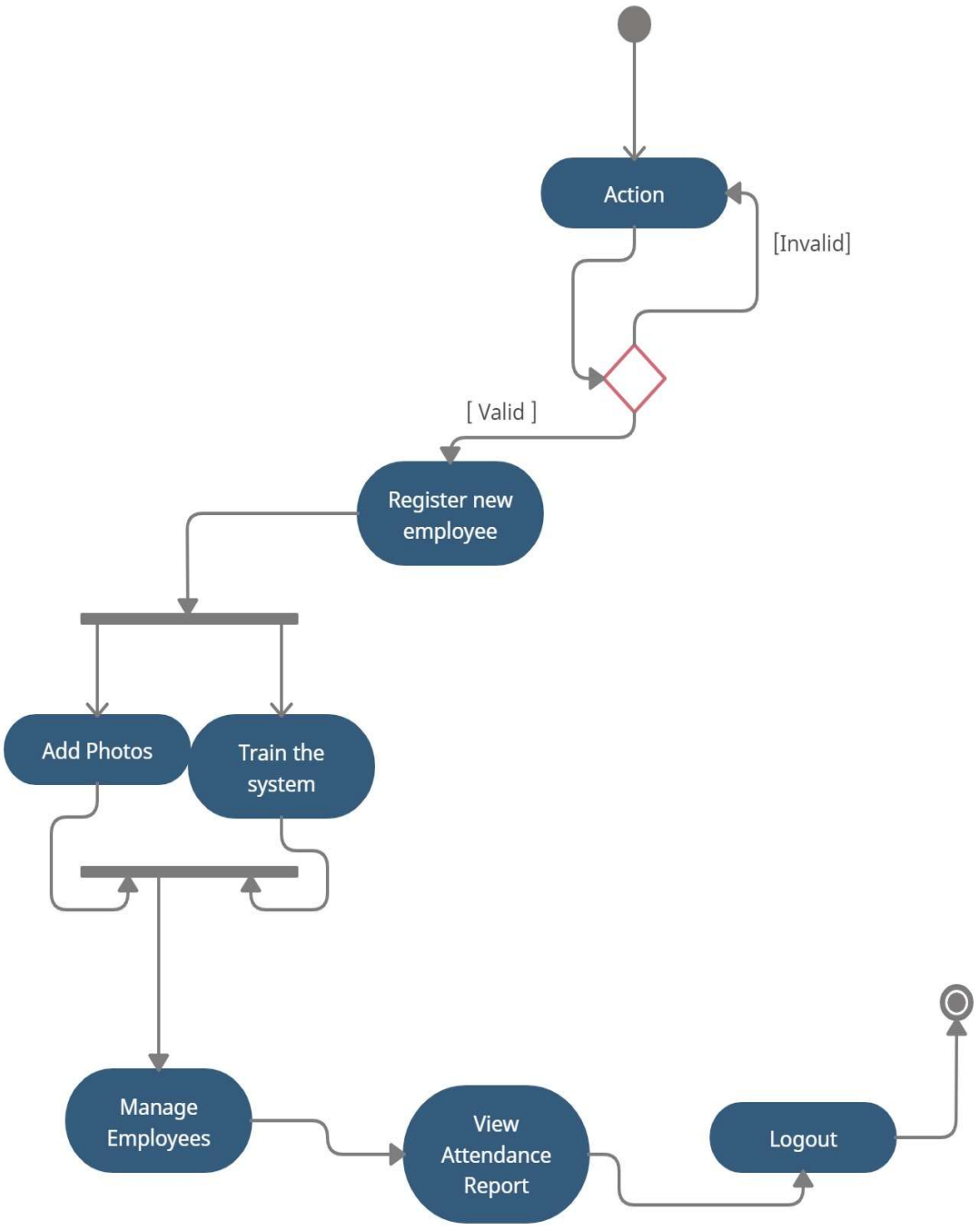


ACTIVITY DIAGRAM

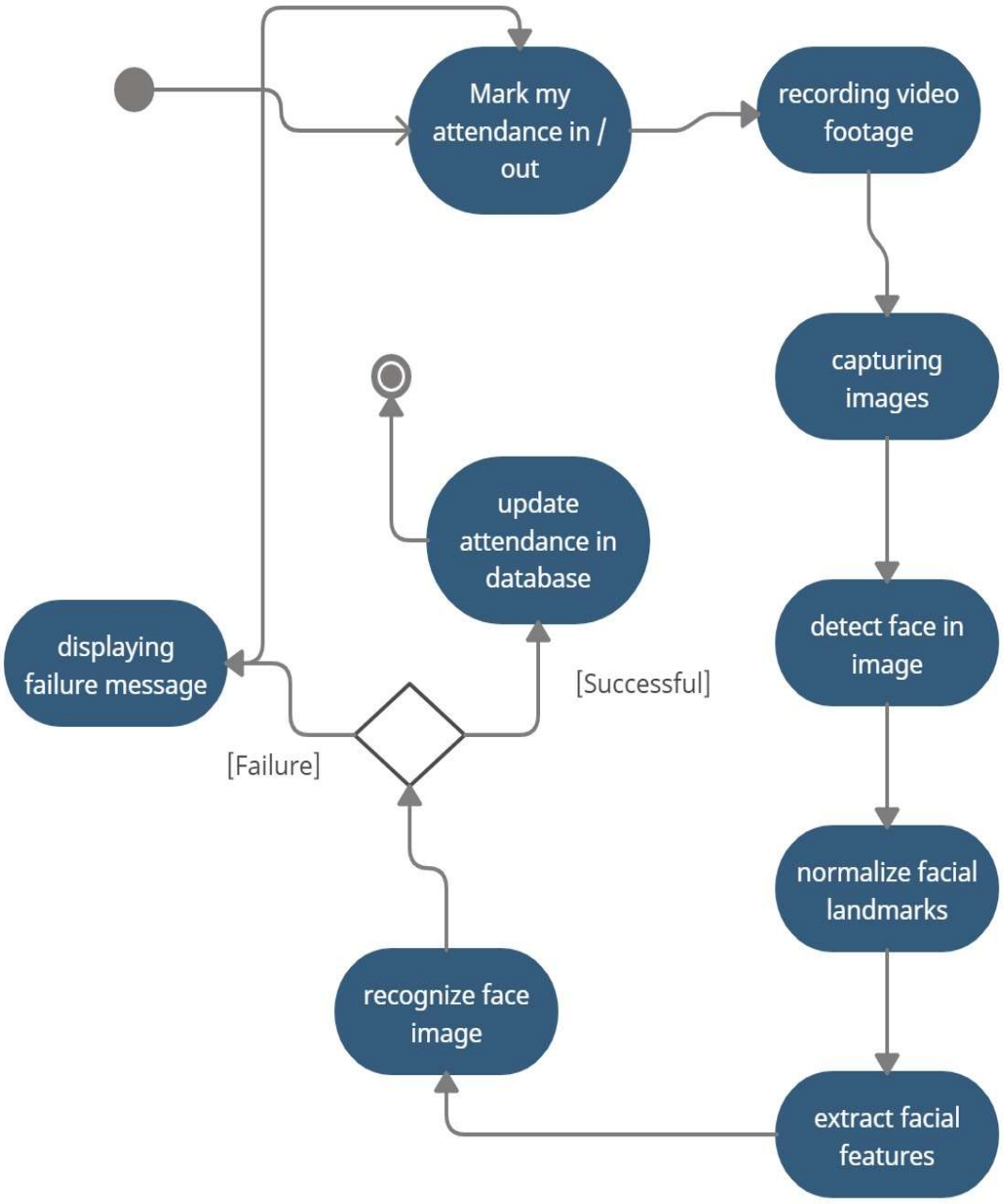
EMPLOYEE ACTIVITY DIAGRAM



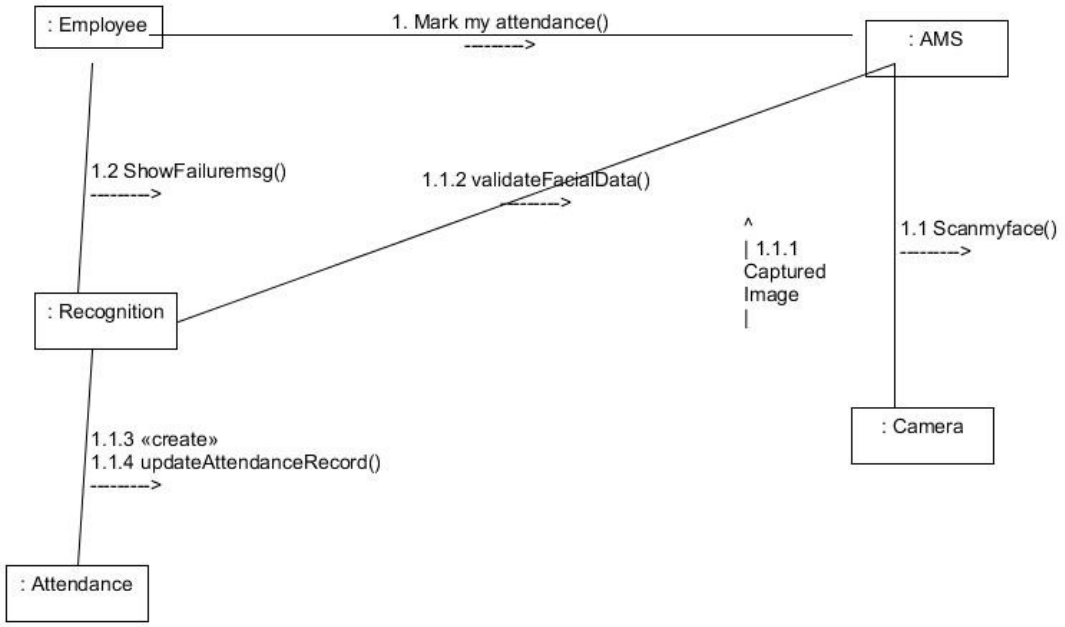
ADMIN ACTIVITY DIAGRAM



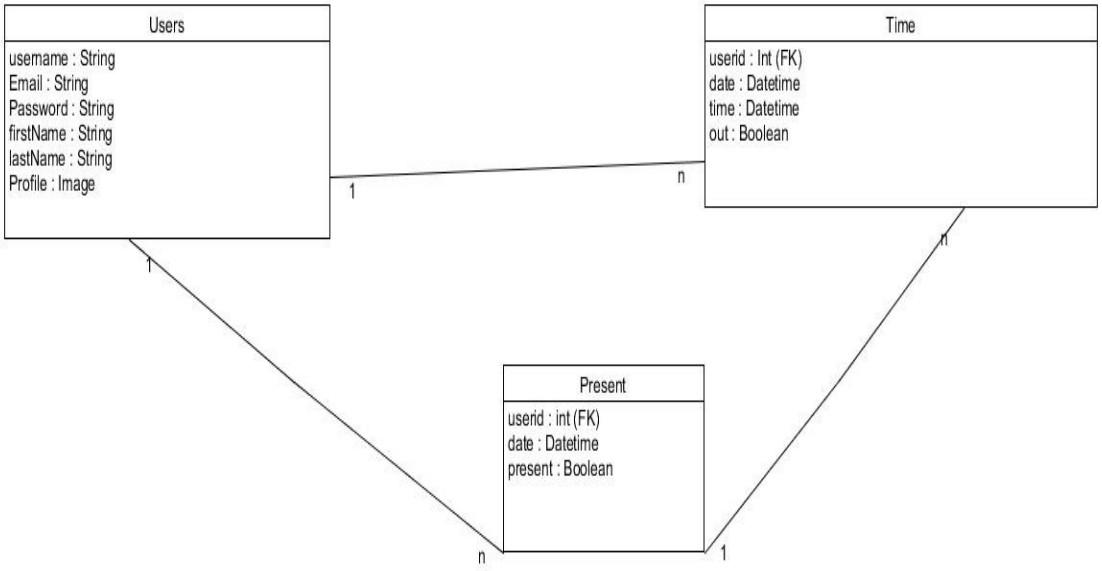
ATTENDANCE TRACKING ACTIVITY DIAGRAM



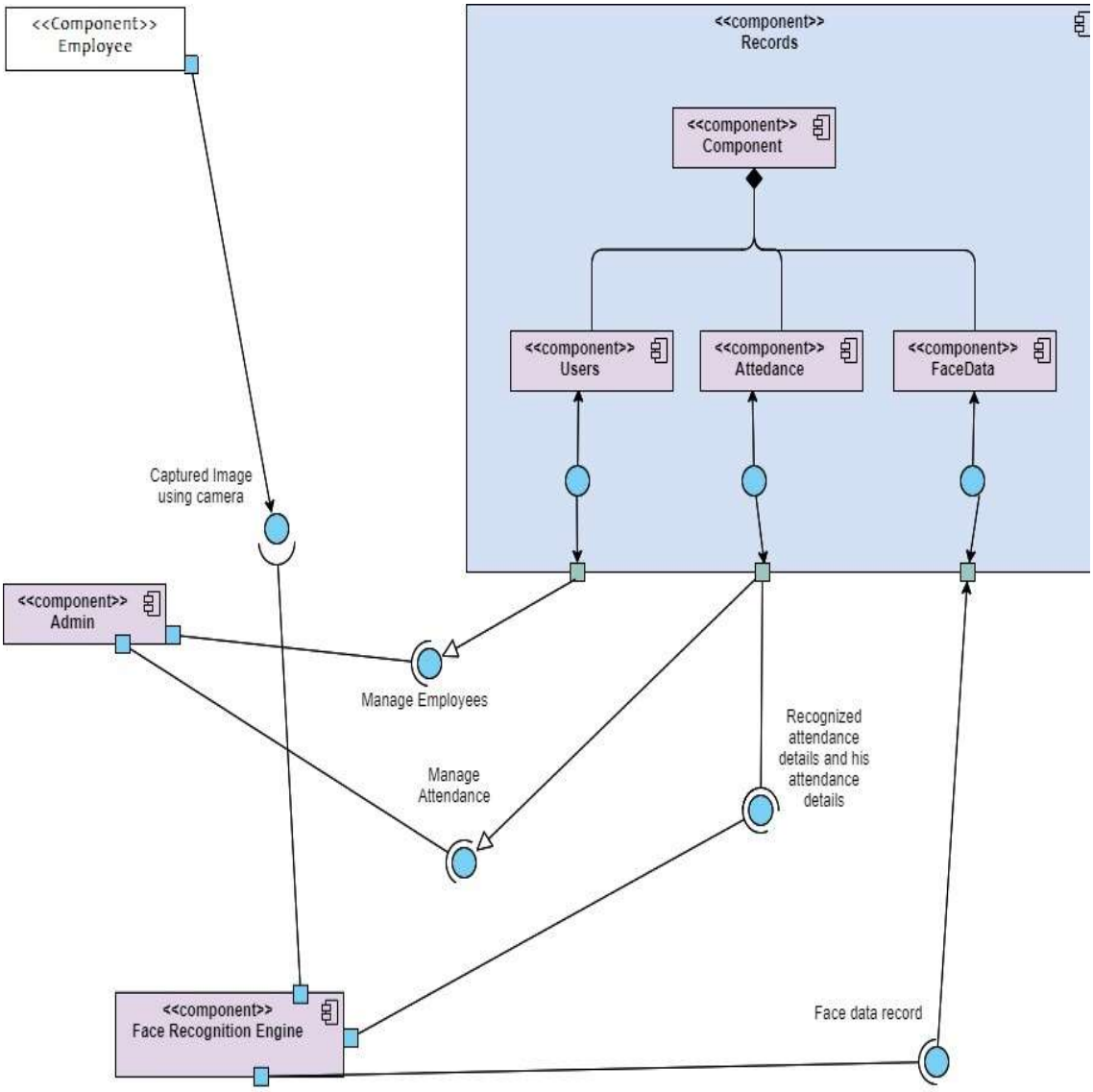
COLLABORATION DIAGRAM



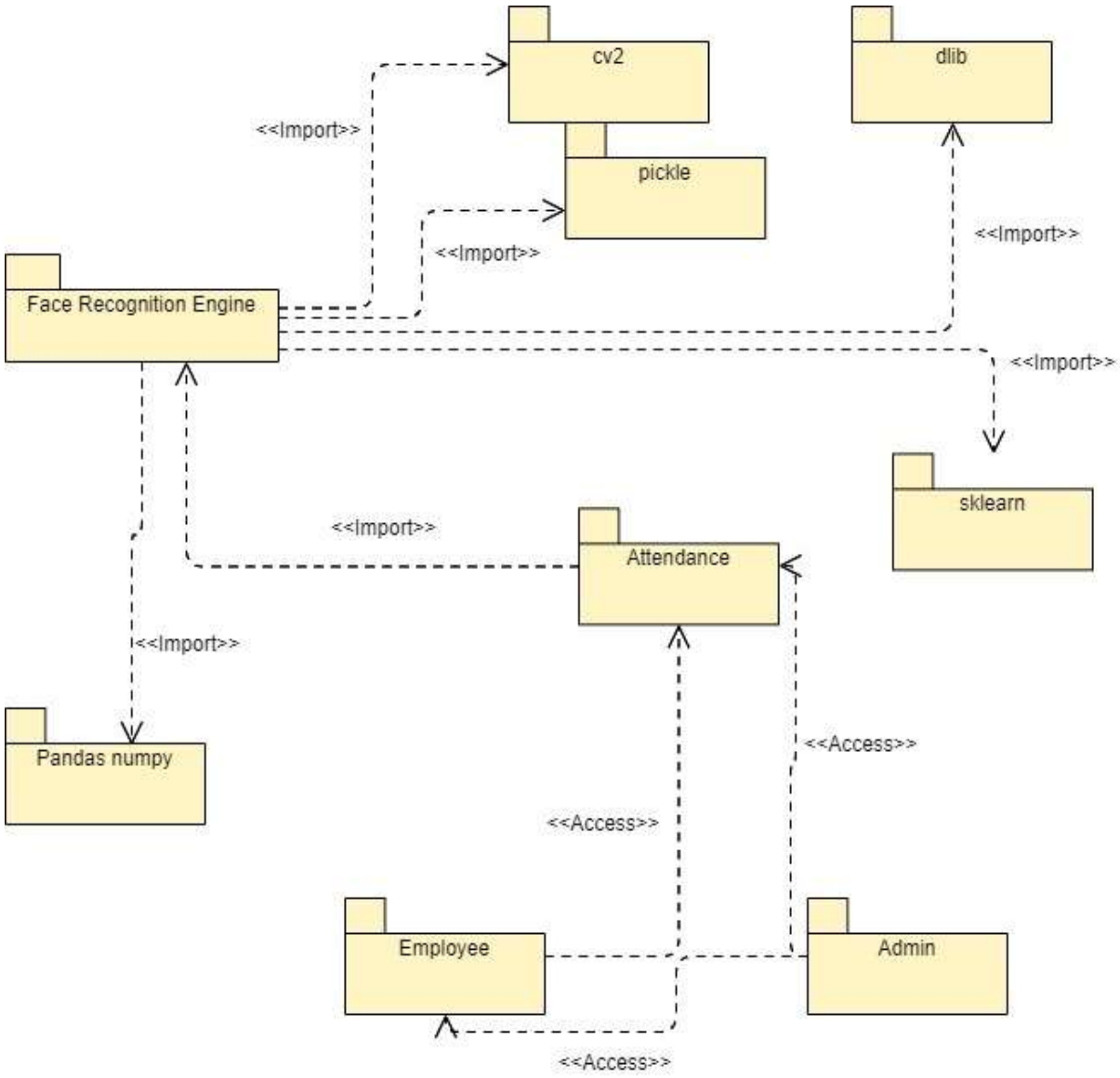
ER DIAGRAM



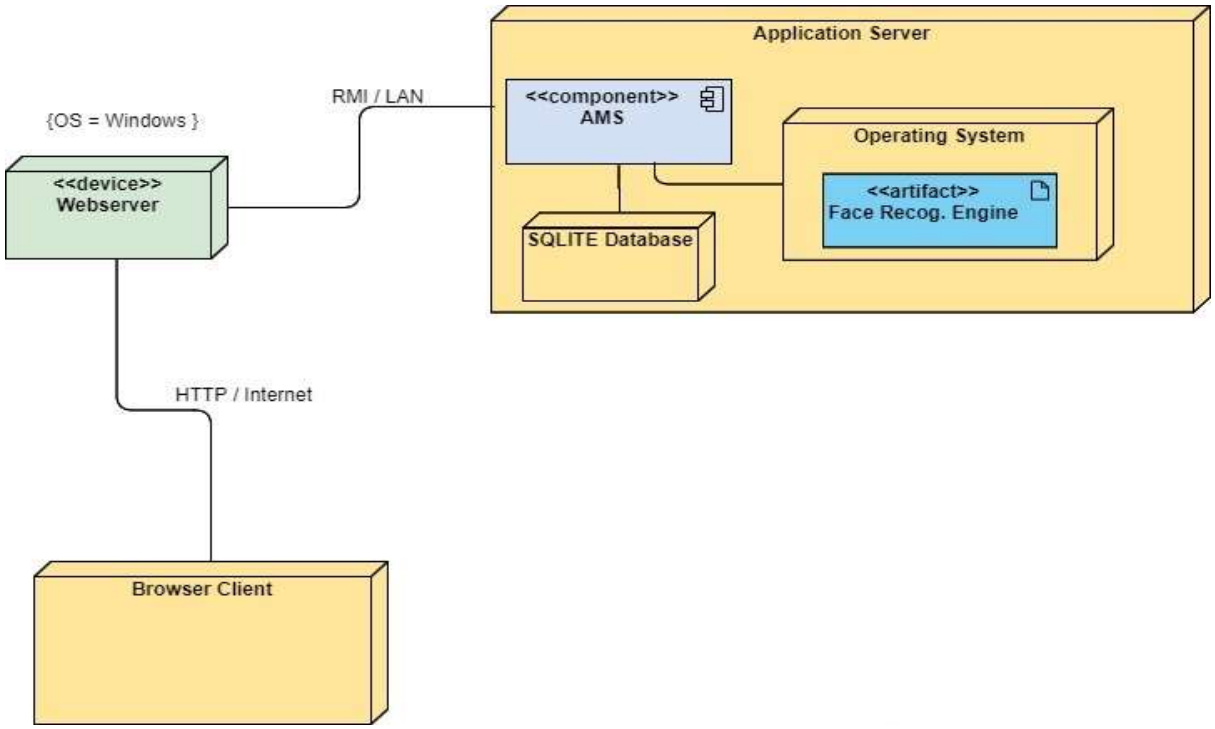
COMPONENT DIAGRAM



PACKAGE DIAGRAM



DEPLOYMENT DIAGRAM



12. DATABASE DESIGN & DEVELOPMENT

Database design

Database is critical for all businesses. A good database does not allow any form of anomalies and stores only relevant information in an ordered manner. If a database has anomalies, it is affecting the efficiency and data integrity. For example, delete anomaly arise upon the deletion of a row which also forces other useful data to be lost. As such, the tables need to be normalized. This fulfils the last objective of ensuring data are accurate and retrieved correctly.

DATA DICTIONARY

USER

S.No	Field name	Data type	Required	Unique	PK / FK	Ref. Table
1	UserId	int	true	true	PK	-
2	Email	string	true	true	-	-
3	Name	string	true	false	-	-
4	Password	string	true	false	-	-
5	CreatedAt	Datetime	true	false	-	-
6	UpdatedAt	Datetime	True	False	-	-

PRESENT

S.No	Field name	Data type	Required	Unique	PK / FK	Ref. Table
1	PId	int	true	true	PK	-
2	Date	Datetime	True	False	-	-
3	User	User	True	False	FK	Users
4	Present	Boolean	True	False	-	-

TIME

S.No	Field name	Data type	Required	Unique	PK / FK	Ref. Table
1	TId	int	true	true	PK	-
2	Date	Datetime	true	True	-	-
3	User	Users	True	False	FK	Users
4	Time	Datetime	False	False	-	-
5	Out	Boolean	True	False	-	-

IMPLEMENTATION DETAILS

Modules

The features of the system are mainly divided into 3 modules.

Registration and Login Module

This module mainly deals with the functionalities related to the registration of any new employee to the organization, Log into the system and managing employee's profile details. Using features provided by this module admin can register new employee to the system and admin / employee both can log into the system using their credentials.

Manage Attendance Details

This module mainly deals with the features related to the employee's attendance. Using this employee can mark their presence, time-in and time-out in the system. Admin can see the availability report of each employee, employee can see his/her attendance report along with some possible filters such as filter by employee and filter by date.

Manage Employee Details

This module mainly deals with the features related to the employee's profile. Using this admin can add a photo of the newly registered employee during registration. Admin can also command the system explicitly to train the model and system will make necessary calculation and will generate some data which will be used internally to identify each employee uniquely.

Function prototypes which implement major functionality

- List<Attendance> viewMyAttendanceReport(int empId);
- Int totalEmployeesRegistered();
- List<Attendance> getAttendanceRecordByEmployee(int empid);
- Boolean updateAttendanceRecord(int empid, Attendance update);
- Boolean registerEmployee(Employee new_employee);
- Boolean addPhoto(int empId, string photo);

13. TECHNOLOGY DESCRIPTION

HTML

HTML, an initialize of Hypertext Markup Language, is the predominant markup language for web pages. It provides a means to describe the structure of text-based information in a document — by denoting certain text as headings, paragraphs, lists, and so on — and to supplement that text with interactive forms, embedded images, and other objects. HTML is written in the form of labels (known as tags), surrounded by angle brackets. HTML can also describe, to some degree, the appearance and semantics of a document, and can include embedded scripting language code which can affect the behavior of web browsers and other HTML processors.

HTML is also often used to refer to content of the MIME type text/html or even more broadly as a generic term for HTML whether in its XML-descended form (such as XHTML 1.0 and later) or its form descended directly from SGML

Hypertext Markup Language (HTML), the languages of the World Wide Web (WWW), allows users to produces Web pages that include text, graphics and pointer to other Web pages (Hyperlinks).

HTML is not a programming language but it is an application of ISO Standard 8879, SGML (Standard Generalized Markup Language), but specialized to hypertext and adapted to the Web. The idea behind Hypertext is that instead of reading text in rigid linear structure, we can easily jump from one point to another point. We can navigate through the information based on our interest and preference. A markup language is simply a series of elements, each delimited with special characters that define how text or other items enclosed within the elements should be displayed. Hyperlinks are underlined or emphasized works that load to other documents or some portions of the same document.

HTML can be used to display any type of document on the host computer, which can be geographically at a different location. It is a versatile language and can be used on any platform or desktop.

HTML provides tags (special codes) to make the document look attractive. HTML tags are not case-sensitive. Using graphics, fonts, different sizes, color, etc., can enhance the presentation of the document. Anything that is not a tag is part of the document itself.

Basic HTML Tags:

<! -- -->	specifies comments
<A>……</A>	Creates hypertext links
<B>……</B>	Formats text as bold
<BIG>……</BIG>	Formats text in large font.
<BODY>…</BODY>	Contains all tags and text in the HTML document
<CENTER>...</CENTER>	Creates text
<DD>…</DD>	Definition of a term
<DL>...</DL>	Creates definition list
<FONT>…</FONT>	Formats text with a particular font
<FORM>...</FORM>	Encloses a fill-out form
<FRAME>...</FRAME>	Defines a particular frame in a set of frames
<H#>…</H#>	Creates headings of different levels (1–6)
<HEAD>...</HEAD>	Contains tags that specify information about a document
<HR>...</HR>	Creates a horizontal rule
<HTML>…</HTML>	Contains all other HTML tags
<META>...</META>	Provides meta-information about a document
<SCRIPT>…</SCRIPT>	Contains client-side or server-side script
<TABLE>…</TABLE>	Creates a table
<TD>…</TD>	Indicates table data in a table
<TR>…</TR>	Designates a table row
<TH>…</TH>	Creates a heading in a table

Attributes

The attributes of an element are name-value pairs, separated by "=", and written within the start label of an element, after the element's name. The value should be enclosed in single or double quotes, although values consisting of certain characters can be left unquoted in HTML (but not XHTML). Leaving attribute values unquoted is considered unsafe.

Most elements take any of several common attributes: id, class, style and title. Most also take language-related attributes: Lang and dir.

The id attribute provides a document-wide unique identifier for an element. This can be used by stylesheets to provide presentational properties, by browsers to focus attention on the

specific element or by scripts to alter the contents or presentation of an element. The class attribute provides a way of classifying similar elements for presentation purposes. For example, an HTML document (or a set of documents) may use the designation class="notation" to indicate that all elements with this class value are all subordinate to the main text of the document (or documents). Such notation classes of elements might be gathered together and presented as footnotes on a page, rather than appearing in the place where they appear in the source HTML.

An author may use the style non-attributable codes presentational properties to a particular element. It is considered better practice to use an element's son- id page and select the element with a stylesheet, though sometimes this can be too cumbersome for a simple ad hoc application of styled properties. The title is used to attach subtextual explanation to an element. In most browsers this title attribute is displayed as what is often referred to as a tooltip. The generic inline span element can be used to demonstrate these various non-attributes.

The preceding displays as HTML (pointing the cursor at the abbreviation should display the title text in most browsers).

Advantages

- A HTML document is small and hence easy to send over the net. It is small because it does not include formatted information.
- HTML is platform independent.
- HTML tags are not case-sensitive.

JAVA SCRIPT

JavaScript is a script-based programming language that was developed by Netscape Communication Corporation. JavaScript was originally called Live Script and renamed as JavaScript to indicate its relationship with Java. JavaScript supports the development of both client and server components of Web-based applications. On the client side, it can be used to write programs that are executed by a Web browser within the context of a Web page. On the server side, it can be used to write Web server programs that can process information submitted by a Web browser and then update the browser's display accordingly

Even though JavaScript supports both client and server Web programming, we prefer JavaScript at Client-side programming since most of the browsers supports it. JavaScript is

almost as easy to learn as HTML, and JavaScript statements can be included in HTML documents by enclosing the statements between a pair of scripting tags

<SCRIPT>……. </SCRIPT>.

<SCRIPT LANGUAGE = "JavaScript">

JavaScript statements

</SCRIPT>

Here are a few things we can do with JavaScript:

- Validate the contents of a form and make calculations.
- Add scrolling or changing messages to the Browser's status line.
- Animate images or rotate images that change when we move the mouse over them.
- Detect the browser in use and display different content for different browsers.
- Detect installed plug-ins and notify the user if a plug-in is required.

We can do much more with JavaScript, including creating entire application.

BOOTSTRAP

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains HTML, CSS and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and Bootstrap is an HTML, CSS and JS library that focuses on simplifying the development of informative web pages (as opposed to web applications). The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all HTML elements. The result is a uniform appearance for prose, tables and form elements across web browsers. In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents. For example, Bootstrap has provisioned for light- and dark-colored tables, page headings, more prominent pull quotes, and text with a highlight.

Bootstrap also comes with several JavaScript components which do not require other libraries like jQuery. They provide additional user interface elements such as dialog boxes, tooltips, progress bars, navigation drop-downs, and carousels. Each Bootstrap component consists of an HTML structure, CSS declarations, and in some cases accompanying JavaScript code.

They also extend the functionality of some existing interface elements, including for example an auto-complete function for input fields.

Bootstrap, originally named Twitter Blueprint, was developed by Mark Otto and Jacob Thornton at Twitter as a framework to encourage consistency across internal tools. Before Bootstrap, various libraries were used for interface development, which led to inconsistencies and a high maintenance burden. According to Otto:

A super small group of developers and I got together to design and build a new internal tool and saw an opportunity to do something more. Through that process, we saw ourselves build something much more substantial than another internal tool. Months later, we ended up with an early version of Bootstrap as a way to document and share common design patterns and assets within the company. After a few months of development by a small group, many developers at Twitter began to contribute to the project as a part of Hack Week, a hackathon-style week for the Twitter development team. It was renamed from Twitter Blueprint to Bootstrap and released as an open-source project on August 19, 2011. It has continued to be maintained by Otto, Thornton, a small group of core developers, and a large community of contributors.

PYTHON

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation via the off-side rule. Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library. Guido van Rossum began working on Python in the late 1980s as a successor to the ABC programming language and first released it in 1991 as Python 0.9.0. Python 2.0 was released in 2000. Python 3.0, released in 2008, was a major revision not completely backward-compatible with earlier versions. Python 2.7.18, released in 2020, was the last release of Python 2. Python consistently ranks as one of the most popular programming languages.

DJANGO

Django is a Python-based web framework that allows you to quickly create efficient web applications. It is also called batteries included framework because Django provides built-in features for everything including Django Admin Interface, default database – SQLlite3, etc. When you're building a website, you always need a similar set of components: a way to handle user authentication (signing up, signing in, signing out), a management panel for your website, forms, a way to upload files, etc. Django gives you ready-made components to use and that too for rapid development.

OPENCV

OpenCV is the huge open-source library for the computer vision, machine learning, and image processing and now it plays a major role in real-time operation which is very important in today's systems. By using it, one can process images and videos to identify objects, faces, or even handwriting of a human. When it integrated with various libraries, such as NumPy, python is capable of processing the OpenCV array structure for analysis. To Identify image pattern and its various features we use vector space and perform mathematical operations on these features. The first OpenCV version was 1.0. OpenCV is released under a BSD license and hence it's free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. When OpenCV was designed the main focus was real-time applications for computational efficiency. All things are written in optimized C/C++ to take advantage of multi-core processing.

Applications of OpenCV: There are lots of applications which are solved using OpenCV, some of them are listed below:

- face recognition
- Automated inspection and surveillance
- number of people – count (foot traffic in a mall, etc)
- Vehicle counting on highways along with their speeds
- Interactive art installations
- Anomaly (defect) detection in the manufacturing process (the odd defective products)
- Street view image stitching

- Video/image search and retrieval
- Robot and driver-less car navigation and control
- object recognition
- Medical image analysis
- Movies – 3D structure from motion
- TV Channels advertisement recognition

OpenCV Functionality

- Image/video I/O, processing, display (core, imgproc, highgui)
- Object/feature detection (objdetect, features2d, nonfree)
- Geometry-based monocular or stereo computer vision (calib3d, stitching, videostab)
- Computational photography (photo, video, superres)
- Machine learning & clustering (ml, flann)
- CUDA acceleration (gpu)

14. DEVELOPMENT TOOLS

VISUAL STUDIO CODE

Visual Studio Code, also commonly referred to as VS Code, is a source-code editor made by Microsoft with the Electron Framework, for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add functionality. In the Stack Overflow 2022 Developer Survey, Visual Studio Code was ranked the most popular developer environment tool among 71,010 respondents, with 74.48% reporting that they use it.

History

Visual Studio Code was first announced on April 29, 2015, by Microsoft at the 2015 Build conference. A preview build was released shortly thereafter. On November 18, 2015, the source of Visual Studio Code was released under the MIT License, and made available on GitHub. Extension support was also announced. On April 14, 2016, Visual Studio Code graduated from the public preview stage and was released to the Web. Microsoft has released most of Visual Studio Code's source code on GitHub under the permissive MIT License, while the releases by Microsoft are proprietary freeware.

Features

Visual Studio Code is a source-code editor that can be used with a variety of programming languages, including C, C#, C++, Fortran, Go, Java, JavaScript, Node.js, Python, Rust. It is based on the Electron framework, which is used to develop Node.js web applications that run on the Blink layout engine. Visual Studio Code employs the same editor component (codenamed "Monaco") used in Azure DevOps (formerly called Visual Studio Online and Visual Studio Team Services). Out of the box, Visual Studio Code includes basic support for most common programming languages. This basic support includes syntax highlighting, bracket matching, code folding, and configurable snippets. Visual Studio Code also ships with IntelliSense for JavaScript, TypeScript, JSON, CSS, and HTML, as well as debugging support for Node.js. Support for additional languages can be provided by freely available extensions on the VS Code Marketplace.

PYCHARM

PyCharm is an integrated development environment (IDE) used for programming in Python. It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems, and supports web development with Django. PyCharm is developed by the Czech company JetBrains. It is cross-platform, working on Microsoft Windows, macOS and Linux. PyCharm has a Professional Edition, released under a proprietary license and a Community Edition released under the Apache License. PyCharm Community Edition is less extensive than the Professional Edition.

Features

- Coding assistance and analysis, with code completion, syntax and error highlighting, linter integration, and quick fixes
- Project and code navigation: specialized project views, file structure views and quick jumping between files, classes, methods and usages
- Python code refactoring: including rename, extract method, introduce variable, introduce constant, pull up, push down and others
- Support for web frameworks: Django, web2py and Flask
- Integrated Python debugger
- Integrated unit testing, with line-by-line coverage
- Google App Engine Python development
- Version control integration: unified user interface for Mercurial, Git, Subversion, Perforce and CVS with change lists and merge
- Scientific tools integration: integrates with IPython Notebook, has an interactive Python console, and supports Anaconda as well as multiple scientific packages including Matplotlib and NumPy.

History

PyCharm was released to the market of the Python-focused IDEs to compete with PyDev (for Eclipse) or the more broadly focused Komodo IDE by ActiveState. The beta version of the product was released in July 2010, with the 1.0 arriving 3 months later. Version 2.0 was released on 13 December 2011, version 3.0 was released on 24 September 2013, and version 4.0 was released on November 19, 2014.

PyCharm became Open Source on 22 October 2013. The Open-Source variant is released under the name Community Edition – while the commercial variant, Professional Edition, contains closed-source modules.

15. TESTING AND DEBUGGING

After successful requirement analysis, designing & coding, the testing and debugging phase was mandatory for successful completion of the project because the importance of software testing to software quality cannot be overemphasized. Once source code has been generated, software must be tested to allow errors to be identified and removed before delivery to the customer. While it is not possible to remove every error in a large software package, the software engineer's goal is to remove as many possible early in the software development cycle. It is important to remember that testing can only find errors; it cannot prove that a program is bug free. Two basic test techniques involve testing module input/output (black-box) and exercising internal logic of software components (white-box). Formal technical reviews by themselves cannot find allow software defects, test data must also be used. For large software projects, separate test teams may be used to develop and execute the set of test cases used in testing. Testing must be planned and designed. The SEPA web site contains template for a generic test plan.

SOFTWARE TESTING OBJECTIVES:

- Testing is the process of executing a program with the intent of finding errors.
- A good test case is one with a high probability of finding an as-yet undiscovered error.
- A successful test is one that discovers an as-yet-undiscovered error.

SOFTWARE TESTING PRINCIPLES:

- All tests should be traceable to customer requirements.
- Test should be planned long before testing begins.
- The Pareto principle (80% of all errors will likely be found in 20% of the code) applies to software testing.
- Testing should begin in the small and progress to the large.
- Exhaustive testing is not possible.
- To be most effective, testing should be conducted by an independent third party.

The total project is divided into six modules. Each module is designed & tested individually.

GOOD TEST ATTRIBUTES:

- A good test has a high probability of finding an error.
- A good test is not redundant.
- A good test should be best of breed.
- A good test should not be too simple or too complex.

TEST CASES-DESIGN STRATEGIES:

- Black-box or behavioral testing (knowing the specified function a product is to perform and demonstrating correct operation based solely on its specification without regard for its internal logic).
- White-box or glass-box testing (knowing the internal workings of a product, tests are performed to check the workings of all independent logic paths).

STRATEGIC APPROACH TO SOFTWARE TESTING:

- Testing begins at the component level and works outward toward the integration of the entire computer-based system.
- Different testing techniques are appropriate at different points in time.
- The developer of the software conducts testing and may be assisted by independent test groups for large projects.
- The role of the independent tester is to remove the conflict of interest inherent when the builder is testing his or her own product.
- Testing and debugging are different activities.
- Debugging must be accommodated in any testing strategy.

STRATEGIC TESTING ISSUES:

- Specific product requirements in a quantifiable manner before testing starts.
- Specific testing objectives explicitly.
- Identify the user classes of the software and develop a profile for each.
- Develop a test plan that emphasizes rapid cycle testing.
- Build robust software that is designed to test itself (e.g. uses ant bugging).
- Use effective formal reviews as a filter prior to testing.
- Conduct formal technical reviews to assess the test strategy and test cases.

UNIT TESTING:

- Black box and white box testing.
- Module interfaces are tested for proper information flow.
- Local data are examined to ensure that integrity is maintained.
- Boundary conditions are tested.
- Basis path testing should be used.
- All error handling paths should be tested.
- Drivers and/or stubs need to be developed to test incomplete software.

INTEGRATION TESTING:

When collections of components have been unit-tested, the next step is ensuring that the interfaces among the components are defined and handled properly. The process of verifying the synergy of system components against the program Design Specification called Integration testing.

Depending on design approach, one of the following integration strategies can be adopted:

- Big Bang approach
- Incremental approach
    - Top-down testing
    - Bottom-up testing
    - Sandwich testing

- **TOP-DOWN INTEGRATION TESTING:**
    1. Main control module used as a test driver and stubs are substitutes for components directly subordinate to it.
    2. Subordinate stubs are replaced one at a time with real components (following the depth-first or breadth-first approach).
    3. Tests are conducted as each component is integrated.
    4. On completion of each set of tests and other stub is replaced with a real component.
    5. Regression testing may be used to ensure that new errors not introduced.

- **BOTTOM-UP INTEGRATION TESTING:**
    1. Low level components are combined in clusters that perform a specific software function.
    2. A driver (control program) is written to coordinate test case input and output.
    3. The cluster is tested.
    4. Drivers are removed and clusters are combined moving upward in the program structure.
- **REGRESSION TESTING:**
    1. Representative sample of existing test cases is used to exercise all software functions.
    2. Additional test cases focusing software functions likely to be affected by the change.
    3. Tests case the focus on the changed software components.
- **SMOKE TESTING**:
    1. Software components already translated into code are integrated into a build.
    2. A series of tests designed to expose errors that will keep the build from performing its functions are created.
    3. The build is integrated with the other builds and the entire product is smoke tested daily (either top-down or bottom-up integration may be used).

GENERAL SOFTWARE TEST CRITERIA:

- Interface integrity (internal and external module interfaces are tested as each module or cluster is added to the software).
- Functional validity (test to uncover functional defects in the software).
- Information content (test for errors In local or global data structures).
- Performance (verify specified performance bounds are tested).

VALIDATION TESTING:

- Ensure that each function or performance characteristic conforms to its specification.
- Deviations (deficiencies) must be negotiated with the customer to establish a means for resolving the errors.

- Configuration review or audit is used to ensure that all elements of the software configuration have been properly developed, cataloged, and documentation to allow its support during its maintenance phase.

ACCEPTANCE TESTING:

- Making sure the software works correctly for intended user in his or her normal work environment.
- Alpha test (version of the complete software is tested by customer under the supervision of the developer's site).
- Beta test (version of the complete software is tested by customer at his or her own site without the developer being present).

The next step is customer's validation of the system against User Requirements Specification (URS). Customer in their working environment does this exercise of Acceptance Testing usually with assistance from the developers. Once the system is accepted, it will be installed and will be put to use.

SYSTEM TESTING:

Once the system is integrated, the overall functionality is tested against the Software Requirements Specification (SRS). Then, the other non-functional requirements like performance testing are done to ensure readiness of the system to work successfully in a customer's actual working environment. The step is called **System Testing.**

- Recovery testing (checks the system's ability to recover from failures).
- Security testing (verifies that system protection mechanism prevents improper penetration or data alternation).
- Stress testing (program is checked to see how well it deals with abnormal resource demands – quantity, frequency, or volume).
- Performance testing (designed to test the run-time performance of software, especially real-time software).

DEBUGGING:

- Debugging (removal of a defect) occurs as a consequence of successful testing.
- Some people are better at debugging than others.
- Common approaches:
    1. Brute force (memory dumps and run-time traces are examined for clues to error causes).
    2. Backtracking (source code is examined by looking backwards from symptom to potential causes of errors).

Cause elimination (uses binary partitioning to reduce the number of locations potential where errors can exist).

PROJECT TESTING

Unit testing of each module was done after successfully completing the module. Each module was tested individually before integrating them with the whole system.
After integrating each module with the system, integration testing was done in order to check if modules are working properly together.
After completing all integrations, black-box testing of the whole system was carried out to ensure the system works in a correct manner.

Black box testing of Major functions of the system

1. **Log in to the system:**

   Case 1: Invalid Username or password entered by the user.
   Output: Error message on the screen saying "Invalid credentials"

   Case 2: Valid credentials.
   Output: The user is redirected to the Dashboard page.

2. **Update Profile:**

   Case 1: username already exists.
   Output: Error message on the screen saying "Username already exists"

   Case 2: Some of required fields missing in input.
   Output: Model validation errors will be displayed to the user.

   Case 3: All input data are valid.
   Output: Profile updated successfully.

3. **View Attendance:**

   Case 1: User is not logged in.
   Output: Redirected to the login page with error message "Please login!".

   Case 2: If a user exists and has the attendance records.
   Output: All the chat history will be displayed

   Case 4: Provided username does not exist in the system.
   Output: 404 Error.

16. PROJECT DESCRIPTION

Introduction

This project aims to automate the traditional attendance system where the attendance is marked manually. It also enables an organization to maintain its records like in-time, out time, break time and attendance digitally. Digitalization of the system would also help in better visualization of the data using graphs to display the no. of employees present today, total work hours of each employee and their break time. Its added features serve as an efficient upgrade and replacement over the traditional attendance system.

Functional requirement

1.1 Manage Registration and Login

1.1.1 Register new employee

Description: Admin can register new
Input: Employee Details
Output: success message displaying the user has been created.

1.1.2 Log-In to the system

Input: User credentials
Output: If the credentials are correct, user will be redirected to the dashboard of the system
Exception Flow: If the entered credentials are incorrect then user will be redirected to the login page again displaying an error message.

1.2 Manage Attendance Details

1.2.1 Mark your attendance-in

Input: User will scan his/her face using the external web camera.
Output: system will identify the user uniquely and will mark his/her in-time to the database. The same success message will be transmitted to the user.

1.2.2 Mark your attendance-out

Input: User will scan his/her face using the external web camera.
Output: system will identify the user uniquely and will mark his/her out-time to the database. The same success message will be transmitted to the user.

1.2.3 View my attendance report

Description: Employee may often need to see his / her attendance record throughout the month or year. Using this feature one can see his / her attendance record till the date.
Input: User selection
Output: Statistical analytics of the particular employee who is currently logged into the system will be displayed.

1.2.4 View employee's attendance report

Description: This feature is for admin. Admin can monitor the availability of each employee till the date. i.e., how many employees are present today out of total employees etc. can be monitored.
Input: user selection
Output: Attendance record of each employee including how many employees are present today out of total along with the availability graph.

1.3 Manage Employee Details

1.3.1 Add photo of the employee

Description: Admin only can access this feature. Admin can add a photo of an employee during the registration process.
Input: Username of an employee
Output: Success message record has been added.
Process: System will process an image and will generate necessary system data to identify each employee uniquely.

1.3.2 Train the system

Input: user selection
Output: system will process all the available records of the employees and will generate necessary system data to identify each employee uniquely.

Other Non-Functional requirements

➢ Performance Requirements

Some performance requirements identified is listed below:

- ❖ The performance of the system should be fast and accurate.
- ❖ The system should be able to handle large amount of data. Thus, it should accommodate high details without any fault.

There are no other specific performance requirements that will affect development.

➢ Safety Requirements

As a part of the safety requirement, we prefer to keep a backup of the system generated data in any external device.

➢ Security Requirements

- ❖ Utilize certain cryptographic technique
- ❖ Keep specific log or history data sets
- ❖ Assign certain functions to different modules
- ❖ Restrict communications between some areas of the program

➢ Software Quality Attributes

- **Portability**: The system is developed for secured purpose, so it is can't be portable.
- **Availability**: This system will available only until the system on which it is install, is running.
- **Scalability:** Applicable.

➢ Business Rules

No specific business rules were taken into an account at this time.

Functionalities implemented successfully:

- Registration
- Login/Logout
- Manage user profile
- Update user profile
- View My Attendance
- View Attendance by Date
- View Attendance by Employee
- Manage Attendance
- Mark my Attendance in
- Mark my Attendance out
- Add photos
- Add new employee
- Train the system
- View Attendance record by date
- View number of employee present today
- View total number of employees

17. FEATURES

The Website provides following functionalities to the users:

- **Administrator**:
  - Login/Logout
  - Register new employees/student to the system
  - Add employees/student's photos to the training data set
  - Train the model
  - View attendance report of all employees. Attendance can be filtered by date or employee

- **Employee**:
  - Login/Logout
  - Mark his/her time-in and time-out by scanning their face
  - View attendance report of self

18. LIMITATIONS AND FUTURE EXTENSIONS

● Limitations

- Attendance can be marked if the picture of an employee is shown
- 300 images of each employee are taken for better accuracy. 300 Images per employee in a larger organization would consume a massive volume to store the images.
- The training time for our classifier takes about 20 seconds for each person. Hence for a large number of employees, it would take a very long time to train. Though training the classifier isn't something that needs to be frequently done, but it would be better if a classifier taking lesser time while maintaining the accuracy can be built.
- The current model is 99.38% Accurate

● Functionalities not implemented

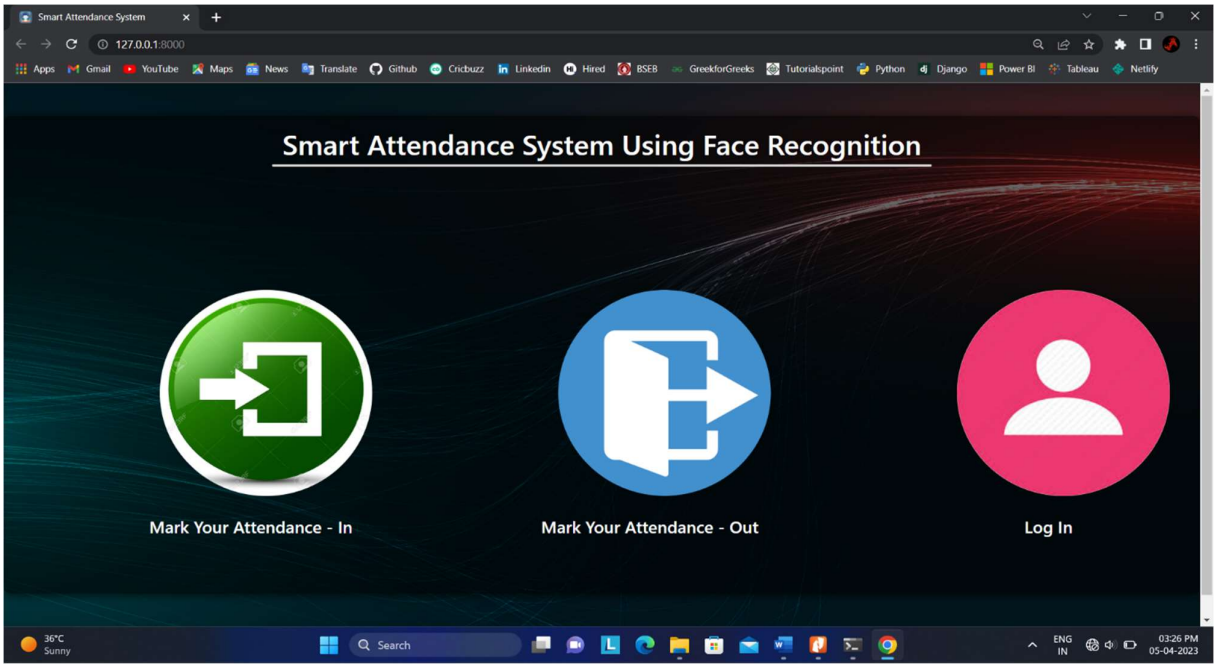
- Alert System
- Forgot Password
- Email Notifications

● Possible future extensions

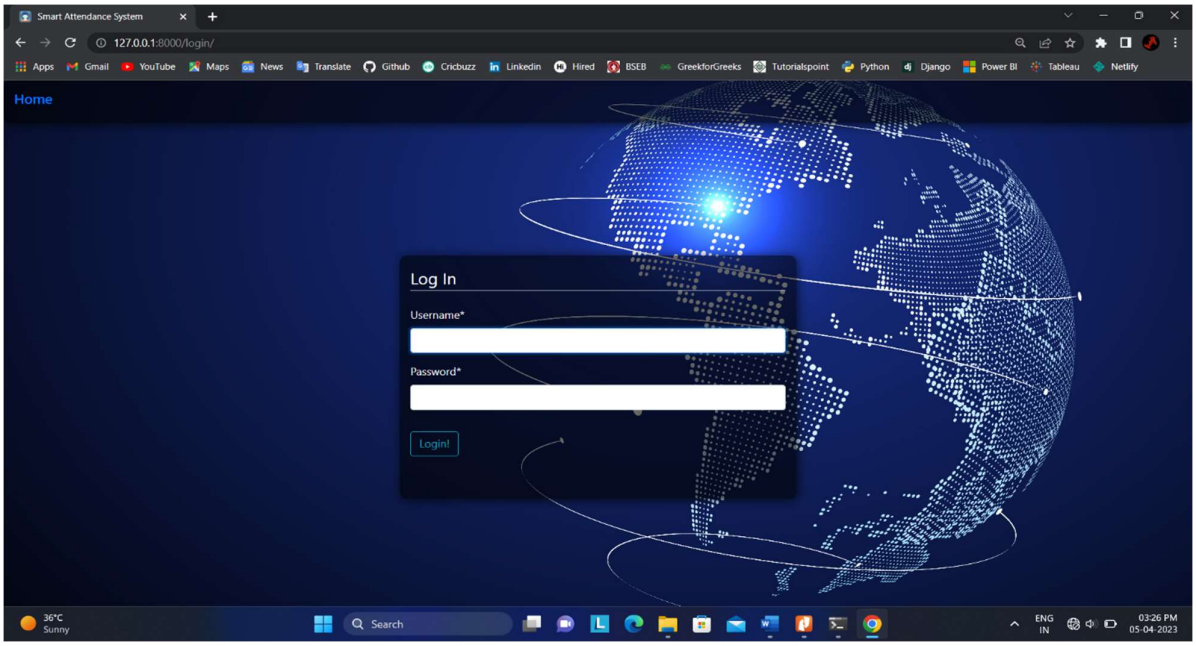
- A feature which can give intruder alert can be included in the system.
- Furthermore, the images of unknown people can be saved in an efficient manner and displayed in the system for better security.
- The number of training images can be reduced so that less storage is required. This can be done by removing duplicate images of the same person, or images with similar embeddings.
- The training time can be reduced by retraining the classifier only for the newly added images.
- A feature can be added where an employee is automatically sent a warning if his attendance or working hours are below the threshold.
- Wrongly classified images can be added to the training dataset with the correct label so as to increase the accuracy of the recognition model.

19. OUTPUT SCREEN

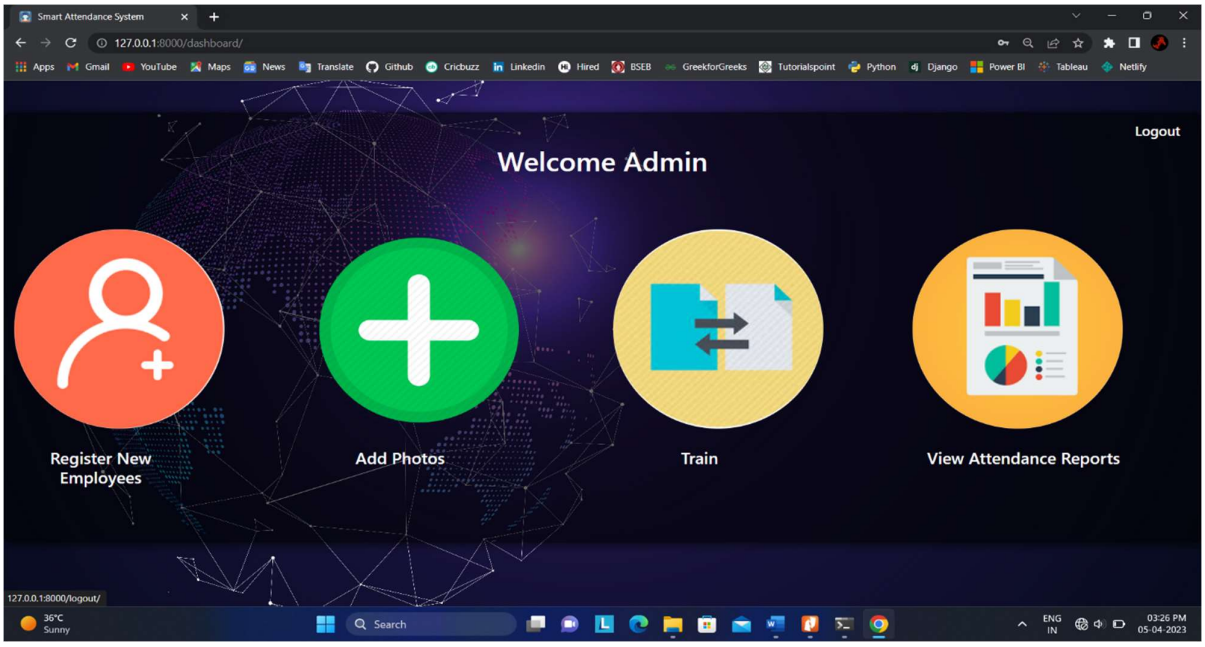
Homepage:



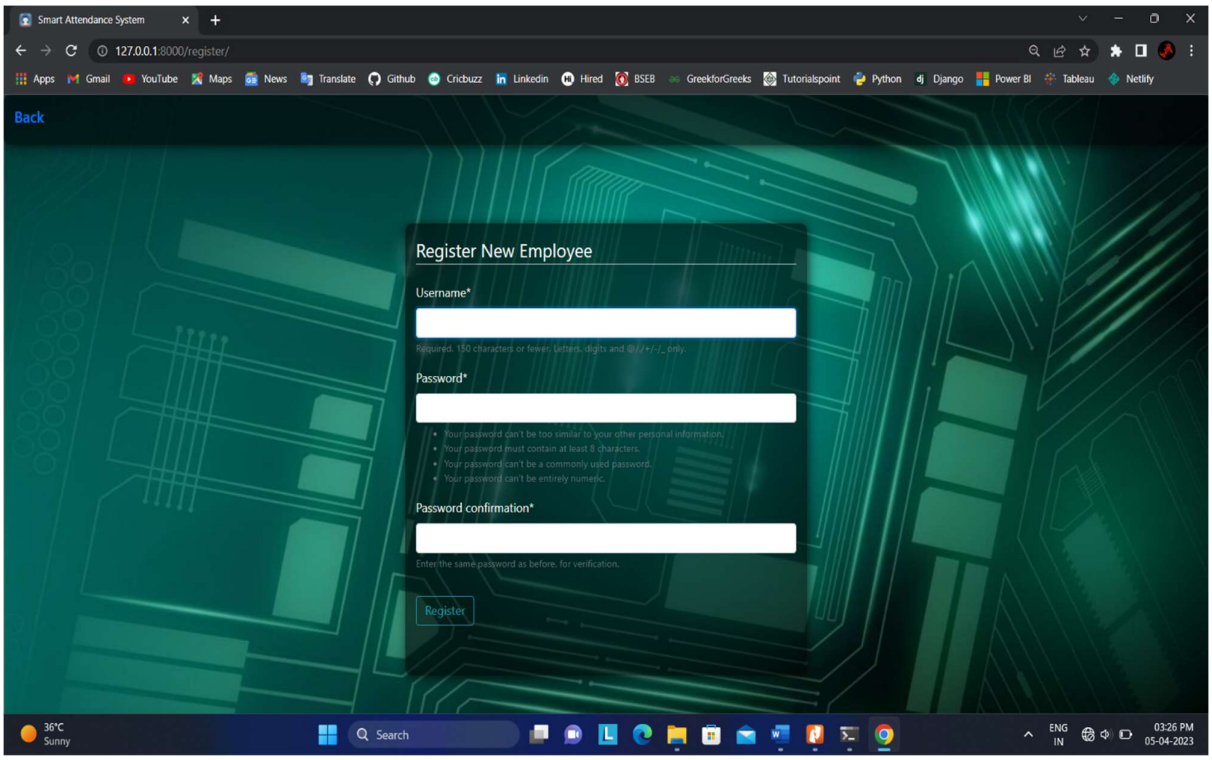
Login:



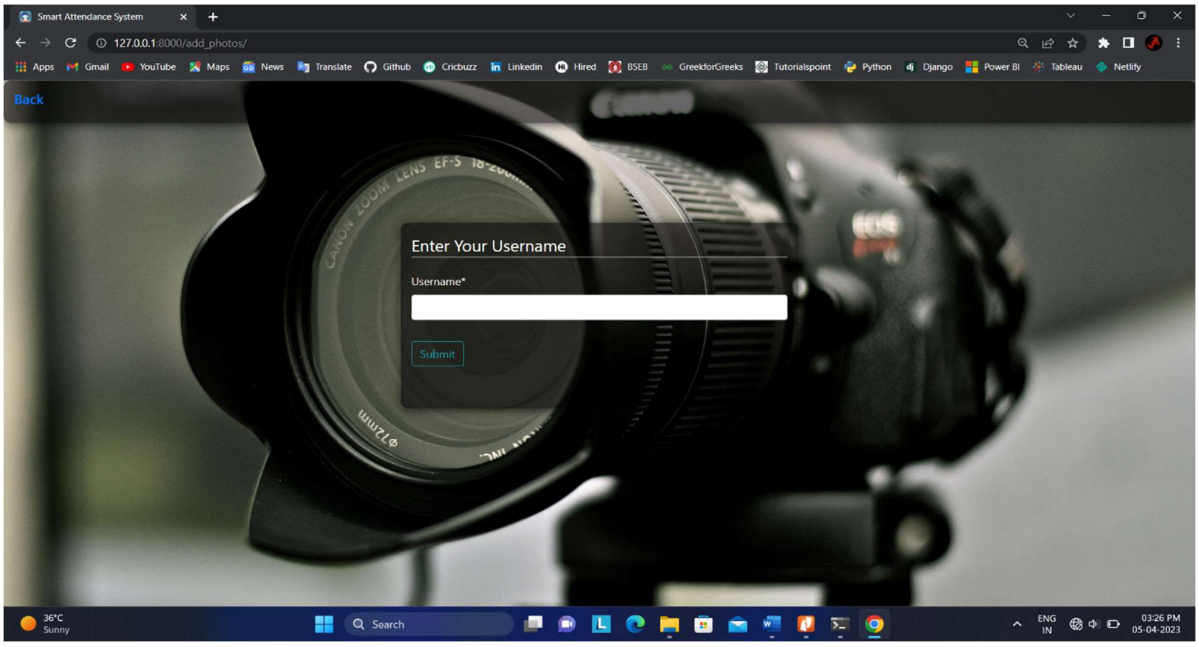
Admin Dashboard:



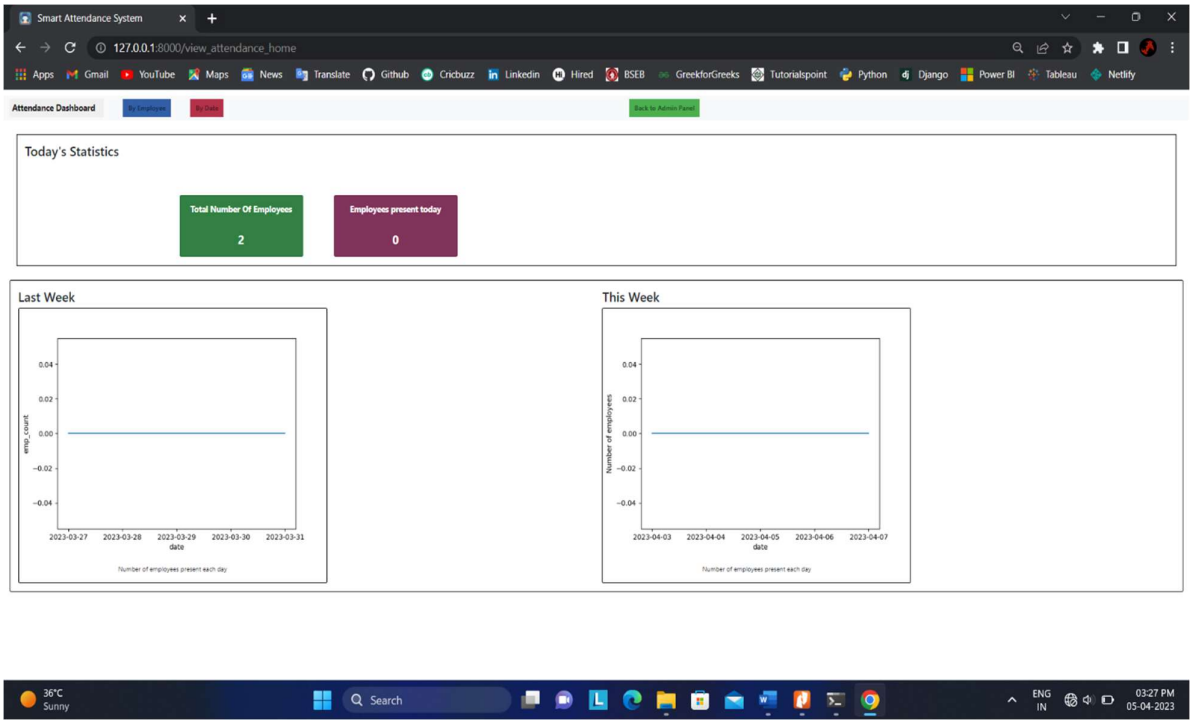
Register New Employee:



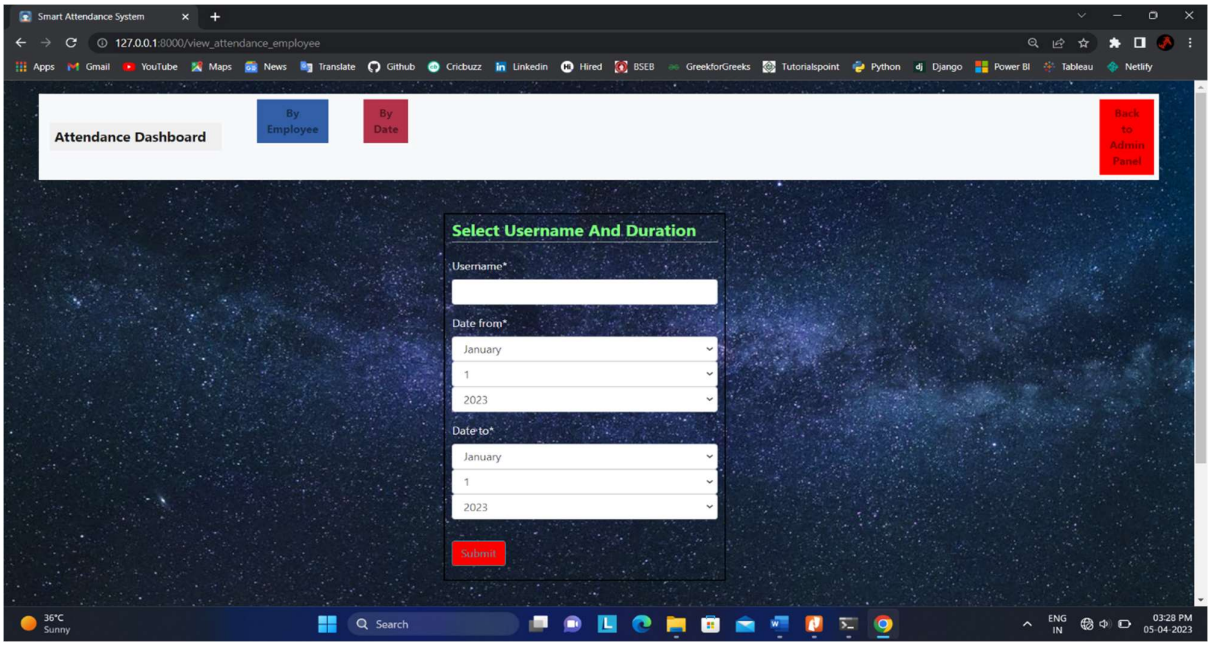
Add Photos:



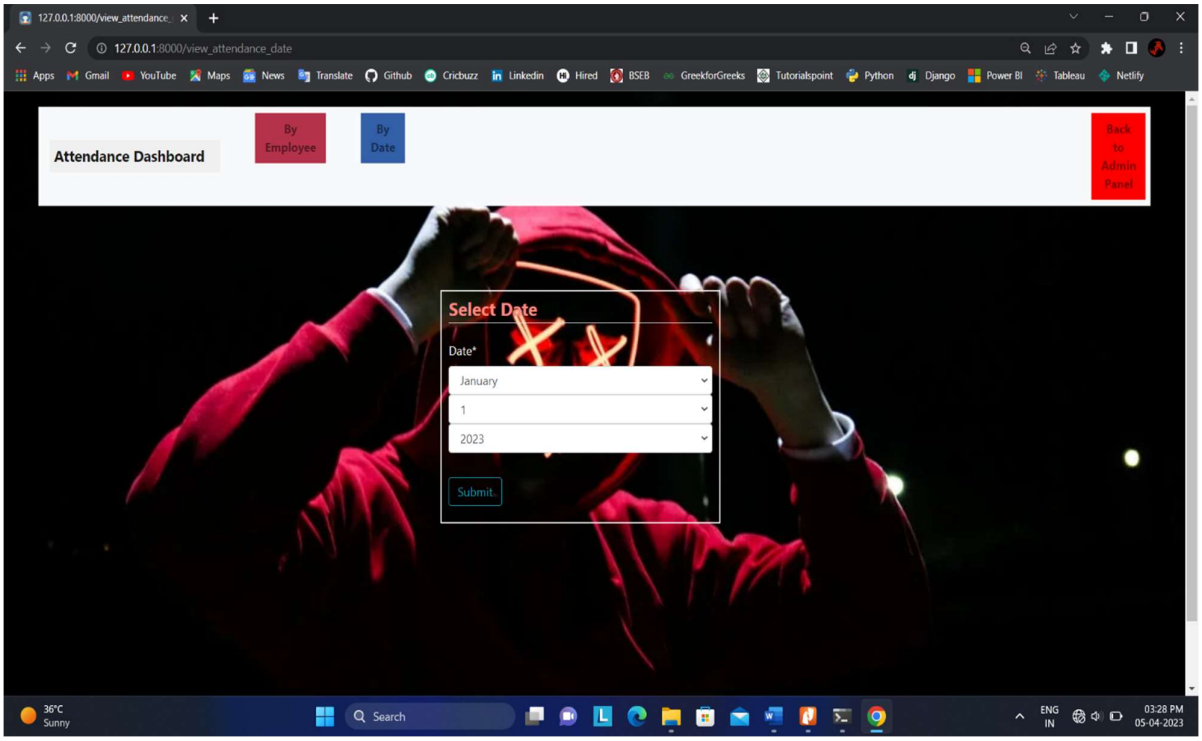
View Attendance by Administrator:



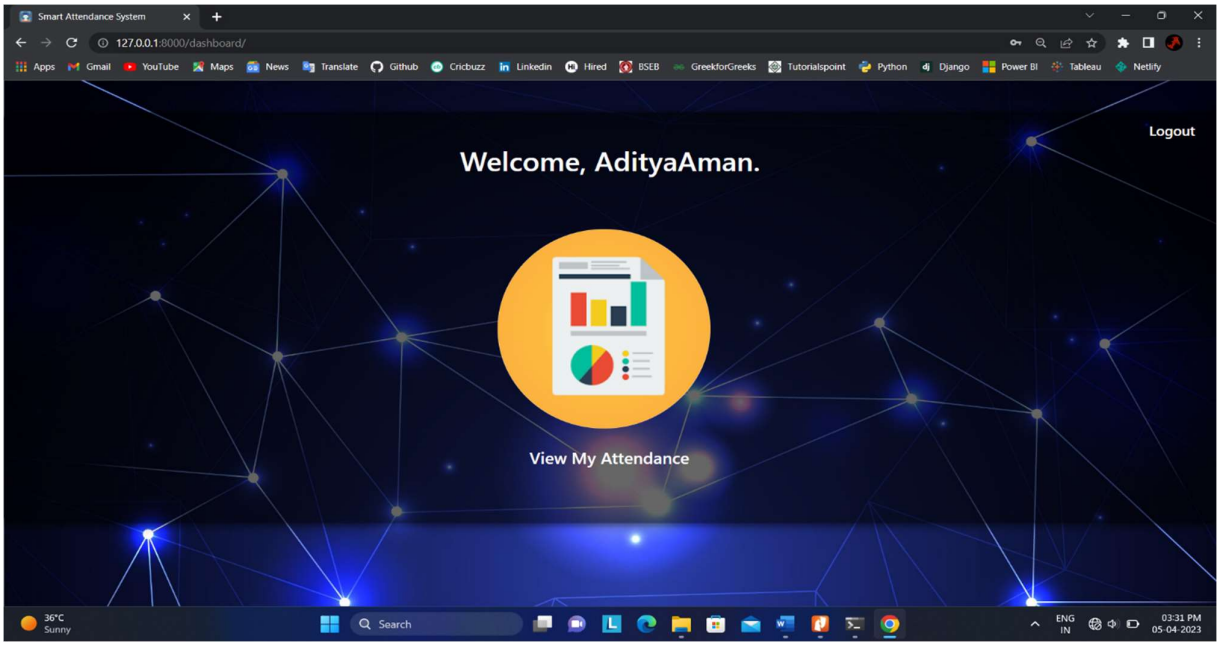
View Attendance by User Name:



View Attendance by Date:



Employee Dashboard:



20. BIBLOGRAPHY

The Web framework for perfectionists with deadlines | Django (djangoproject.com)
https://medium.com/
http://dlib.net/
https://opencv.org/
https://geeksforgeeks.org/
https://github.com/
https://www.w3schools.com/
https://python.org/
https://tutorialspoint.com/

- CSS & Bootstrap:
  https://getbootstrap.com/

- For debugging:
  https://stackoverflow.com/

- Reference Books:
  1. Building Websites with Django Build and deploy professional websites with Python programming and the Django framework by Awanish Ranjan
  2. Django for Beginners (William S. Vincent)
  3. Eloquent JavaScript_ A Modern Introduction to Programming
  4. Programming and problem solving with Python

THANK YOU