



Thakur Educational Trust's (Regd.)

THAKUR COLLEGE OF SCIENCE & COMMERCE

(NAAC Re-Accredited Grade –A, CGPA -3.10 & ISO 9001:2008 Certified)
Shyamnarayan Thakur Marg, Thakur Village, Kandivali (East), Mumbai -400 101.

Project Documentation on
Attendance Management System using Facial
Recognition using Raspberry Pi

Submitted by:

Lokesh Vyas

22645

March, 2022-2023

Under the guidance of:

Ms. Sagarika Prakash

Submitted in partial fulfilment of requirements for qualifying

B.Sc.- (IT), Semester VI Examination



Thakur Educational Trust's (Regd.)

THAKUR COLLEGE OF SCIENCE & COMMERCE

(NAAC Re-Accredited Grade –A, CGPA -3.10 & ISO 9001:2008 Certified)
Shyamnarayan Thakur Marg, Thakur Village, Kandivali (East), Mumbai -400 101.

Project Certificate

This is to certify that the project entitled **ATTENDANCE SYSTEM WITH FACIAL RECOGNITION USING RASPBERRY PI**

undertaken at the Thakur College of Science and Commerce by **Mr. Lokesh Vyas (Roll No. 22645)** in partial fulfilment of B.Sc. IT degree (Semester VI) Examination had not been submitted for any other examination and does not form part of any other course undergone by the candidates.

It is further certified that they have completed all the required phases of the project.

Signature External Examiner

Signature Internal Examiner

Signature Project Guide

Signature HOD/In-charge/Co-ordinator

ACKNOWLEDGEMENT

We have taken efforts in this project. However, it would not have been possible without the kind support and help of our Project Guide **Ms. Sagarika Prakash, and many Faculties.** We would like to extend our sincere thanks to all of them.

We are highly indebted for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

We would like to express our gratitude towards my parents & member of the teacher for their kind cooperation and encouragement which help us in the completion of this project. Last but not least, many thanks go to the head of the project, whose have invested his full effort in guiding the team to achieving the goal.

We would like to express our special gratitude and thanks to all the above-mentioned people for giving us such attention and time. Our thanks and appreciations also go to our colleagues in developing the project and the people who have willingly helped us out with their abilities.

PREFACE

This Report has been prepared as a part of our final project. The report is prepared with the view to include all the details regarding the project we carried out.

The use of technology in education and corporate environments has drastically increased over the years. With the widespread use of digital platforms and devices, there is an increasing need for efficient and accurate methods of attendance management. Traditional methods of attendance management involve manual entry, which is time-consuming and prone to errors. This led to the development of automated attendance management systems using various technologies.

The "Attendance Management System Using Facial Recognition Using Raspberry Pi" project aims to provide a solution to the challenges of attendance management in educational and corporate settings. This project utilizes facial recognition technology to automate the attendance marking process, eliminating the need for manual entry.

This project is designed to be a cost-effective, efficient, and user-friendly solution to attendance management. It is also environmentally friendly as it eliminates the need for paper-based attendance sheets. The development of this project requires the use of readily available and affordable resources such as Raspberry Pi, a low-cost computer, and a camera module.

This project is intended for educational institutions, corporate organizations, and any other organizations that require an efficient attendance management system. It is also intended for developers and individuals interested in learning more about facial recognition technology and its applications.

INDEX

Sr. No	Topic	Pg No
1	Introduction	7 - 11
1.1	Objective and Scope	7
1.2	Theoretical Background	6
1.3	Problem Definition	8
1.4	Software Requirement Specification (SRS)	8
1.5	Feasibility Study	9
1.6	Details of Hardware and Software	11
2	System Analysis and Design	12 - 43
2.1	Prototyping Model	12
2.2	Detailed Life Cycle	14
2.3	Data Flow Diagram	16
2.4	Flow Chart	21
2.5	Entity- Relationship Diagram	24
2.6	Data Structure Diagram	25
2.7	Use-Case Diagram	26
2.8	Activity Diagram	32
2.9	Component Diagram	35
2.10	Table Design	36
2.11	Deployment Diagram	38
2.12	Circuit Diagram	39
2.13	Component Level Description and Specification	40
2.14	Block Diagram	41
2.15	State Transition	42
2.16	System Architecture Diagram	43
3	System Planning	44 - 45
3.1	Gantt Chart	44
3.1.1	Process involved	45

3.1.2	Methodology adopted	
3.2	Activity Chart Diagram	
4	System Implementation	46- 51
5	Cost and Benefit Analysis and Software Parameter Estimation	52 – 55
6	System Testing	56 – 60
6.1	Methodologies Used	
6.2	Test Report	
7	System Maintenance and Evaluation	61-62
8	User Manual	64-69
9	Conclusion and Scope for future work	70-72
9.1	Conclusion	
9.2	Scope for future work	
10	Reference	73-74

List of Figures

Fig. 2.1	Prototyping Model	12
Fig. 2.2	Detailed Life Cycle	15
Fig. 2.3	Context Diagram (Level 0)	19
Fig. 2.4	Admin Level 1 DFD	20
Fig. 2.5	Faculty Level 1 DFD	21
Fig. 2.6	Student Level 1 DFD	22
Fig. 2.7	Flow Chart (Raspberry Pi)	23
Fig. 2.8	Flow Chart (Website)	24
Fig. 2.9	Entity- Relationship Diagram	25
Fig. 2.10	Class Diagram	26
Fig. 2.11	Admin Login Use-Case	27
Fig. 2.12	Student Login Use-Case	28

Fig. 2.13	Faculty Login Use-Case	29
Fig. 2.14	Admin Function Use-Case	30
Fig. 2.15	Faculty Function Use-Case	31
Fig. 2.16	Student Function Use-Case	32
Fig. 2.17	Admin Activity	33
Fig. 2.18	Faculty Activity	34
Fig. 2.19	Student Activity	35
Fig. 2.20	Component Diagram	36
Fig. 2.21	Deployment Diagram	37
Fig. 2.22	Circuit Diagram	38
Fig. 2.23	Raspberry Pi 3 Board	39
Fig. 2.24	USB Camera	40
Fig. 2.25	Block Diagram	41
Fig. 2.26	Admin and Faculty State Transition	42
Fig. 2.27	Student State Transition	42
Fig. 2.28	System Architecture Diagram	43
Fig. 3.1	Gantt Chart	44
Fig. 3.2	Activity Chart Diagram	45

List of Tables

Table 1	Hardware Details	11
Table 2	Software Details	11
Table 3	Admin Table	36
Table 4	Faculty Table	36
Table 5	Student Table	37
Table 6	Attendance Table	37
Table 7	Unadjusted Function Point (UFP)	46
Table 8	Complexity Adjustment Factor (CAF)	47
Table 9	COCOMO model	55

1. Introduction

“Automation is Solving the Problem once and then putting it on Autopilot.”

- Michael Hyatt.

Attendance is an important part of daily classroom evaluation. At the beginning and ending of class, it is usually checked by the teacher, but it may appear that a teacher may miss someone or some students answer multiple times. Face recognition-based attendance system is a problem of recognizing face for taking attendance by using face recognition technology based on high-definition monitor video and other information technology.

1.1 Objective and Scope:

The objective of this process is to create a software that can autonomously mark the attendance and store them in a database, this technology aims to upgrade the traditional method of marking attendance. The current methods of attendance monitoring are mainly either by speech e.g.: Roll Call or smart cards both these systems are extremely vulnerable to proxies. Our Project has the potential to eliminate this as it collects the attendance data by the means of facial scanning. Unlike the current methods the entries by facial recognition are extremely difficult to fake adding a layer of security to the process. The machine mainly comprises of two main components namely the camera module and the brains in this project we have decided to use raspberry pi as the processor because of its compact size, ready and cheap availability and the pi camera module, the language used is python. The Goal of our project is simple, to create an attendance monitoring system that uses facial recognition to record and mark attendance in the database that should be accessible remotely. As the system uses facial recognition it is much more secure and accurate in comparison to the current methods and virtually does not consume any time as it can work autonomously. As the system is a type of calm technology it can also be used for security and for entering places such as classrooms as it can segregate authorized personnel from the unauthorized ones and send an alert to the admin should an unauthorized person enter the class.

1.2 Theoretical Background:

Many organizations, companies and institutions are taking periodic attendance using RFID methods, Biometric Fingerprint method and Registers. These methods generally take more time for calculation. RFID uses electromagnetic fields to automatically identify and track tags attached to persons. Biometric fingerprint identification systems employ fingerprint as a unique

identity. It is one of the most accurate systems running effectively today. But recognition of an individual fingerprint from a set of enrolled fingerprints is a difficult process. The fingerprint system does not reveal any information regarding the original fingerprint. This may have been proved to be false as many algorithms reveal that a fingerprint can be reconstructed with minute templates. Iris Recognition is another type of implementation where the iris of people is scanned, stored and then retrieved for the comparison and attendance is managed automatically in the server. But there is difficulty in capturing iris of the students or employees and hence a fast implementation of face recognition with decreased illumination effect can be used.

1.3 Problem Definition:

The current systems that are used for updating attendance automatically are usually RFID based, Biometric based. Usually, the manual method of taking attendance is difficult and a time-consuming process. Hence it is important to construct an efficient method for managing attendance automatically. Other than being time consuming these age-old methods can be inaccurate at times and can easily be used to fake attendance.

1.4 Software Requirements Specifications (SRS):

There are two types of requirements from a software application: Functional requirements and strategic or Non-Functional requirements.

1.4.1 Functional Requirements:

‘Functional Requirements’ of a software application refers to what functionality the software should include. Functional requirement includes but is not limited to calculations, Data processing, data manipulation and storage and such that describes or creates a structure describing what the software must accomplish.

The following are the functional requirements for “*Attendance System Using Face Recognition Using Raspberry Pi*”:

Recognizing faces marking attendance –

The camera acts as a motion detector in passive mode upon sensing motion it enters active mode and starts searching for faces with the help of knowledge from its training data set. Upon detecting the faces of the user it clicks multiple pictures and runs it through the data in the database with the help of the algorithms it compares the images pixel by pixel till it finds the picture with the highest match rate, then the system matches the face to its file and marks it the user as present. If the user is moving or for some factor the camera is unable to get a viable picture, it will alert the user regarding it through a signal, similarly the system signals an acknowledgement to the user upon successful detection. The Users whose faces were not detected during the attendance time period are marked as absent.

Data manipulation –

Data manipulation consists of changing data in any way throughout the software, especially in the database. Assuming the owner wishes to add a new employee or remove an employee for whatever reason, they can simply access the database and update it by removing or adding their files. They can also mark or unmark attendances. As this is an extremely powerful feature that can be used in many ways it is reserved only to the admin and faculty level users. An admin user has access to every module and authorization to edit it.

View attendance –

The main feature of *“Attendance System Using Face Recognition Using Raspberry Pi”* is to be able to view attendance. The access to view attendance is given to every user however there are a few restrictions, considering an example of a university, the student user can only view their own attendance, the faculty level users can view their own attendance as well as the attendance for all students and edit student attendance, the admin level user can View attendance of all personnel and edit it.

Account Creation/Registration –

In order to access the system, the system needs to be able to authenticate and authorize the user, by registering themselves they create a profile for themselves that the system uses to authenticate them. A profile must be created in order to access the system and hence is the primary and basic act the user must do. The user can register themselves through the registration page where the system asks for various info about the user such as name, uid, roll no, etc. and asks to create a password and username to be used later for login. The user can use

any device for this process, after filling in the details the user will be taken to a facial registration page, over here pictures of the user will be clicked and uploaded to the database for facial recognition.

Account login and maintenance –

In order to use the system, any new user will have to Log themselves in. This module will be available to everyone as it is an essential module for the functioning of the software. During login the user will have to input their key information such as roll number, employment id etc. and password. The ‘Login’ screen is, for obvious reasons, the first screen to be displayed after selecting respective module when the software is invoked. This would accept the User-ID and Password, and if they are confirmed to be correct, opens the landing page for the user depending on his account type. The option for registration is also available on this page. However, the administrators have a bigger role to play here. They can directly create and add accounts as well as access any account.

1.4.2 Non- Functional Requirements:

Also known as strategic requirements, the non-functional requirements consist of criteria to judge the system operation and usability rather than the specific functions or features it consists of. It is also known as Qualities of the system and defines system attributes such as security, reliability, performance, maintainability, scalability.

Extensibility –

“Attendance System Using Face Recognition Using Raspberry Pi” is a system that has Excellent future scope. It can be integrated into all sorts of work environments for functions such as tracking attendance of employees, students etc. To support it, it should allow integration of new features without affecting the existing system.

Maintainability –

Given the scale of the system which may exponentially increase in the near future, it is extremely important for the *“Attendance System Using Face Recognition Using Raspberry Pi”* to be maintainable.

Security –

As “*Attendance System Using Face Recognition Using Raspberry Pi*” would be storing confidential data, it is necessary for the system data to be secure. Several methods such as authentication and encryption issuing session id’s can be used for this purpose.

Reliability –

Reliability refers to the Mean Time Between Failures (MTBF). This is the predicted elapsed time between inherent failures of a system during operation. This should be an acceptable value. The system must perform without failure in 95 percent of use cases during a month

Usability –

“*Attendance System Using Face Recognition Using Raspberry Pi*” must be user friendly to maximize adaptability and ease the transition from other methods.

Scalability –

The performance of “*Attendance System Using Face Recognition Using Raspberry Pi*” should be consistent irrespective of the load on the system, it should work efficiently and accurately across devices to ensure that it gives equal and independent performance on any device.

1.5 Feasibility study:

In the case of the attendance management system using facial recognition, a feasibility study would be necessary to determine the practicality of the proposed project. The study would consider all critical aspects of the system, including the costs associated with the development and implementation of the system, potential benefits such as increased accuracy and efficiency of attendance management, and any potential issues or problems that could arise during the project's development and implementation. It would also assess the community's response to such a system and evaluate any environmental impacts. The feasibility study would help to determine the likelihood of success of the proposed project and whether it is feasible to proceed with the development and implementation of the attendance management system using facial recognition.

1.5.1 Technical:

A technical feasibility study assesses the details of how a project intends to deliver a product or service to customers. It takes into account the various technical aspects like modules, software, algorithms etc. and the hardware requirements the project utilizes. It's the logistical or tactical plan of how the project will produce, store, deliver, and track its products or services. It is assessed based on the Prepared Outline of the project, the labor and the material requirements. This system is developed in a flexible form, which covers all operation services with the help of optimized code logic to bridge the hardware and software along with Python, HTML, CSS and MySQL, for logic, UI Framework and data management respectively. The Project is deemed Technically Feasible.

1.5.2 Economical:

Once the technical feasibility and market studies are complete, it is time to determine Business Feasibility. The first purpose of this effort is to financially model the venture opportunity and achieve a break-even analysis. In other words, based upon the costs of goods sold, capital costs, and management and administration, how much revenue generated from units sold is required to break-even and over what period of time. The simple objective is to determine what level of revenue is required to satisfy the return on investment demanded by the founder and/or the investors. Economic feasibility elements include, but are not limited to: Increased agency revenue, decreased agency revenue, increased agency costs, decreased agency costs, increased revenue to other agencies and/or the general public, decreased revenue to other agencies and/or the general public, Increased costs to other agencies and/or the general public, Decreased costs to other agencies and/or the general public, Other public benefits. In essence economic feasibility study is done to understand if a project will incur a profit or loss to an organization and its extent and thus the execution of the project is dependent on the ROI (Return on Investment) estimated. This project aims at automating the task of attendance marking for organizations such as schools, universities, Companies, etc. The scope and requirement of the

project are estimated and aimed to serve a huge market. The Materials required for hardware are readily available and cheap at mass and requires minimal maintenance.

1.5.3 Resource:

Resources that are required for the *“Attendance System Using Face Recognition Using Raspberry Pi”* include:

Programming device (Laptop),

Hardware (readily available, Bulk manufactured),

Programming tools (freely available),

Programming individuals.

So, it's clear that the project has the required resource feasibility.

1.5.4 Operational:

Operational feasibility is the measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development. its success is dependent on how well the humans/ users interact with it, If the software for a new system is too difficult to use, employees may make too many errors and avoid using it. Thus, it would fail to show operational feasibility. The proposed system will be deemed operationally feasible if users can use our site with ease. There are many ways of achieving this including but not limited to user training, Designing user friendly/ intuitive UI, etc. This “Attendance System Using Face Recognition Using Raspberry Pi” was designed while keeping it’s operability and user interaction at heart, making sure that the UI is easy to navigate by users of every level ensuring excellent functionality and use of the system by the client on their end. Thus, in conclusion “Attendance System Using Face Recognition Using Raspberry Pi” satisfies all the necessary feasibility requirements and can be deemed feasible.

1.6 Details of Hardware and Software:

1.6.1 Hardware Details –

1	Hard Disk	1 TB
2	RAM	16 GB
3	Processor	Intel i5 11 th Gen

Table: 1

1.6.2 Software Details –

1	Operating System	Windows 11
2	Database System	Xampp Server
3	Front End	Visual Studio Code

Table: 2

2. System Analysis and Design

2.1 Prototyping Model:

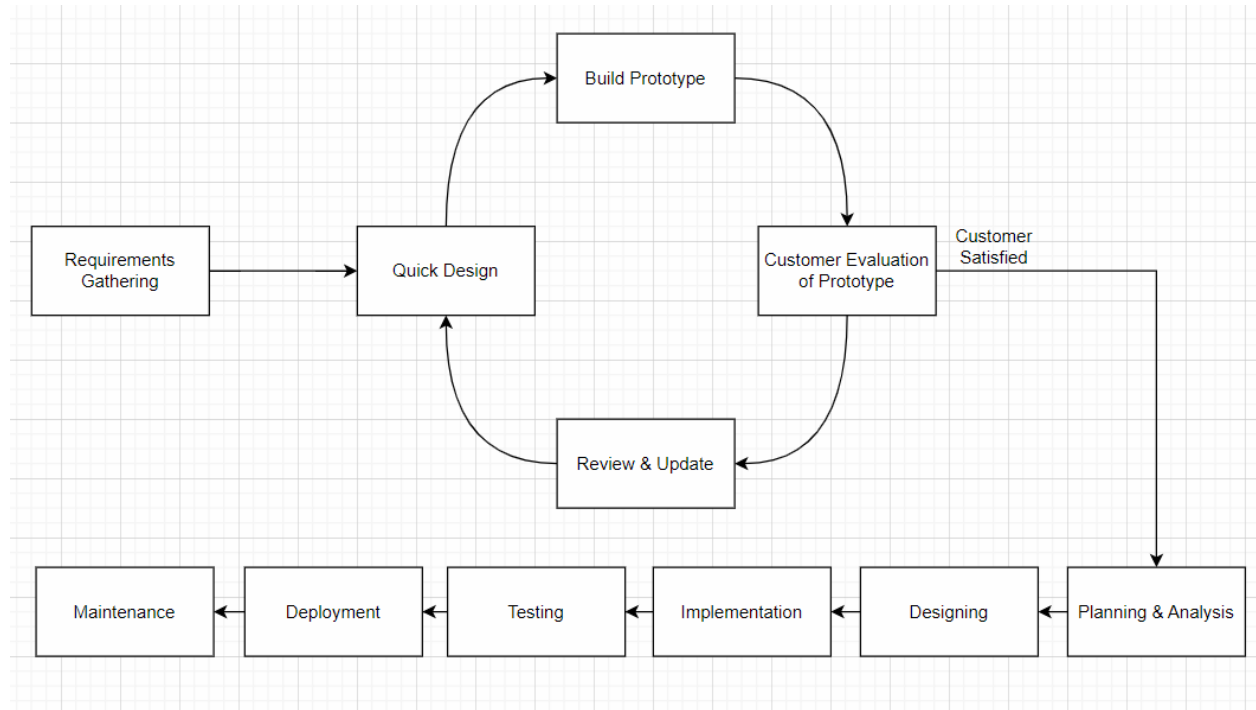


Fig. 2.1

The Prototyping model functions by creation of a small basic functioning unit of the project with respect to the software requirements and then enhancing it through various iterations until the final deployable product is achieved. In simple words the project is broken down into smaller manageable tasks or modules and these modules are developed over various prototyping iterations.

“Attendance System Using Face Recognition Using Raspberry Pi” is a typical project whose specifications and requirements align with the development structure proposed by The Prototyping model. As a result, it was the model selected. As the requirements are well defined and the team is familiar with the development platform however has not had experience with concepts such as machine learning, data cleaning, hardware implementation and deployment building the project using this model satisfied the requirements such as feasible and time constraint. The Prototyping model provides room for error and testing to the team, it allows the team to iterate upon the hardware to test the various permutations and combinations that might arise throughout the development process and allow them to test these to find the one that functions best with the system. At the end of each prototyping cycle the output was a functional

prototype of the software. After every iteration the software was tested for bugs and glitches along with functionality. As there is no client, the functional prototypes were discussed among the members and the guide to obtain feedback. Before the final prototype was deployed a complete risk analysis and testing was performed to determine the robustness of the software and determine if the software functions as desired in the intended environment. Keeping in line with The Prototyping model development methods, the project was initialized with the gathering of requirements to check if the project satisfied development criteria. Following the requirement gathering when the project satisfied all necessary criteria a quick sketch was developed. This sketch was the very first prototype of the project and all other prototypes in some way originate from this. A simple working structure was created to give some idea about functioning to the client with the help of a simulator. A basic functioning prototype was created to show the user a partial implementation of the project. This partial implementation was showed to the client to gain input to understand if it satisfied their expectations and requirements. Upon further iterations a satisfactory prototype was reached and the actual development of the project was started. The system underwent various iterations to accommodate various needs and requirements that were discovered throughout the development. For each module various prototypes were created, tested and eliminated until a satisfactory prototype that was ready to be integrated into the system was achieved.

Requirements Gathering:

Project Requirements were received and checked if they satisfied development criteria. The criteria were satisfied and the project was accepted for development.

Quick Design:

A basic understanding of the functioning of various aspects of the system needs to be created in order to understand client expectations. Keeping the line of communication clear and taking the client input a simple design of the system was created.

Build Prototype:

The design was implemented upon and a basic prototype was created with the help of simulator. This prototype was designed to be used as a based concept and be further improved upon. The prototype functioned as a partial implementation to gain approval from the user and verify that it satisfies the expectation of the client. Upon further prototyping a suitable prototype was reached that satisfied the client and was in line with the development team. The current system was built upon iterations of this base prototype.

Customer Evaluation of Prototype:

As explained earlier, keeping in line with open communication between the client and the team the partial implementations were shown and discussed with the client and iterated upon until the client demands were met.

Review & Update:

Any feedback from the client was reviewed by the team. The reviews were used to add or change functionalities in the prototypes while integrating client ideas and demands in the process. The updated prototype was then shown to the client to obtain feedback.

2.2 Detailed Life Cycle:

Our Software Development Life Cycle (SDLC) consists of the following 6 stages -

Stage 1 – Project Planning & Analysis

It is the first step of our SDLC. In this stage, we have done the research activity and also, we have gone through the related research papers which have been published in past years. Also, we have referred “Thakur College of Science and Commerce” Website for more related studies. We divided our work into small task. During the planning staged we also decided the costing as well as the requirement part. Also, we have planned how the project will complete in given schedule and budget. In this project we used prototyping model. We decided to keep total 6 phase and for these we use the work break-down structure technique to execute our project.

The 6 phases are:

- A. Planning and Requirement Analysis
- B. Designing
- C. Coding
- D. Testing
- E. Deployment
- F. Maintenance

Stage 2 – Project Designing

Project Designing represents the features of the software that helps us to develop it effectively, the architecture, the user interface, and the component level detail.

Stage 3 – Project Execution

In this phase the plans turned into the action. We allocated our work equally and started our first phase of the project according to the plan.

Stage 4 – Project Testing

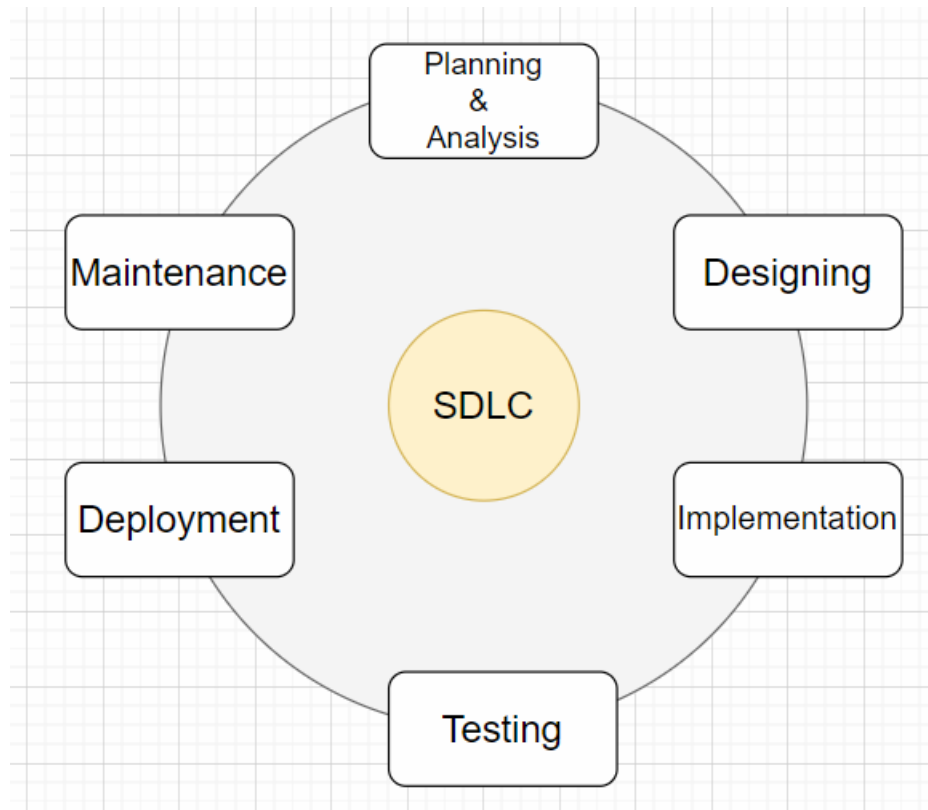
This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However, this stage refers to the testing only stage of the product where product defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.

Stage 5 – Project Deployment

Once the product is tested and ready to be deployed it is released formally to the Client/User.

Stage 6 – Project Maintenance

The purpose of Project Maintenance is to modify and update software applications after delivery to correct faults and to improve performance.

**Fig. 2.2**

2.3 Data Flow Diagram (DFD):

A Data Flow Diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled.

A Data Flow Diagram (DFD) represents graphically a flow of data within a system. It illustrates how data is input and output from the system.

In other words, DFD represents the information flow as well as where data comes from, where data goes and how it is stored.

2.3.1 Context Diagram (Level 0):

When it comes to simple data flow diagram examples, context one has the top place.

Context data flow diagram (also called Level 0 diagram) uses only one process to represent the functions of the entire system.

It does not go into details as marking all the processes.

Our system is of Attendance Management System in which we show 3 entities (Student, Faculty and Admin) Student Logs in into the system and the receives a response from the system. Faculty can login into the system and the system responds to it. And lastly, Admin can also login in to the system and get response from the system.

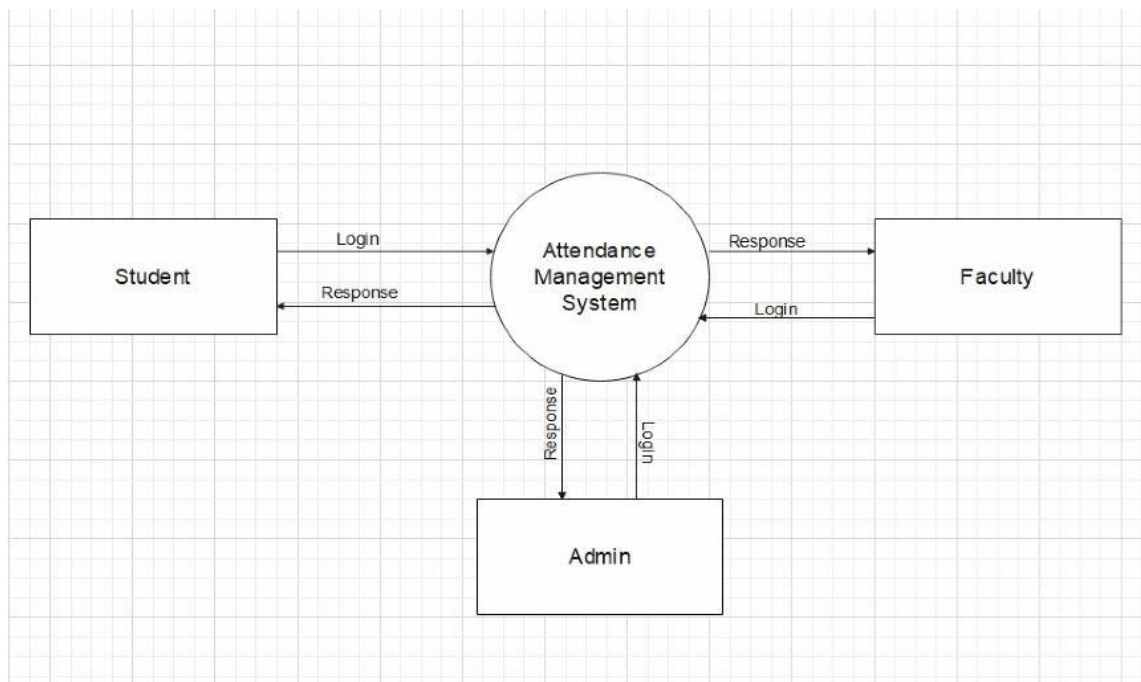


Fig. 2.3

2.3.2 Level 1 DFD:

As you saw above context DFD contains only one process and does not illustrate any data store. This is the main difference with level 1 DFD.

Level 1 DFD breaks down the main process into subprocesses that can then be seen on a deeper level. Also, level 1 DFD contains data stores that are used by the main process.

Admin Level 1 DFD

In the Admin level DFD, Admin can login into their profile and check the details related to faculty and student attendances.

Admin can add the attendance, modify the attendance and request to view the attendances of Faculty and Student.

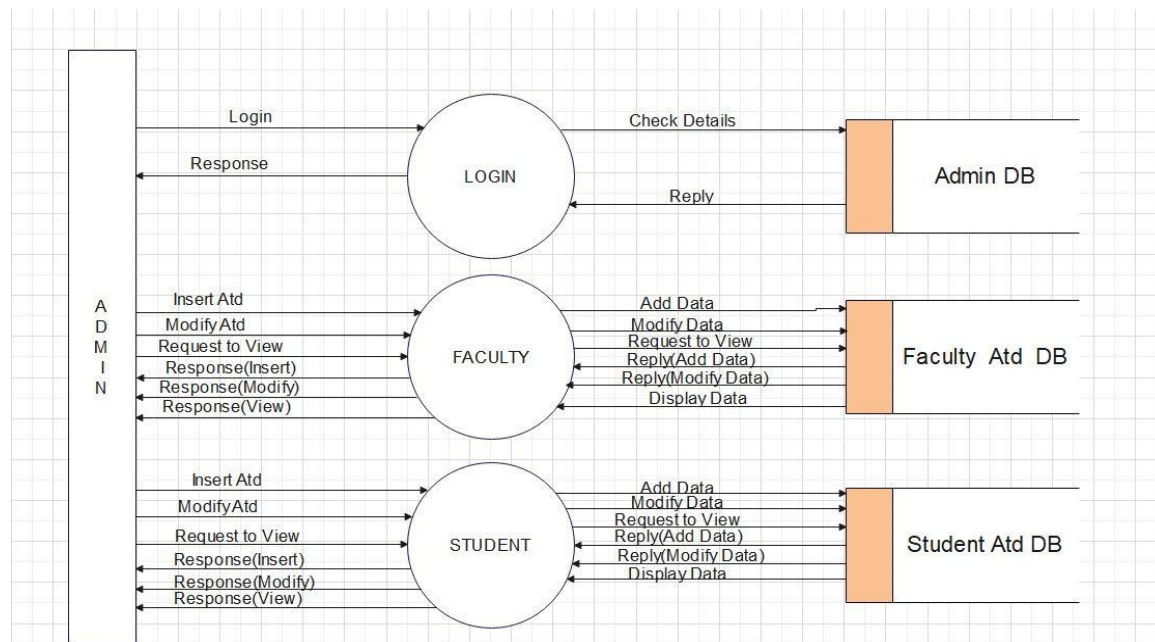


Fig. 2.4

Faculty Level 1 DFD

In the Faculty level DFD, Faculty can login in order to check the details.

They can request to view their attendance and gets response from the Faculty Attendance Database.

Faculty can also Add, Modify and View the attendance of Student from Student Attendance Database.

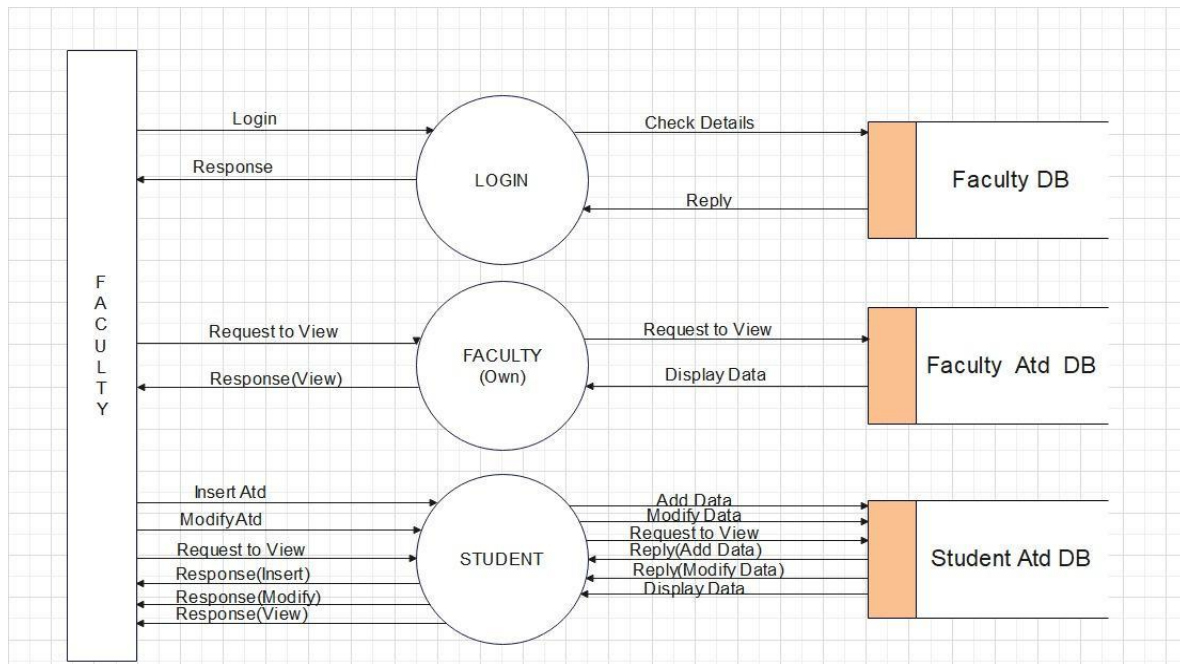


Fig. 2.5

Student Level 1 DFD

Student Level DFD, the student can login to their profiles and check their details and receives a response from Student Database.

They can request to view their attendance from the Student Attendance Database.

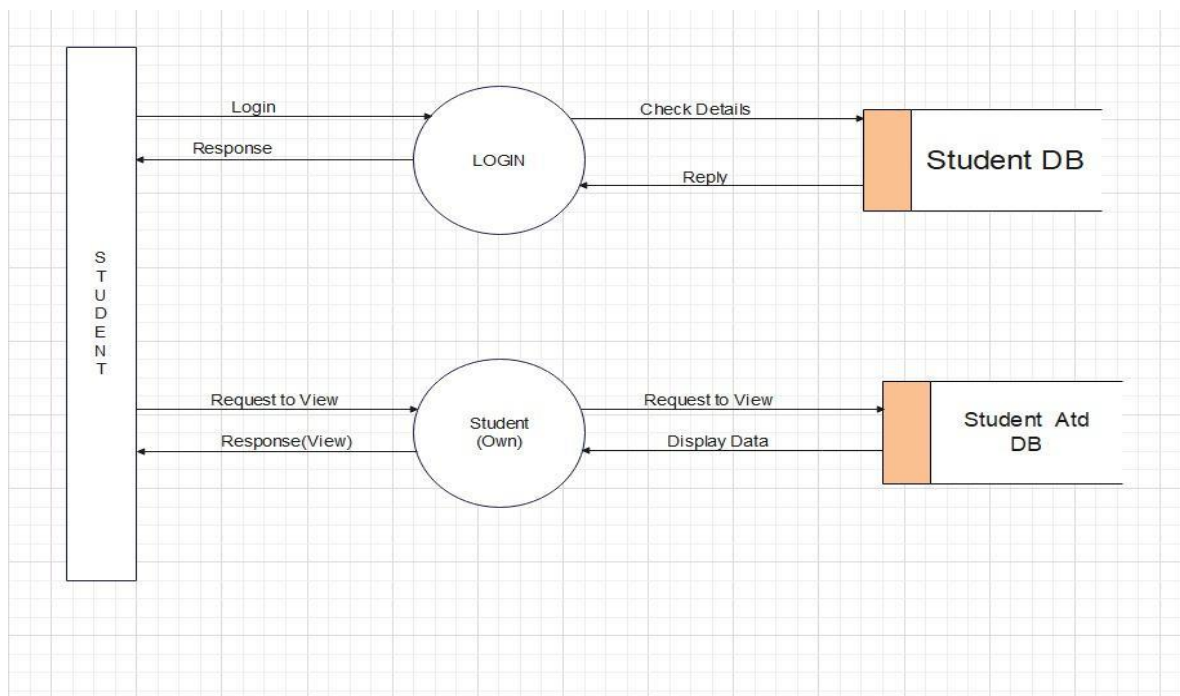


Fig. 2.6

2.4 Flow Chart:

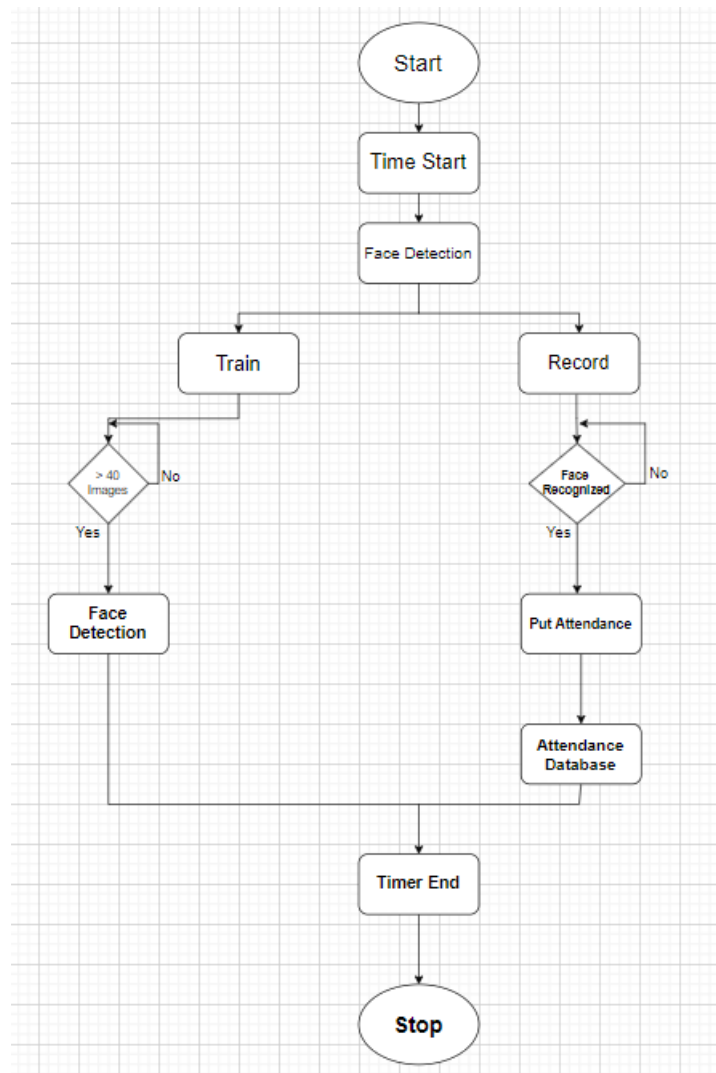
The Camera acts as a motion sensor in passive mode, upon detecting motion the camera enters active mode and starts detecting faces with the help of the algorithms. Upon triggering the camera starts a timer to click a burst of images of the detected face. These images, if viable are processed and then sent to the *Face Database* in the form of data. This data is processed in and crossed checked with the existing data in the database to find a match. Assuming the photo taken by the system doesn't match with any of the existing images in this case the attendance will not be marked autonomously and the user will require to complete the registration process correctly. This database can then be accessed to view or edit the attendance of the individuals.

The user can log into the system with their credentials which will lead them to their respective page with pre allocated clearance or level i.e., *Admin, Faculty, Student*. The ability to edit and update both the databases is reserved to an *Admin Level User*.

The *Faculty Level User* has clearance to mark or remove attendance of an individual from the attendance database and can view any attendance they wish, individual or as a whole. The student level user is limited to viewing their own attendance. Thus, automating the process of attendance marking.

Flow Chart (Raspberry Pi): -

Based on the college working hours, the camera can be configured to automatically start and stop. Whenever the camera detects any face, it will automatically start recognition process and will try to match the coordinates of the captured image from the dataset. If the face coordinates match, the attendance will be marked of that particular student. And if did not match the attendance will be marked absent.



Flow Chart (Website): -

The flow chart starts with 3 entities (Admin, Faculty, Student) If they Login successfully they can go ahead otherwise if it fails, they have to go back and login again.

The admin entity once logged in successfully can do the following: Give Attendance to Student and Faculty. Modify their Attendances (insert, delete) and View their Attendances.

The faculty entity once logged in successfully can do the following: Give Attendance to Student. Modify their Attendances (insert, delete) and view their own (faculty) attendance as well as view student attendance.

The student entity once logged in successfully can do the following: View their own Attendance (Student).

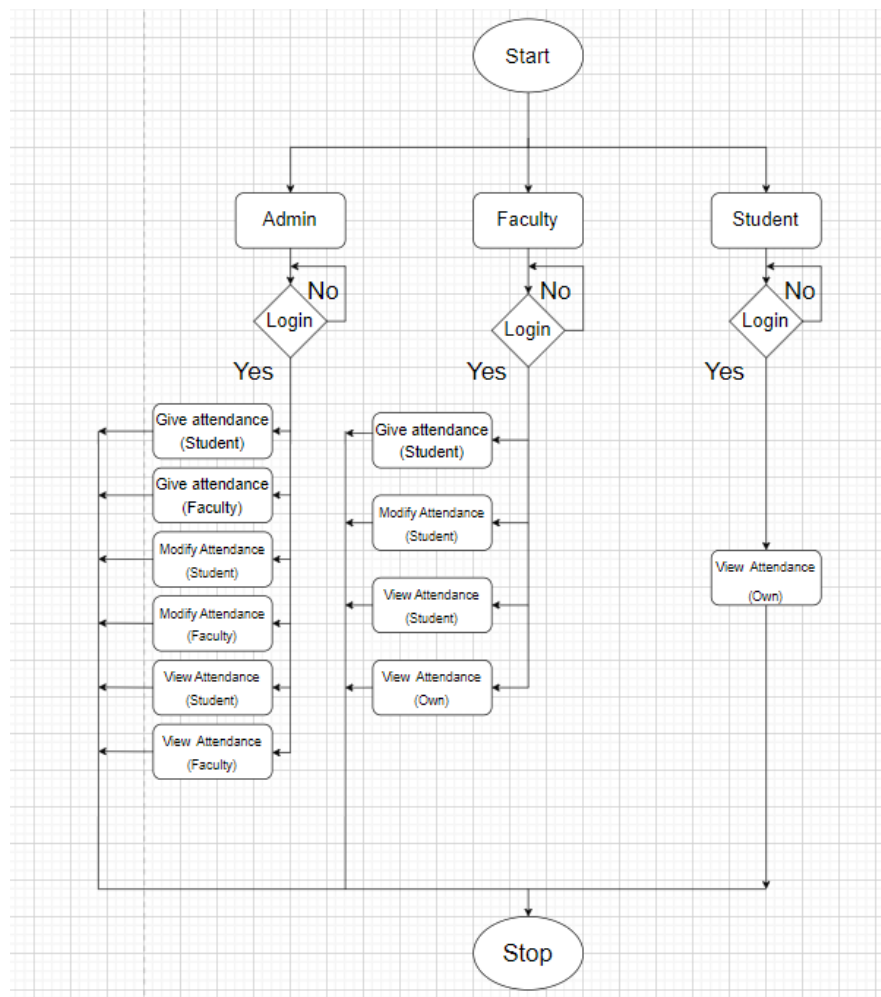


Fig. 2.8

2.5 Entity Relationship Diagram:

An Entity–Relationship model describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types and specifies relationships that can exist between entities.

Entities represent data components within a database that could be real or abstract, so long as their data is stored in the database.

Entities in our ER diagram are: Admin, Faculty, Student, Database and Raspberry Pi.

An **attribute** is a description of the properties of an entity or relationship.

Entity Admin has 5 attributes such as admin name (adm_name), student attendance(std_atd), student ID (std_id), faculty attendance (f_atd) and faculty ID (f_id). Entity Admin manages Faculty and Database entities.

Entity Faculty has 6 attributes such as faculty ID (f_id), faculty name (f_name), Student ID (std_id), student attendance (std_atd), faculty attendance (f_atd) and faculty subject (f_subject). Entity Faculty also manages Database(attendance).

Entity Student has 5 attributes such as student name (std_name), student ID (std_id), student email (std_email), student subject (std_subject) and student class (std_class). Entity student has entity Database.

Entity Database has 3 attributes such as student attendance (std_atd), faculty attendance (f_atd) and attendance ID (atd_id). Database entities send data to Raspberry Pi entity.

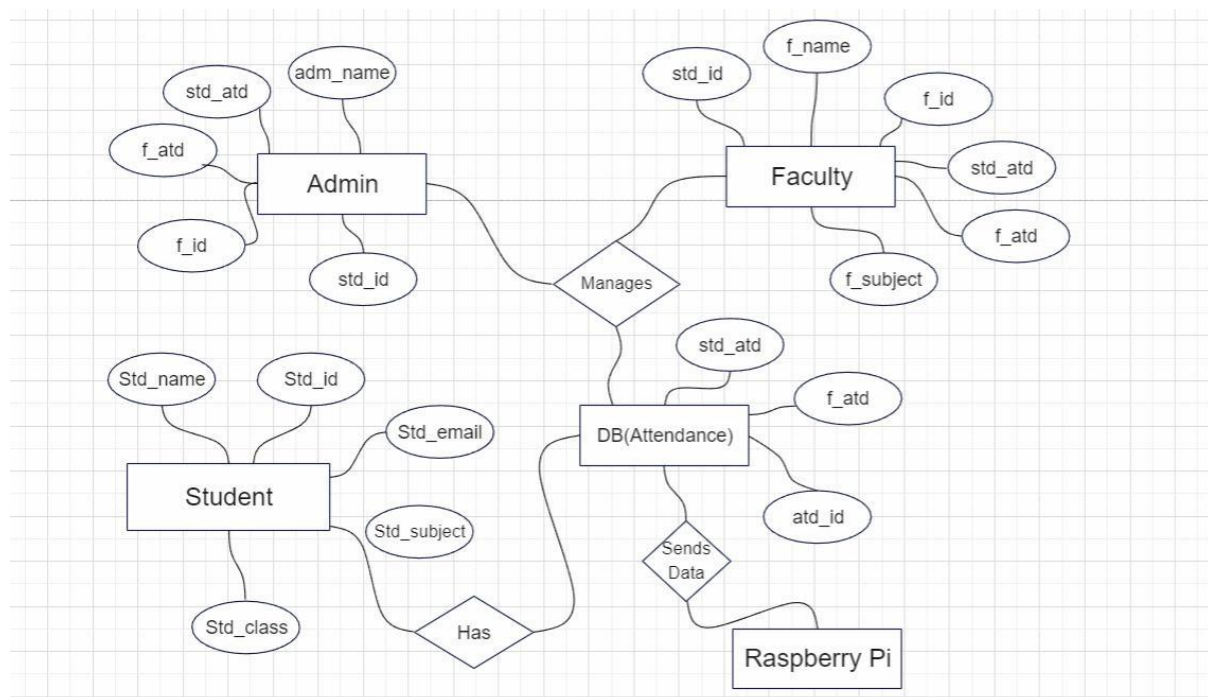


Fig. 2.9

2.6 Data Structure Diagram:

Data Structure are one of the most useful types of diagrams in UML as they clearly map out the structure of a particular system by modelling its classes, attributes, operations, and relationships between objects. One of the more popular types in UML is the class diagram. Popular among software engineers to document software architecture, class diagrams are a type

of structure diagram because they describe what must be present in the system being modelled. No matter your level of familiarity with UML or class diagrams, our UML software is designed to be simple and easy to use.

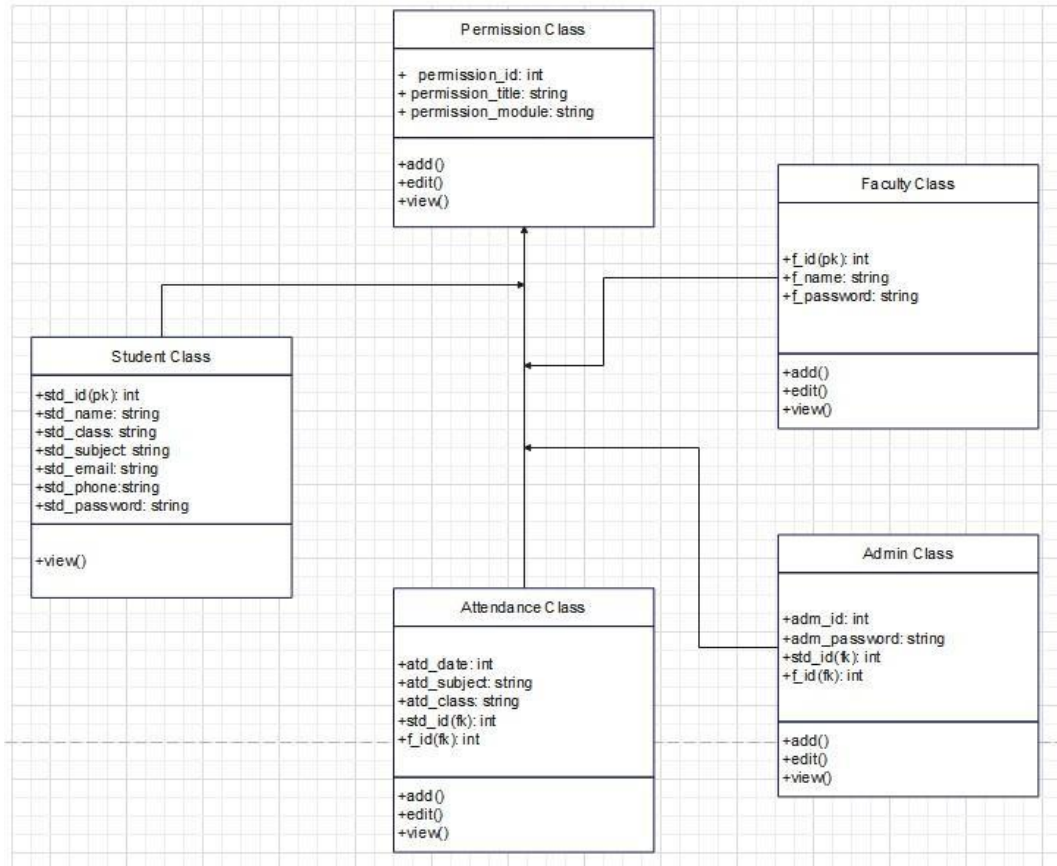


Fig. 2.10

2.7 Use-Case Diagrams:

A use case diagram shows various use cases and different types of users the system has and diagram depictions of how the user interacts with the system. It takes into account every possible interaction of developer intended features of the system and the user. Uses cases are used as a software modelling technique that helps developers determine which features to implement, and determine how to gracefully resolve errors.

2.7.1 Admin Login Use Case:

In the Admin login use case, Admin enters their credentials (Username and Password). The system verifies the credential from the database, if the credentials are incorrect, it will show a failure message. And when the credentials are correct it will show a success message.

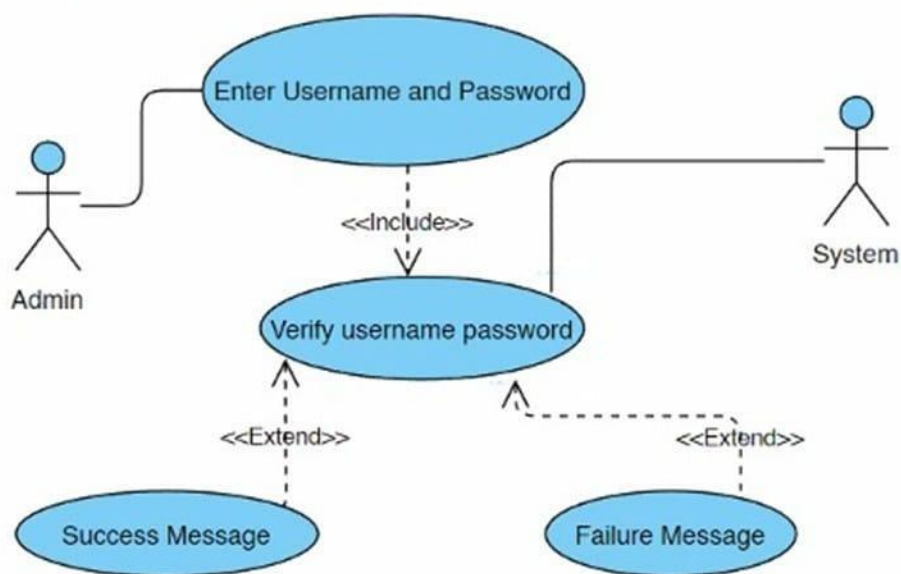


Fig. 2.11

2.7.2 Student Login Use Case:

In the Student Login Use Case, Students enter their credentials (Username and Password). The system verifies the credential from the database, if the credentials are incorrect, it will show a failure message. And when the credentials are correct it will show a success message.

Student Login Use Case

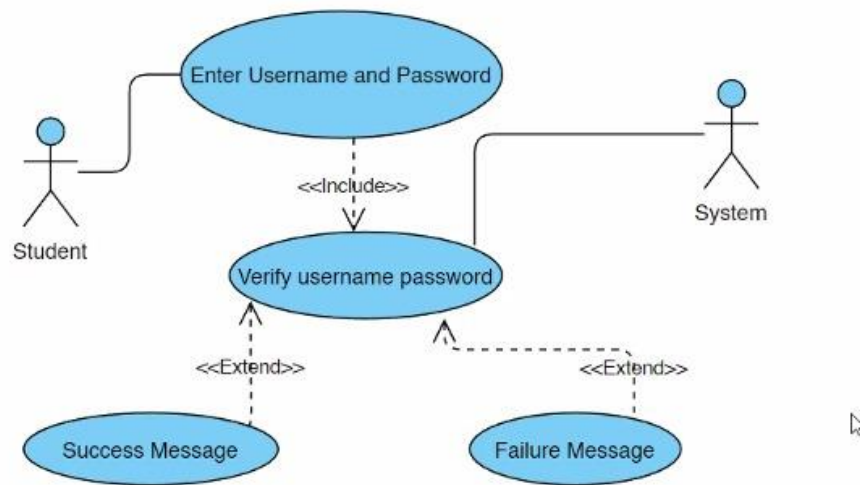


Fig. 2.12

2.7.3 Faculty Login Use Case:

In the Faculty Login Use Case, Faculty enters their credentials (Username and Password). The system verifies the credential from the database, if the credentials are incorrect, it will show a failure message. And when the credentials are correct it will show a success message.

Faculty Login Use Case

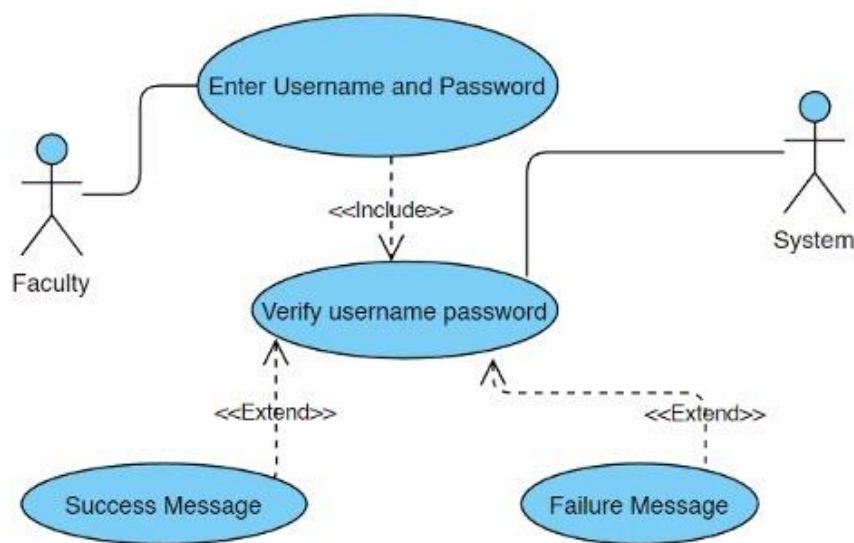
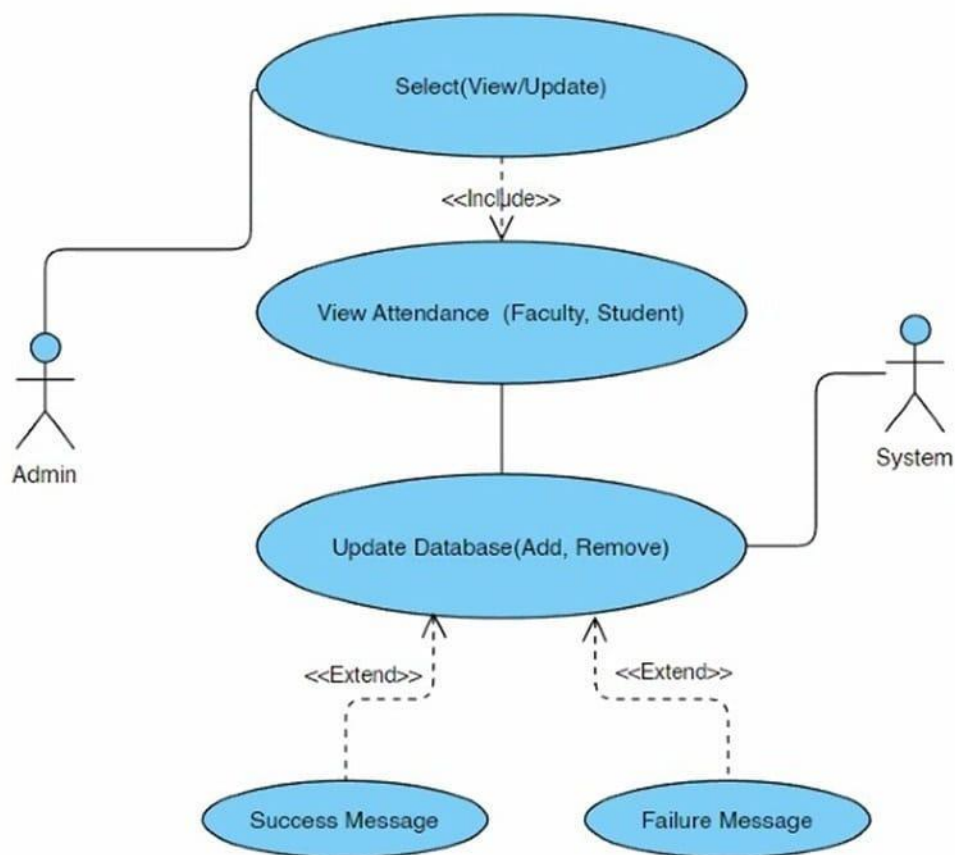


Fig. 2.13

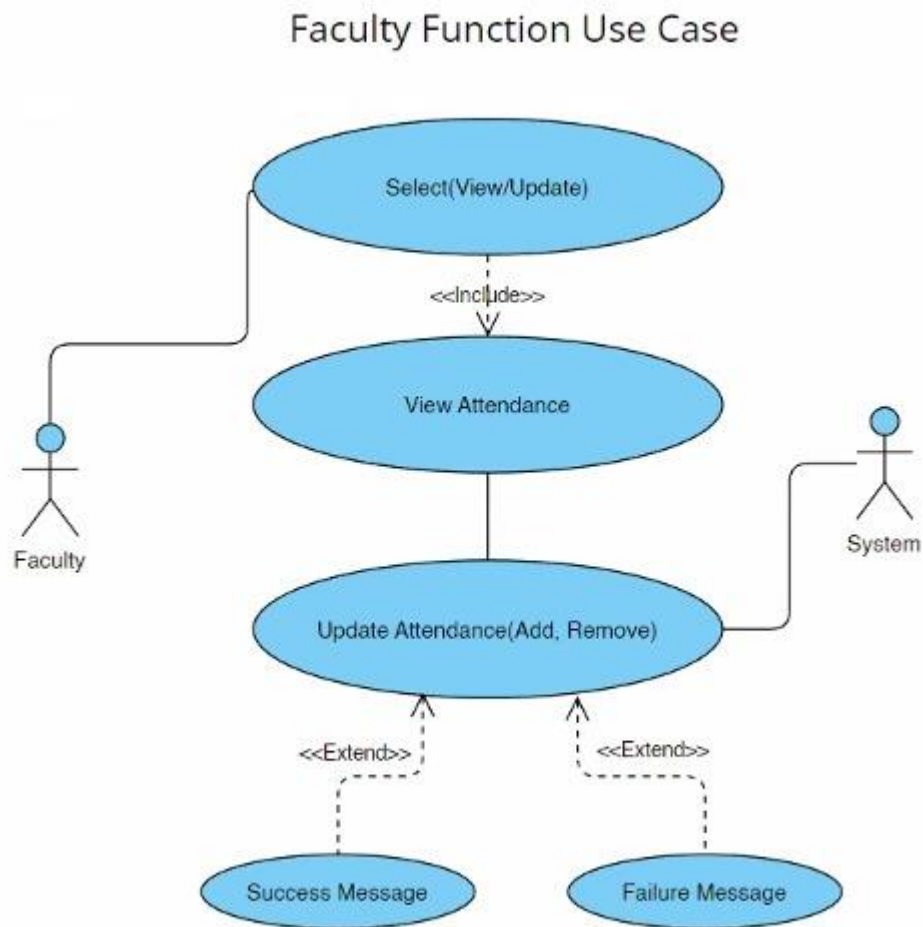
2.7.4 Admin Function Use Case:

Admin functions are: they can View the attendance of both Faculty and Student, Update the database by adding or removing the attendance of Faculty and Student using the system. If the update is successful, it will show a success message otherwise a failure message will be displayed.

**Fig. 2.14**

2.7.5 Faculty Function Use Case:

Faculty functions are: they can view the attendance of students, update the database by adding or removing the attendance of students using the system. If the update is successful, it will show a success message otherwise a failure message will be displayed.

**Fig. 2.15**

2.7.6 Student Function Use Case:

Student functions are: they can View their own attendance (Student) using the system. If the action is successful, it will show a success message otherwise a failure message will be displayed.

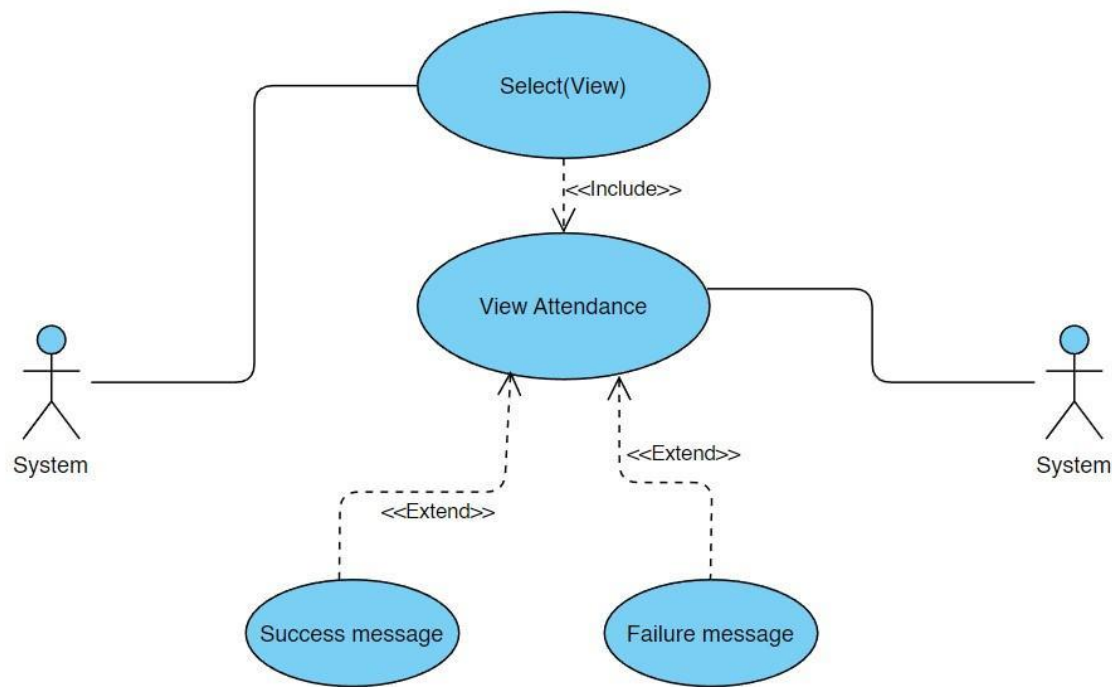


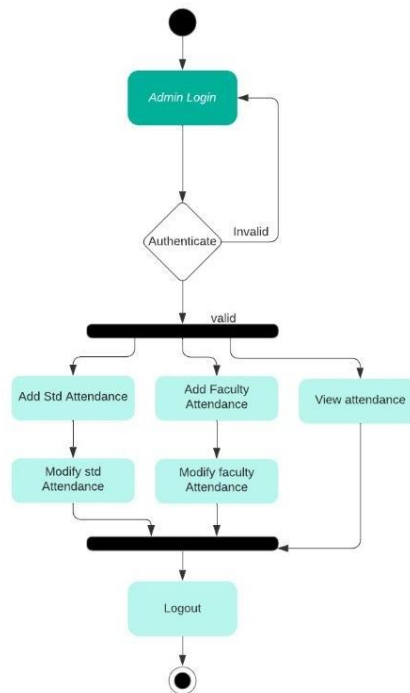
Fig. 2.16

2.8 Activity Diagram:

Activity diagram is another important behavioural diagram in UML diagram to describe dynamic aspects of the system. Activity diagram is essentially an advanced version of flow chart that modelling the flow from one activity to another activity. Activity Diagrams describe how activities are coordinated to provide a service which can be at different levels of abstraction. Typically, an event needs to be achieved by some operations, particularly where the operation is intended to achieve a number of different things that require coordination, or how the events in a single use case relate to one another, in particular, use cases where activities may overlap and require coordination. It is also suitable for modelling how a collection of use cases coordinates to represent business workflows.

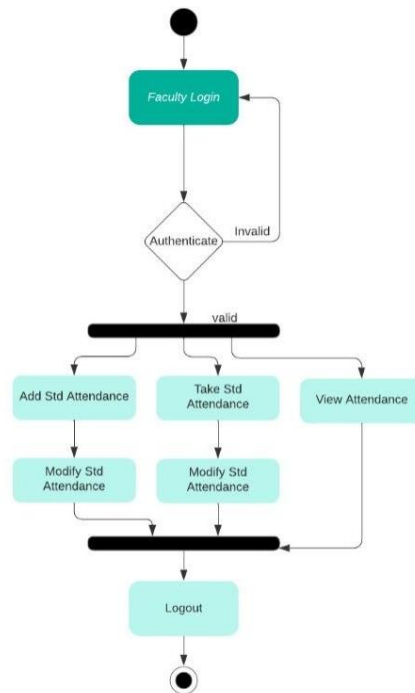
2.8.1 Admin Activity:

The activity starts, First the Admin logs in using their credentials (Username, Password) it then gets authenticated, if the credentials are incorrect, it will show invalid otherwise it will show valid and go ahead. Once logged in, they can add, view and modify the attendance of both faculty and student. And then can logout.

**Fig. 2.17**

2.8.2 Faculty Activity:

The activity starts, First the Faculty logs in using their credentials (Username, Password) it then gets authenticated, if the credentials are incorrect, it will show invalid otherwise it will show valid and go ahead. Once logged in, they can add, take, view and modify the attendance of students. And then can logout.

**Fig. 2.18**

2.8.3 Student Activity:

The activity starts, First the Student registers themselves by giving certain details. Then Student logins using their credentials (Username, Password) it then gets authenticated, if the credentials are incorrect, it will show invalid otherwise it will show valid and go ahead. Once logged in, they can view their own attendance. And then can logout.

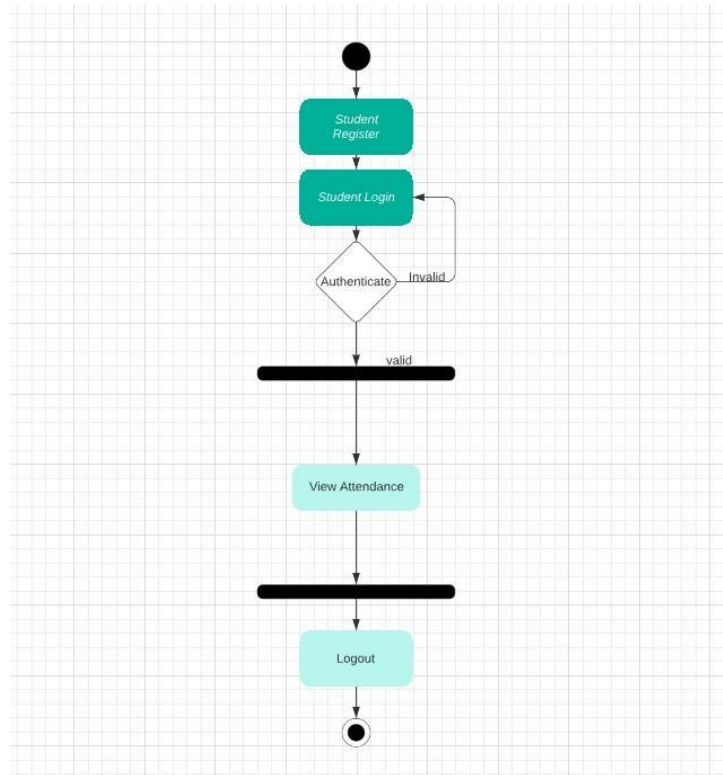


Fig. 2.19

2.9 Component Diagram:

Component diagrams are used in modelling the physical aspects of object-oriented systems that are used for visualizing, specifying, and documenting component-based systems and also for constructing executable systems through forward and reverse engineering. Component diagrams are essentially class diagrams that focus on a system's components that are often used to model the static implementation view of a system.

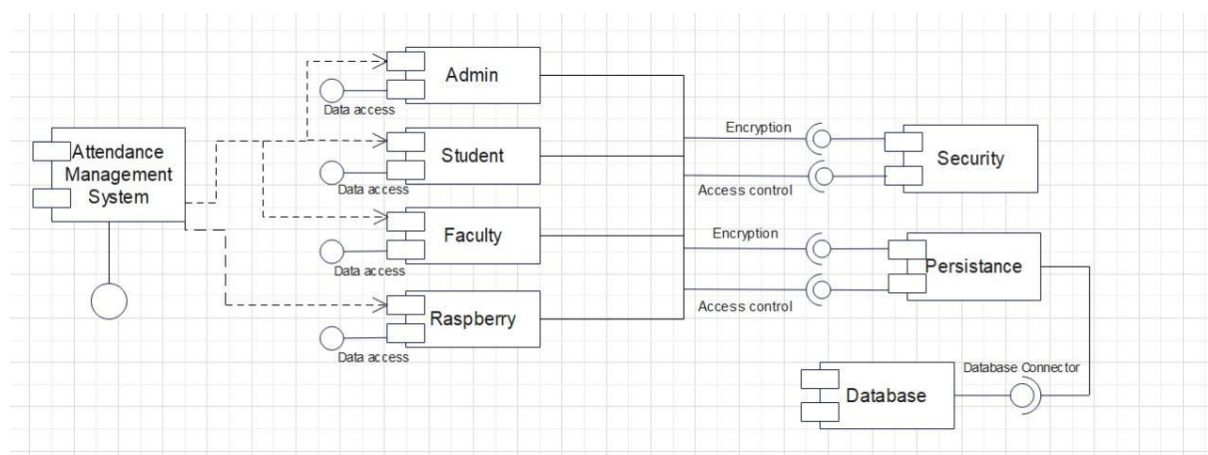


Fig. 2.2

2.10 Table Design:

2.10.1 Admin Table

Attributes	Description	Datatypes	Size	Required
<u>Adm_id</u>	The unique system generated ID given to admin	Integer	10	Yes
Adm_Password	Unique Password for admin	Varchar	50	Yes
F_id	ID of Faculty	Integer	10	Yes
Std_id	ID of Student	Integer	10	Yes

Table: 3

2.10.2 Faculty Table

Attributes	Description	Datatypes	Size	Required
<u>F_id</u>	The unique system generated ID given to admin	Integer	10	Yes
F_name	To define the Faculty Name	Varchar	50	Yes
F_class	Details of different class	Varchar	10	Yes
F_subject	Details of different subjects	Varchar	20	Yes
F_password	Unique Password for faculty	Varchar	50	Yes

Table: 4

2.10.3 Student Table

Attributes	Description	Datatypes	Size	Required
<u>Std_id</u>	The unique system generated ID given to admin	Integer	10	Yes
Std_name	To define the Student Name	Varchar	50	Yes
Std_class	Details of different class	Varchar	10	Yes
Std_subject	Details of different subjects	Varchar	20	Yes
Std_email	Email of student	Varchar	100	Yes
Std_phone	Contact Number of Student	Varchar	12	Yes
Std_password	Unique Password for student	Varchar	50	Yes

Table: 5

2.10.4 Attendance Table

Attributes	Description	Datatypes	Size	Required
Atd_date	Date of Attendance	Date	10	Yes
Atd_subject	Subject of Attendance	Varchar	20	Yes
Std_id	ID of Student	Integer	10	Yes
F_id	ID of Faculty	Integer	10	Yes
Atd_class	Attendance for class	Varchar	10	Yes

Table: 6

2.11 Deployment Diagram:

Deployment diagrams are used to visualize the topology of the physical components of a system, where the software components are deployed. Deployment diagrams are used to describe the static deployment view of a system. Deployment diagrams consist of nodes and their relationships.

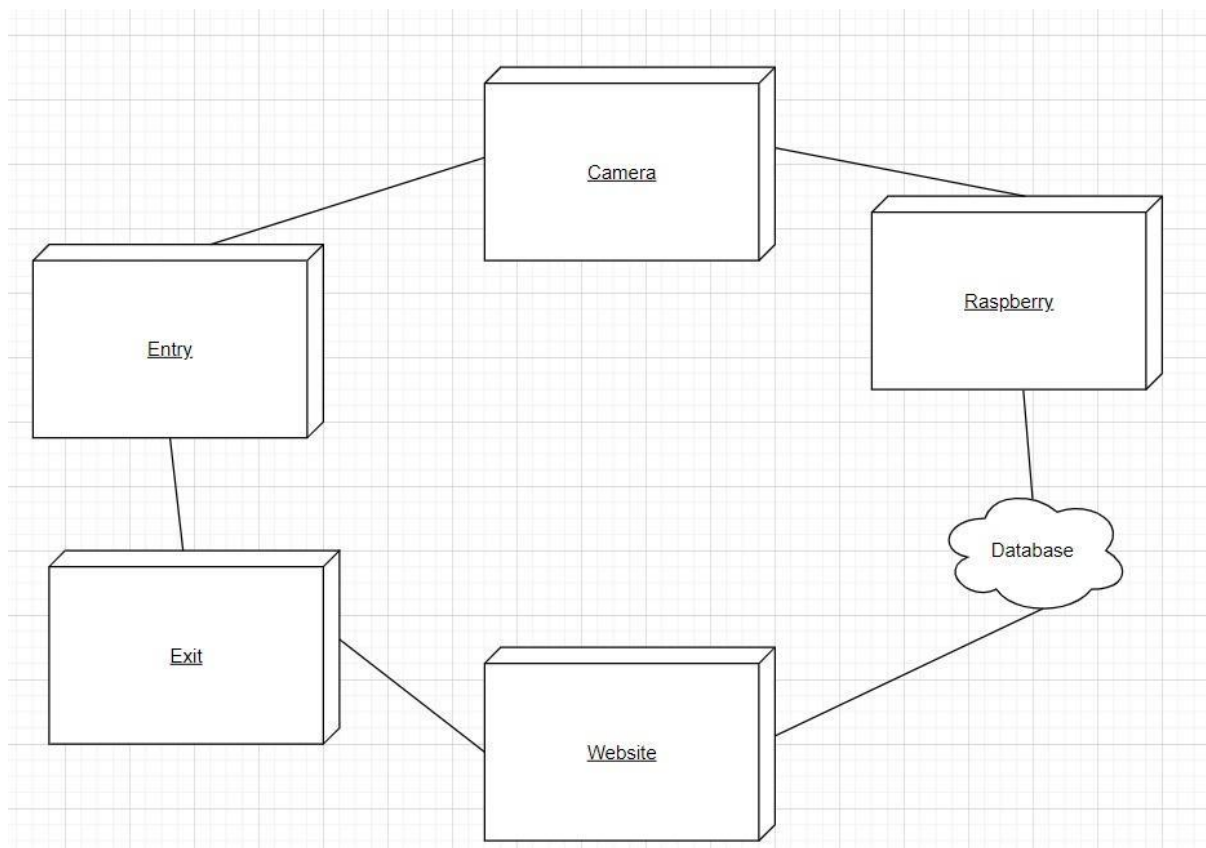
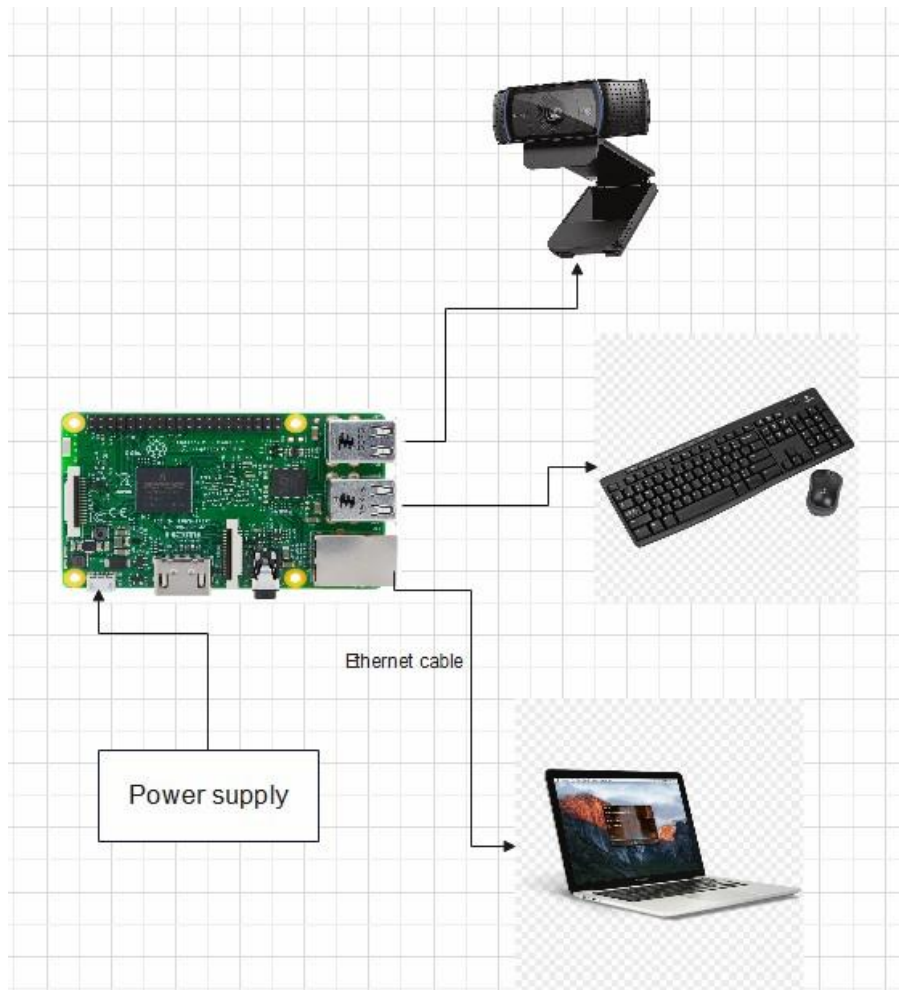


Fig. 2.21**2.12 Circuit Diagram:****Fig. 2.22****2.13 Component Level Description & Specification:****A. Raspberry Pi 3 Board:**

Raspberry Pi can be called a mini-computer which is of the same size as that of a credit card. The Raspberry Pi is a small and affordable single-board computer that we will use to design and develop practical IoT devices while learning about programming language and computer hardware systems. In addition, you will learn how to set up the Raspberry Pi, get a Linux operating system running, and write and execute some basic Python code on the Raspberry Pi. You will also learn how to use Python based IDE

(integrated development environments) for the Raspberry Pi and how to trace and debug Python code on the device.



Fig. 2.23

B. USB Camera:

The camera plays a vital role in mechanization purposes. The camera is used in our work to monitor the room from a remote place and to capture video to detect faces. For interfacing the USB camera with Raspberry Pi use the terminal and run "sudo raspi-config". If the "camera" option is not scheduled, you will need to dash a few instructions to update your Raspberry Pi. For this Run "sudo apt-get update" and "sudo apt-get upgrade" commands. Again run "sudo raspi-config" again - you must now see the "camera" preference and enable it.



Fig. 2.24

2.14 Block Diagram:

A block diagram is a diagram of a system in which the principal parts or functions are represented by blocks connected by lines that show the relationships of the blocks. They are heavily used in engineering in hardware design, electronic design, software design, and process flow diagrams.

Once when the button is pressed Raspberry pi is turned on while it's connected with the Attendance Database.

Later, User face is detected with the help of the raspberry pi and the data stores in the Attendance Database.

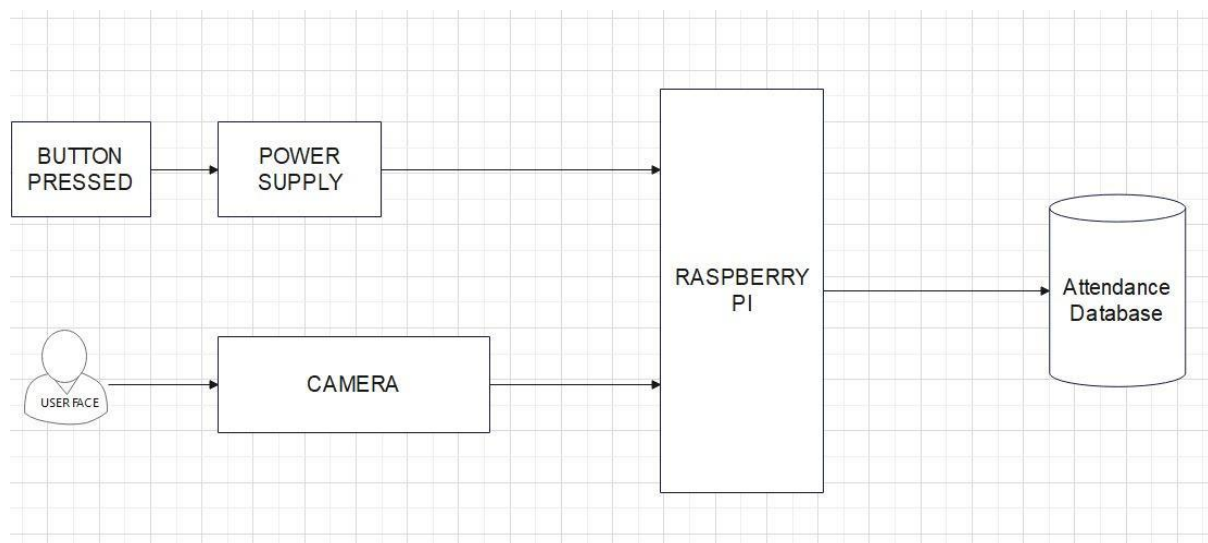


Fig. 2.25

2.15 State Transition:

2.15.1 Admin and Faculty State Transition

In the Admin and Faculty state transition, there are 4 states: Authentication, Add, View and Edit.

Authentication state has login() function and verified event occurs, Add state has add_attendance() function, after submitted event there is View state which has view_attendance() function and view event occurs and the last state is Edit which has edit_attendance() function. In the edit state the conditions should be fulfilled by reviewing it.

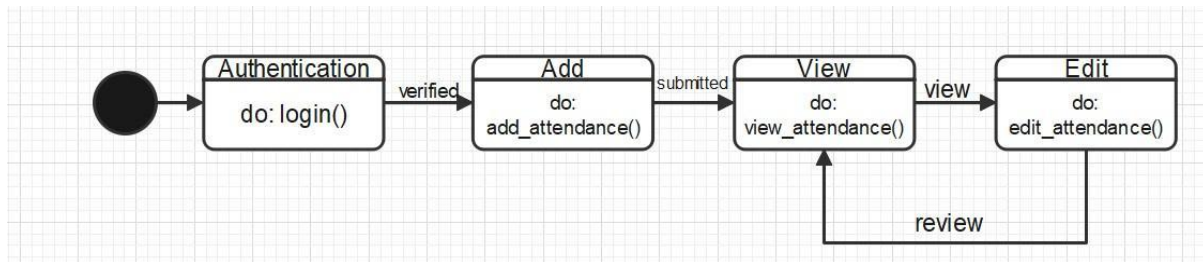


Fig. 2.26

2.15.2 Student State Transition

In the Student state transition, there are 3 states: Registration, Authentication and View.

Registration state has register () function and verified event occurs, Authentication state has login () function, after verified event there is View state which has view_attendance() function.

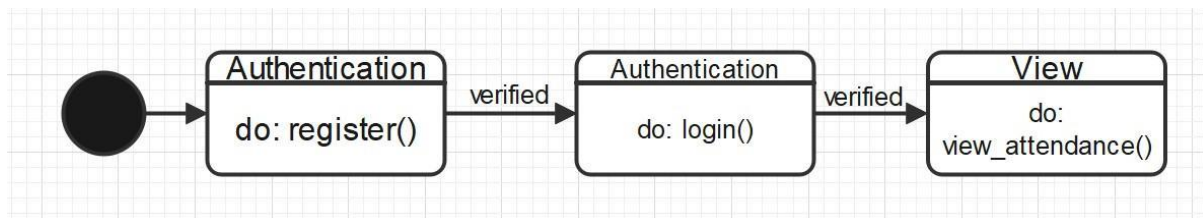


Fig. 2.27

2.16 System Architecture Diagram:

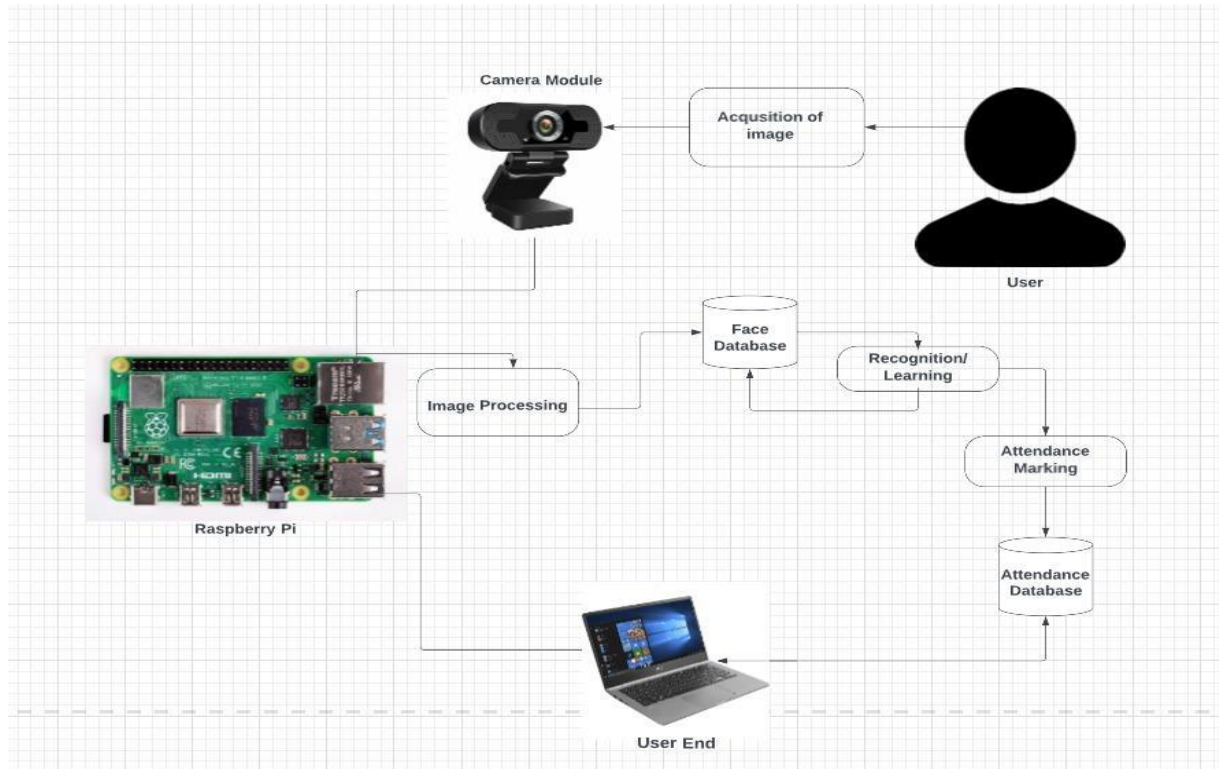


Fig. 2.28

2.17 Machine Learning

Machine learning (ML) is a field of inquiry devoted to understanding and building methods that ‘learn’, that is methods that leverage data to improve performance on some set of tasks. It is a branch of artificial intelligence (AI) and computer science, which focuses on the use of the data and algorithms to imitate the way that humans learn, gradually improving its accuracy. ML algorithms build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so. Machine learning algorithms are used in a wide variety of applications, such as in speech recognition, email filtering, face recognition and much more.

2.18 Face Recognition

Face recognition is a method of identifying or verifying the identity of an individual using their face. Face recognition systems are used to identify people in photos, videos, or in real-time. Here we give the system a set of images of a person, and try to recognize the person by comparing the image from the database.

2.19 Face Recognition Based Attendance System

A face recognition attendance system incorporates face recognition technology to recognize and verify a student's or person's facial features and to record attendance automatically. A face recognition attendance system is a non-contact approach to managing students' attendance.

2.20 Working of Face Recognition Module

1. First, the image is captured from a webcam or camera.
2. Eye locations are determined.
3. Then the image is converted to grayscale and cropped and converted to a template used by the search engine for facial comparison results.
4. Image is searched and matched using an algorithm to compare the template to other images in the database.

3. System Planning

3.1 Gantt Chart:

A Gantt chart, commonly used in project management, is one of the most popular and useful ways of showing activities (tasks or events) displayed against time. On the left of the chart is a list of the activities and along the top is a suitable time scale. Each activity is represented by a bar; the position and length of the bar reflect the start date, duration, and end date of the activity. This allows you to see briefly:

- What the various activities are
- When each activity begins and ends
- How long each activity is scheduled to last
- Where activities overlap with other activities, and by how much
- The start and end date of the whole project.

7 Months Activity



Fig. 3.1

3.2 Activity Chart Diagram:

A Project Activity Diagram is a chart that shows the arrangement of tasks within a project. It can also contain other information such as the duration and connection between each task. This is an essential part of project management since you will be able to streamline the execution process.

Aside from the ones mentioned above, there are other benefits that you can get from using a Project network chart. With that being said, we listed down some of the benefits of using this diagram.

- **Time-efficiency**

This diagram shows the estimated time needed for a task to finish. Having a clear idea about the time it takes to complete a task, gives the project manager a clear view of the next steps needed to proceed with the task.

- **Control over the task sequence**

As a project manager, you are responsible for making sure that all steps are done in chronological order. This is where project network diagram examples come in. By using this tool, you can ensure that all necessary tasks are done without any complications to avoid any future issues.

- **Task time management**

There is no absolute way to avoid minor hiccups in any project. However, there are ways to minimize the effects of these hiccups by knowing when to do the mending even before the actual problem arrives.

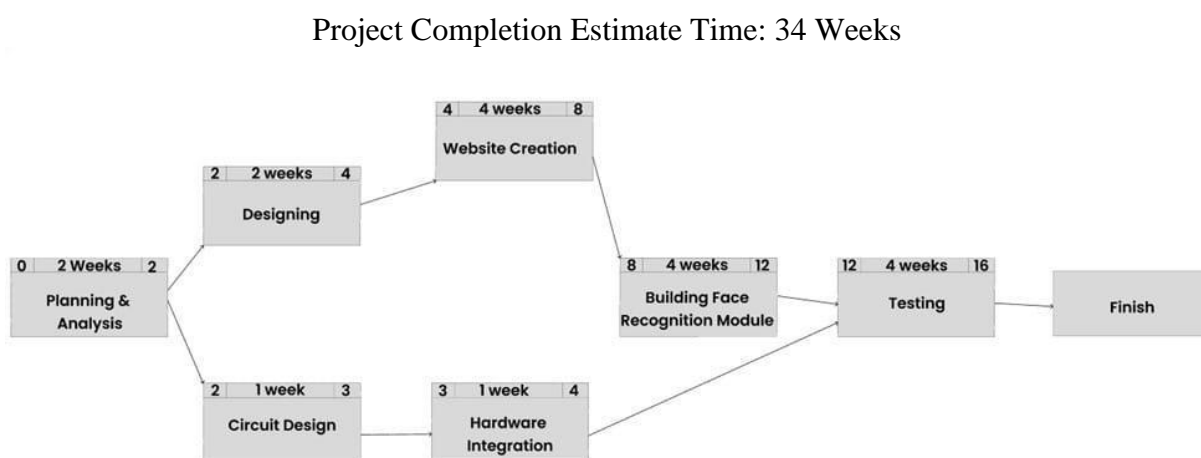


Fig. 3.2

Activity	Description	Predecessor
A	Planning And Analysis	-
B	Designing	A
C	Circuit Design	A
D	Hardware Integration	C
E	Website Creation	B
F	Building Face Recognition Module	F

4. System Implementation

Following are the screenshots showing the system implementation:

1) Home Page:

This is the first screen the user will see upon accessing the website.



2) Registration Page:

User can register themselves to the system by filling out this form.



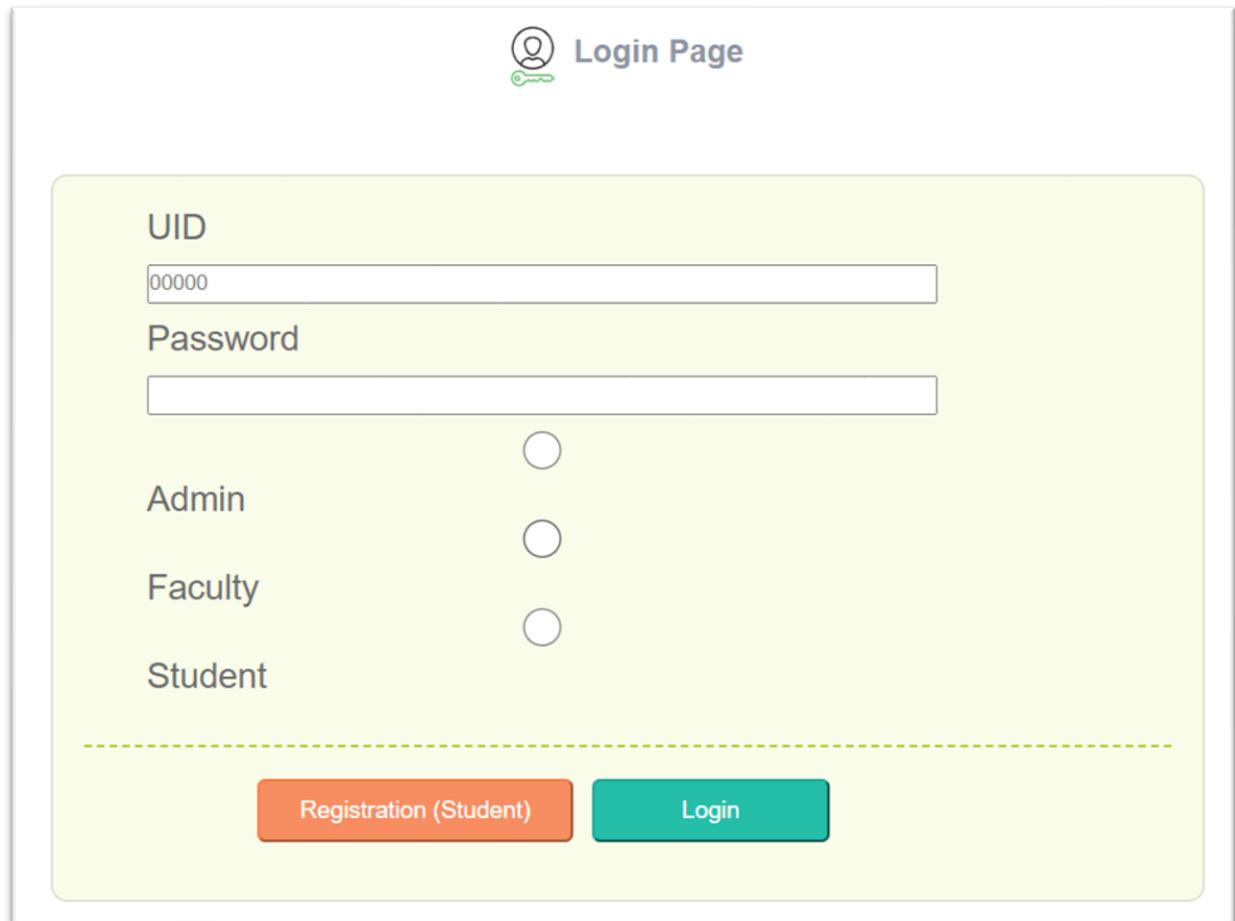
Registration Page

Class	Division
--Select a class--	--Select a division--
<hr/>	
Back to Login Page	Save and Continue

©Site is designed and hosted by Mugiwara

3) Login Page:

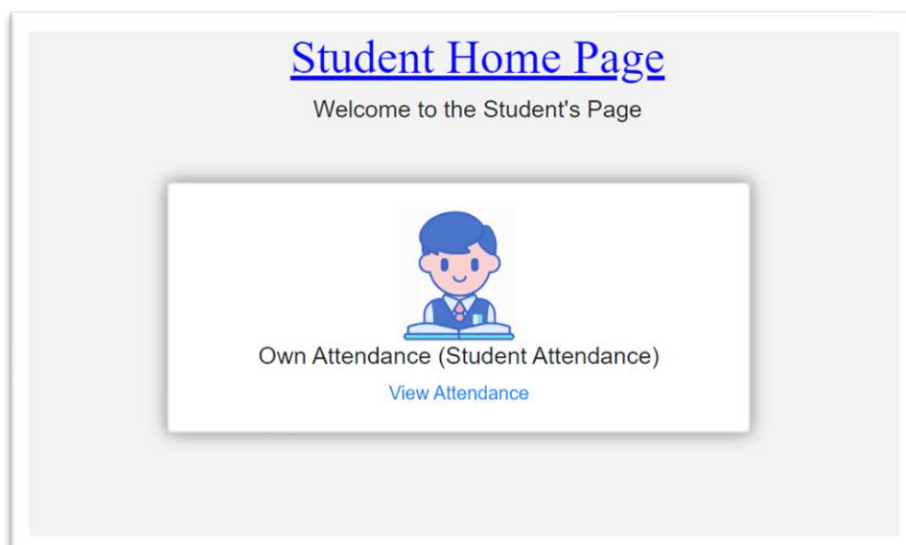
User can login themselves to the system by filling out this form.



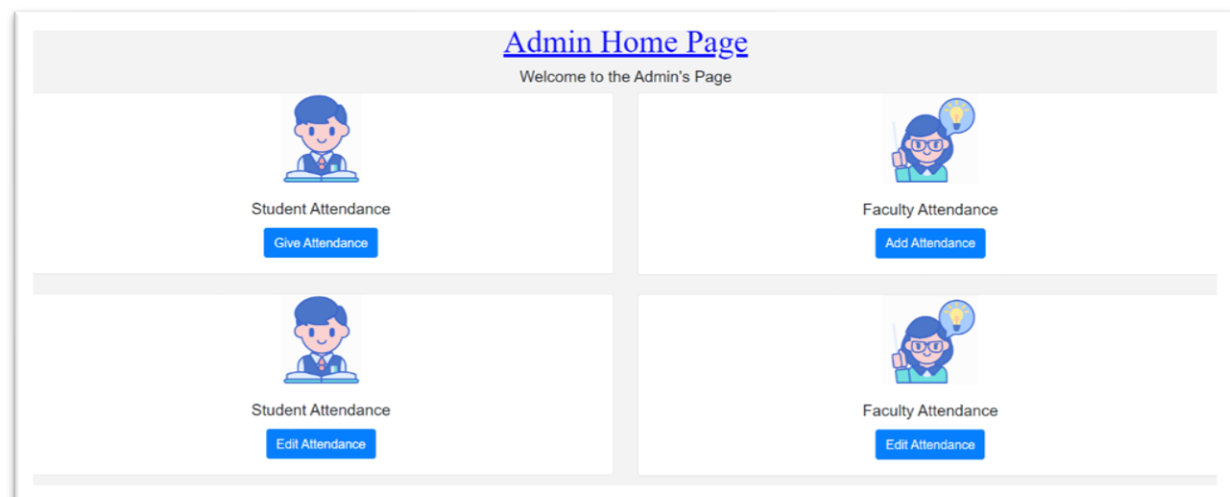
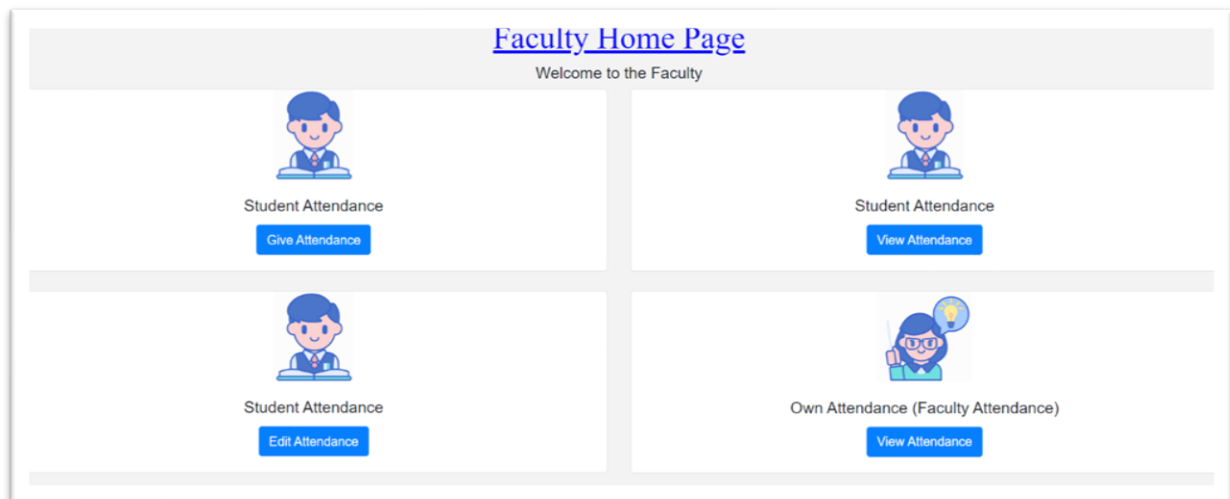
The screenshot shows a web page titled "Login Page" with a user icon. The page has a light green background. It contains two input fields: "UID" with the value "00000" and "Password". Below these fields are three radio buttons for user roles: "Admin", "Faculty", and "Student". At the bottom, there are two buttons: "Registration (Student)" in orange and "Login" in teal.

4) Student Home Page:

This is the page the person will see when they login with their respective credentials based on their access level.



The screenshot shows a web page titled "Student Home Page" with the subtitle "Welcome to the Student's Page". It features a cartoon illustration of a student with blue hair, wearing a blue shirt and tie, sitting at a desk. Below the illustration, the text "Own Attendance (Student Attendance)" is displayed, followed by a blue link "View Attendance".



5) Mark Attendance Page:

This is the page when the Admin and Faculty can give the attendance.

The screenshot shows the 'MARK ATTENDANCE' form. At the top, there is a header banner for Thakur Educational Trust's (Regd.) THAKUR COLLEGE OF SCIENCE & COMMERCE. The banner includes the college's logo, its affiliation with the University of Mumbai, NAAC accreditation, and a 'Best College Award' for 2018-2019. The form itself is a white box with a red title 'MARK ATTENDANCE'. It contains two dropdown menus for 'Select a class--' and 'Select a division--', a text input for 'Subject', a date picker for 'Date:' with a placeholder 'mm/dd/yyyy', and an orange 'Continue' button at the bottom.

6) Modify Attendance Page:

This is the page where the Admin and Faculty can Modify the attendance of the respective person by adding or removing the attendance.

The screenshot shows the 'CLASS MODIFY ATTENDANCE LIST' form. It has a red title 'CLASS MODIFY ATTENDANCE LIST'. The form includes two dropdown menus for 'Select a class--' and 'Select a division--', a date picker for 'Date:' with a placeholder 'mm/dd/yyyy', and four text input fields for 'Subject', 'UID', and 'Attendance'. An orange 'Modify Attendance' button is located at the bottom.

CLASS MODIFY ATTENDANCE LIST
Date:

UID

Modify Attendance

7) View Attendance Page:

This is the page where the Admin and Faculty and Student can view the attendance of the respective person.

VIEW STUDENT ATTENDANCE
View Class Attendance **View By Roll Number**

CLASS VIEW ATTENDANCE

--Select a class--

--Select a division--

Date:
mm/dd/yyyy

View Attendance

<div><div>THAKUR TRUSTS</div><div><i>Thakur Educational Trust's (Regd.)</i> THAKUR COLLEGE OF SCIENCE & COMMERCE AUTONOMOUS COLLEGE, PERMANENTLY AFFILIATED TO UNIVERSITY OF MUMBAI NAAC Accredited Grade 'A' (3rd Cycle) & ISO 9001: 2015 (Certified) Best College Award by University of Mumbai for the Year 2018-2019</div><div>75 Azadi Ka Amrit Mahotsav</div><div>tcsc CELEBRATING 25 YEARS OF GLORY</div></div>			
Date	UID	Subject	Attendance
2023-01-24	22645	abcd	Present
2023-01-24	22628	abcd	Present

5. Cost and Benefit Analysis and Software Parameter Estimation

As defined, “Cost Estimation is a statement that gives the value of the cost incurred in the manufacturing of finished goods. Cost estimation helps in fixing the selling price of the final product after charging appropriate overheads and allowing a certain margin for profits.”

There are 5 Functional units used to calculate Function Point (FP):

1. Internal Logic Files (**ILF**): To control the information or logically related data that is present within the system.
2. External Interface Files (**EIF**): The control data referenced by the system but present in another system.
3. External Inputs (**EI**): Data/ control info that comes from outside our system.
4. External Outputs (**EO**): Data that goes out of the system after generation.
5. External Inquires (**EQ**): Combination of input-output resulting data retrieval.

To Compute FP:

We'll use, **FP = UFP * CAF**

Where, **UFP** = Unadjusted Function Point &

CAF = Complexity Adjustment Factor

Step 1: Calculate **UFP**

To find **UFP** we need to sum all the Complexities of all the **EI**, **EO**, **EQ**, **ILF** and **EIF**.

Function Unit	Weighting Factors					
	Count		Low	Average	High	
External Inputs (EI)	12	*	3	<u>4</u>	6	48
External Outputs (EO)	10	*	4	<u>5</u>	7	50
External Inquires (EQ)	14	*	3	<u>4</u>	6	56
Internal Logic Files (ILF)	4	*	7	<u>10</u>	15	40
External Interface Files (EIF)	14	*	5	<u>7</u>	10	98
Total Count	292					

Table: 7**Step 2: Calculating CAF:**

Formula: **CAF** = 0.65 + (0.01 * **DI**)

Where, **DI** = Value adjustment factors based on responses to the following 14 questions

1	Data Communication	5
2	Distributed Data Processing	4
3	Performance Criteria	4
4	Heavily Utilized Hardware	5
5	Online Data Entry	5
6	High Transaction Rate	0
7	Online Updating	2
8	End-user Efficiency	5
9	Complex Computations	4
10	Reusability	2
11	Ease of Installation	2
12	Ease of Operation	3
13	Portability	2
14	Maintainability	4
Degree of Influence (DI)		47

Table: 8

Step 3: Calculating **Function Point**

$$\text{Function Point (FP)} = \text{UFP} * (0.65 + 0.01 * \text{DI})$$

$$\text{Function Point (FP)} = 292 * (0.65 + 0.01 * 47)$$

$$\text{Function Point (FP)} = 292 * 1.12$$

$$\text{Function Point (FP)} = 327.04$$

That Means, Function Point is **327.04**

COCOMO MODEL

COCOMO stands for Constructive Cost Model. It is a regression model based on LOC i.e., lines of code. It is a procedural cost estimate model for software projects and is often used as process of reliably predicting the various parameters associated with making a project such as the effort, development time, average team size and effort.

COCOMO model are classified to 3 categories including, **organic**, **semi-detached** and **embedded**.

In our project we have used Organic COCOMO model, because it well suited for a small software team, where the developers have familiar environment and there are no tight deadlines and it helps to understand the problem well and solve it easily.

Whereas, Semi-detached and Embedded are for large software teams and complex projects.

There are Basic COCOMO, Intermediate COCOMO, and Detailed COCOMO.

By default, the Basic COCOMO is used in Organic mode.

Arithmetic Formula

1. Effort applied to the project: $E = a(\text{KLOC})^b$ (in Person – month)
2. Development time: $D = c(\text{Effort})^d$ (in month)
3. Average Staff Size: $P = \text{Effort} / \text{Development time}$ (in Person)
4. Productivity: $\text{KLOC} / \text{Effort}$ (KLOC/person-month)

SW Project	a	b	c	d
Organic	2.4	1.05	2.5	0.38
Semi-Detached	3.0	1.12	2.5	0.35
Embedded	3.6	1.20	2.5	0.32

Table: 9

Calculating cost using COCOMO

We have KLOC = 3

- 1) Effort = $a (\text{KLOC})^b$
 $= 2.4 (3)^{1.05}$
 $= 7.6$ [In person month]
- 2) Development Time = $c (\text{Effort})^d$
 $= 2.5 (7.6)^{0.38}$
 $= 5.4$ [Months]
- 3) Staff size = Effort / Development time
 $= 7.6 / 5.4$
 $= 1.407$ [Persons]
- 4) Productivity = KLOC / Efforts
 $= 3/7.6$
 $= 0.39$

6. System Testing

System Testing:

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black-box testing, and as such, should require no knowledge of the inner design of the code or logic.

The facial recognition-based attendance management system undergoes system testing to ensure that the system adheres to its specified requirements. System testing involves a combination of both functional and non-functional testing, which includes unit testing, integration testing, validation testing, performance testing, and user acceptance testing.

System testing is a combination of both functional and non-functional testing. Functional testing, in simple terms, tests the functionality of whether an application is working to requirements or not. (Smoke, sanity, retesting, regression, priority-based testing, risk-based testing etc.)

6.1 Methodology used for testing

In this project we have used the following methodologies for testing purposes:

- **Unit Testing:**

Unit testing is the testing of an individual unit or group of related units. It falls under the class of white box testing. Unit testing is carried out by the developers to check if the individual units are producing the expected output for a given input.

- **Integration Testing:**

Integration testing is testing in which a group of components are combined to produce output. Also, the interaction between software and hardware is tested in integration testing if software and hardware components have any relation. It may fall under both white box testing and black box testing.

- **Validation Testing:**

Validation is the process to make sure the product satisfies the specified requirements at the end of the development phase. In other words, to make sure the product is built as per customer requirements.

Validation testing ensures that the system satisfies the customer's specified requirements and that the product is built as per the customer's needs. Performance testing assesses the system's speed and efficiency and ensures that it generates results within a specified time frame.

- **Performance Testing:**

Performance testing is the testing to assess the speed and effectiveness of the system and to make sure it is generating results within a specified time as in performance requirements. It falls under the class of black box testing.

- **User Acceptance Testing:**

Acceptance testing is often done by the customer to ensure that the delivered product meets the requirements and works as the customer expected. It falls under the class of black box testing. Unit testing is carried out by the developers to check if the individual units are producing the expected output for a given input. Integration testing is conducted to test the interaction between the software and hardware components, as well as to combine the related groups of components to produce the desired output.

- **White Box Testing:**

- A software testing method in which the internal structure/design/implementation of the item being tested is known to the tester.
- So here we the group members have tested the code and checked if the codes are running properly as they should.
- Each file of the project was tested by us.
- Also checking the inputs and outputs.
- We have also tested the database to verify data integrity

- **Black Box Testing:**

Black box testing is a type of software testing in which the functionality of the software is not known. The testing is done without the internal knowledge of the products. Solely based on the inputs and outputs.

- **Manual Testing:**

A method of testing whereby software is tested manually (by a human). Manual testing is a crucial aspect of the software testing process, and it plays an essential role in ensuring the quality and integrity of System Using Face Recognition Using Raspberry Pi system .

Manual testing is the process of verifying the functionality of an application manually, without the use of any automated tools or scripts. In the case of the System Using Face Recognition Using Raspberry Pi system, manual testing is carried out by our team acting as testers by simulating the various scenarios that users may encounter when using the app. The goal of manual testing is to identify any defects or issues in the app's functionality and user interface, as well as to ensure that the app meets the user's expectations and requirements.

- **Equivalence Partitioning Method:**

Also known as Equivalence Testing, It is a software testing technique or black-box testing that divides input domains into classes of data, and with the help of these classes of data, test cases can be derived. An ideal test case identifies a class of error that might require many arbitrary test cases to be executed before general error is observed. Whenever any input is given, then type of input condition is checked, then for these input conditions, Equivalence class represents or describes a set of valid or invalid states.

Equivalence Partitioning follows specific guidelines for example, If the range condition is given as an input, then one valid and two invalid equivalence classes are defined. If a specific value is given as input, then one valid and two invalid equivalence classes are defined. If a member of a set is given as an input, then one valid and one invalid equivalence class is defined. If Boolean no. is given as an input condition, then one valid and one invalid equivalence class is defined.

With respect to our system, the attendances are divided by respective classes A, B, C, etc. so these will be our values. The team performed tests to determine if the user's attendance would be registered in the correct class.

6.2 Data Used for Test Cases:

The proper selection of the data is very important. If the test data is not appropriate or representative of the data to be provided by the user, the result of the test case becomes unreliable.

○ Facial Data: The facial data is generated by collecting facial scans of our friends and family and assigning information to it, later testing if the system can correctly identify the people based on the earlier facial scans.

6.3 Test Cases

Sr.no	Test case	Expected result	Steps	Outcome
1	Registration (User Data)	Details successfully stored	Fill out the details in the registration page.	Successful
2	Login (User data)	Retrieving existing account	Fill out the details in the login page.	Successful
3	Give attendance (Manual)	Attendance should be marked	Enter the student's credentials	Successful
4	View attendance	Attendance should be displayed	Login and view the attendance page.	Successful
5	Facial recognition	Faces should be correctly identified.	Program should run automatically	Successful
6	Recognition capacity	Should be able to recognize 10 faces.	Program should run automatically	Failed, recognizing only 5 faces.
7	Attendance marking (Auto)	Attendance should be marked automatically	Program should run automatically	Successful
8	Module Access testing	Various modules such as Admin, Faculty	Testing Each module with respect to its functions	Successful

		should work as intended		
9	Security	The security features such as encryption should work as planned	Testing image encryption and database security	Successful
10	Usability	All links and buttons should work as expected	Conducting a level 1 testing on the system with respect to buttons and links	Successful
11	Raspberry Pi Malfunction/ Unavailability	System should work normally as long as a camera is provided (eg: desktop)	Running the system without the Raspberry Pi	Successful
12	Attendance being marked in the correct class	The system should register the attendance into the attendance class up recognition	Running recognition on a pre-registered person	Successful

7. System Evaluation and Maintenance Report

The Facial Recognition Attendance Management System is designed to accurately record employee attendance using facial recognition technology. In this report, we evaluate the system's performance, identify potential areas of improvement, and outline a maintenance plan to ensure that the system continues to function effectively.

7.1 System Evaluation:

We evaluated the system based on its performance, usability, and reliability. The following are our findings:

Performance: The system performs well in recognizing faces and recording attendance. It is fast and accurate in identifying employees, even under varying lighting conditions.

Usability: The system is easy to use, and the user interface is intuitive. The system provides clear instructions on how to use it, and the facial recognition process is straightforward.

Reliability: The system is reliable, and we did not experience any significant issues during our evaluation. However, we did observe that the system struggled to recognize faces when employees were wearing glasses or hats.

Areas for Improvement

Although the system performed well during our evaluation, there are still some areas for improvement. The following are our recommendations:

Enhance the facial recognition algorithm to improve accuracy and speed.

Develop a mechanism to handle face recognition when employees are wearing hats, glasses or masks.

Include a backup power source, such as a battery or generator, to prevent data loss during power outages.

Improve the system's reporting capabilities to provide more detailed insights into attendance patterns.

7.2 Maintenance Plan

To ensure that the Facial Recognition Attendance Management System continues to function effectively, we recommend the following maintenance plan:

Regularly clean the cameras and sensors to ensure optimal performance.

Regularly update the software and firmware to ensure that the system is up to date with the latest security patches and features.

Regularly back up attendance records to prevent data loss in the event of a system failure.

Conduct regular testing to identify and address any potential issues before they become critical.

Train employees on how to use the system effectively and troubleshoot common issues.

The Facial Recognition Attendance Management System is a reliable and efficient solution for managing employee attendance. While there are some areas for improvement, the system performs well overall, and with regular maintenance, it can continue to function effectively for years to come. By implementing the maintenance plan outlined in this report, organizations can ensure that their attendance management processes are streamlined, accurate, and reliable.

8. User Manual

The Attendance Management System using Facial Recognition is a user-friendly and efficient way to mark attendance. It is designed to make attendance simple and hassle-free for schools, universities, companies, and other organizations. This system works by capturing the image of the person and recognizing their face using facial recognition technology.

Hardware Requirements:

The system requires a Raspberry Pi board, a Raspberry Pi camera module, a monitor, and an internet connection. These hardware components are readily available and can be easily obtained.

(Note: Even though a Raspberry Pi is recommended, the system is built to be able to work without it with a few modifications this is done to ensure system availability incase of Raspberry Pi failure or unavailability).

Software Requirements:

The following software needs to be installed for the system to function correctly:

Python

OpenCV library

Dlib library

face_recognition library

Flask library

Installation:

Follow the steps below to install the required software:

Install Python 3 on your computer/server..

Install the OpenCV library by running the command: `pip install opencv-python`

Install the Dlib library by running the command: `pip install dlib`

Install the face_recognition library by running the command: `pip install face_recognition`

Install the Flask library by running the command: `pip install Flask`

For the Hardware i.e Camera module, ensure the camera is in a well lit area where it can scan faces without obstruction. We recommend placing it over the entry door or gate with ample lighting.

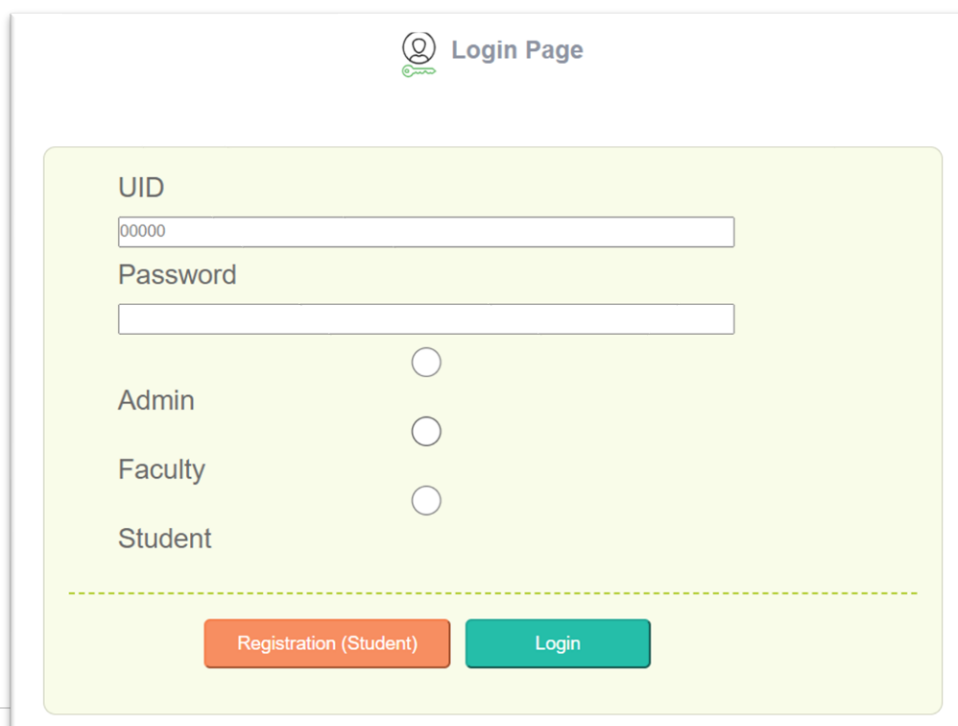
User Interface:

The user interface is simple and easy to use. It consists of 3 modules: Student, Faculty and Admin.

Admin:

The admin module can't be registered like the other modules, it can be created directly during installation or later by an existing admin.

The admin can login to the system by entering their credentials on the login page and selecting the admin module. Upon login the admin had the highest access level and power in the system hence, they can modify, delete or view attendances, profiles or add classes for both student and faculty. They can simply perform any of these functions by selecting the respective module icon on their screen.



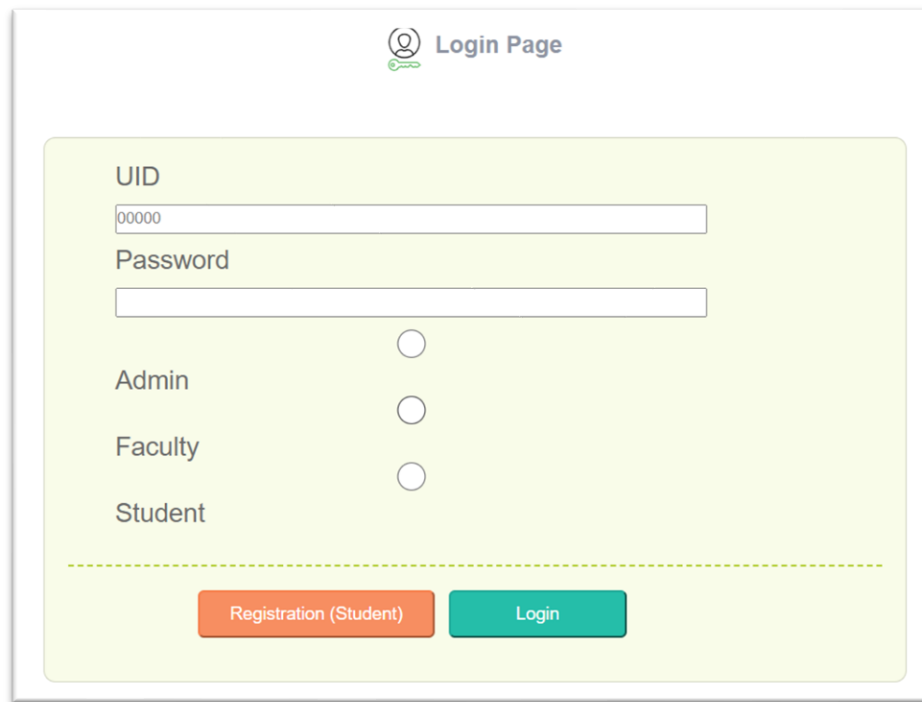
The screenshot shows a web-based login interface. At the top, there's a header with a key icon and the text "Login Page". Below this, there's a light green box containing the login form. The form has two input fields: "UID" with the value "00000" and "Password". Below the password field are three radio buttons labeled "Admin", "Faculty", and "Student". At the bottom of the form, there are two buttons: "Registration (Student)" in orange and "Login" in teal.



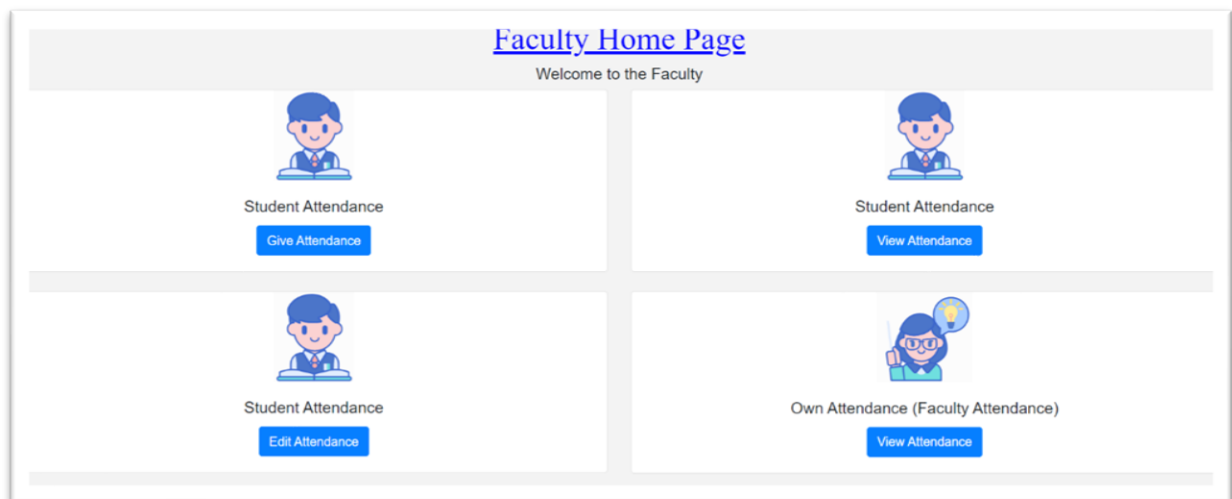
Faculty:

The faculty module can be created by the admin. This is done to make the system more secure. The faculty can login into the system by selecting the admin module in the login page, upon logging in the faculty can Either view, add or delete student attendances. They are also authorized to view their own attendance however cannot edit it.

The faculty has a manual attendance feature in order to ensure functionality in case of module, server or camera failure. The faculty can manually mark the attendance of the respective students by going to the mark attendance tab selecting class and marking present. Similarly, they can also mark students absent. They can view the attendance of students by selecting the view student attendance section and entering class or id details. In order to view their own attendance, they can simply click on view attendance and view it.



The image shows a web-based login page titled "Login Page" with a user icon. It features a light green background for the form area. There are two input fields: "UID" with the value "00000" and "Password". Below these fields are three radio buttons for user roles: "Admin", "Faculty", and "Student". At the bottom, there are two buttons: "Registration (Student)" in orange and "Login" in green.



The image shows a web-based interface titled "Faculty Home Page" with a subtitle "Welcome to the Faculty". It is divided into four quadrants, each with a cartoon icon of a person and a button. Top-left: "Student Attendance" with a "Give Attendance" button. Top-right: "Student Attendance" with a "View Attendance" button. Bottom-left: "Student Attendance" with an "Edit Attendance" button. Bottom-right: "Own Attendance (Faculty Attendance)" with a "View Attendance" button.

Student:

The students need to register in order to gain access to the system. they can register themselves by clicking on the register link in the home page. After entering details such as UID, Name, Class, Div, Phone Number, Confirm password, Email Id. Once registered they can log into the system by entering the credentials. After logging in they have the access to view their own attendance. They can do this by selecting the view attendance module on the screen.



Registration Page

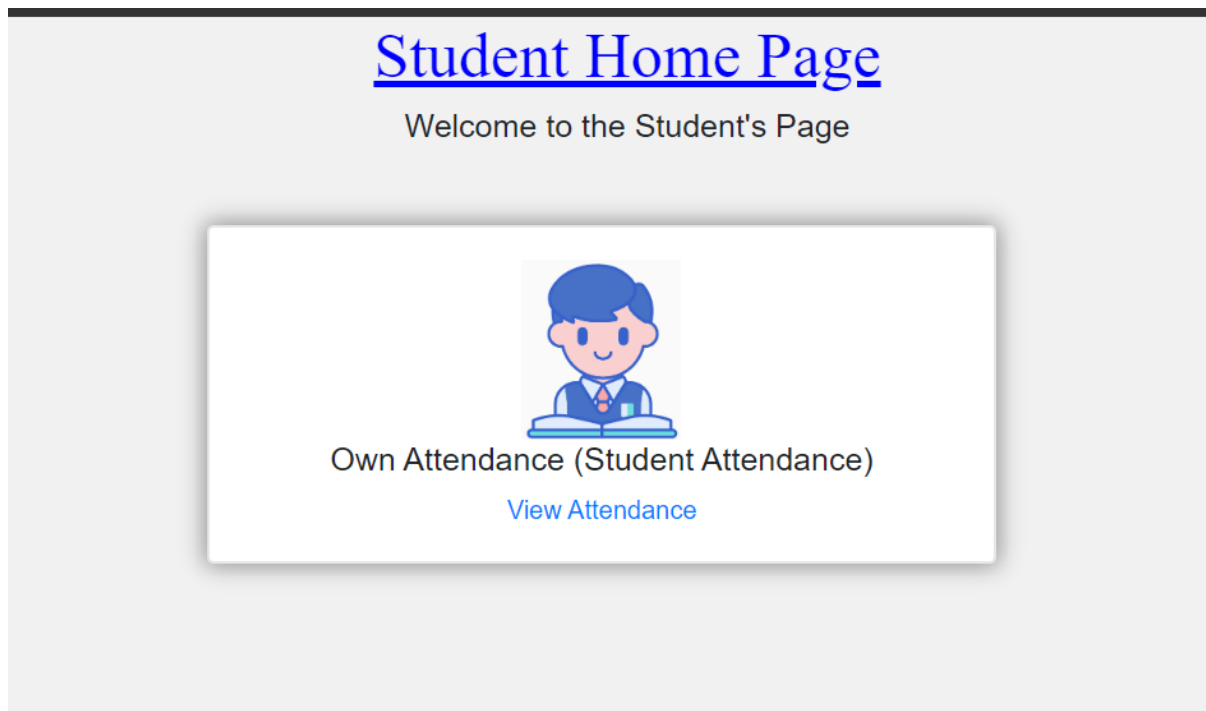
Class	Division
<input type="text" value="--Select a class--"/>	<input type="text" value="--Select a division--"/>
<hr/>	
<input type="button" value="Back to Login Page"/>	<input type="button" value="Save and Continue"/>

©Site is designed and hosted by Mugiwara



Login Page

UID	
<input type="text" value="00000"/>	
Password	
<input type="text"/>	
	<input type="radio"/>
Admin	<input type="radio"/>
Faculty	<input type="radio"/>
Student	
<hr/>	
<input type="button" value="Registration (Student)"/>	<input type="button" value="Login"/>

**Usage:**

Follow the steps below to use the system:

Connect the Raspberry Pi board, camera module, and monitor.

Connect to the internet/wifi/ Server. Either wirelessly or with the help of wires

Install the required software.

Install the hardware.

Boot the system. The team would assist in this process to create the Admin access, configuration, etc.

The system should work autonomously after this.

Troubleshooting:

If the system is not working correctly, follow the steps below:

Check the connections of the Raspberry Pi board, camera module, and monitor.

Check the connection.

Check if the required software is installed correctly.

Try restarting the system.

Contact the team.

9. Conclusion and Scope for Future work

9.1 Conclusion

In conclusion, the Attendance Management System using Facial Recognition is an innovative solution for organizations that want to automate their attendance tracking process. This system uses advanced facial recognition technology to accurately identify individuals and record their attendance in real-time. The system provides several benefits, including reducing manual errors, increasing efficiency, and providing detailed attendance reports. Additionally, the system ensures better security by preventing unauthorized access and eliminating the possibility of proxy attendance. With its user-friendly interface and easy-to-use features, the Attendance Management System using Facial Recognition is an excellent tool for any organization that values time, accuracy, and security. By adopting this system, organizations can enhance their attendance tracking process and improve their overall productivity.

The system utilizes advanced facial recognition technology to accurately and quickly identify employees and record their attendance. The system offers several benefits, including improved accuracy, reduced time and effort required for attendance management, and enhanced security.

Through the use of this system, organizations can streamline their attendance management processes and gain insights into employee attendance patterns, enabling them to make data-driven decisions. The system is easy to use and can be customized to meet the specific needs of an organization.

Overall, the Attendance Management System using Facial Recognition offers a modern and effective approach to attendance management, ensuring accurate attendance records, and reducing the burden of manual attendance management. With its numerous benefits, this system is a valuable tool for any organization looking to improve its attendance management processes.

9.2 Future Work

The same system can be utilized for several security applications where authentication is needed to access the privileges of the respective system. It can be used in recognizing guilty parties involved in unauthorized business. Face recognition algorithms can be improved with

respect to the utilization of resources so that the project can recognize a greater number of faces at a time which can make the system far better. Many variants of the project can be developed and utilized for home security and personal or organizational benefits. We can also trace a particular student in an organization quickly with the help of this system. Identification using Facial recognition is a powerful tool. It can be used as a means of authentication for not only attendance but a vast number of applications in almost every field. The system can be utilized for several security applications where authentication is needed to access the privileges of the respective system. Assume a system or a research facility containing confidential data that is only meant to be seen by certain people. Traditionally an id/password is issued for security and authentication however if someone gains access to these, the system is compromised and it is relatively easy to gain access to. Now, replacing these same id/passwords with facial recognition makes the authentication much more secure. In order to gain access to the system the hacker would have to steal a person's face, which as of yet is a rather difficult task. Hence this system can be used in authentication processes to identify personnel that can and can't access a certain resource or system. This however requires a more dynamic facial recognition algorithm, one that can differentiate between a picture and person. Moreover, the system would need to adapt to changes in a person's faces such as aging or facial hair etc. It would require a more sophisticated system with multiple cameras and different sensors such as infrared sensors. Every minute a child gets abducted, a car gets stolen, someone breaks the law, some of these are minor crimes but some of these are major crimes such as abduction of a child and is particularly time sensitive. In a similar security environment, the system can be deployed in public areas to help identify a person that has gone missing. The system would simply keep scanning every face that passes in front of it. When it finds a suspect, it can send an alert with the location and time to the authorities. This makes it much more likely to catch the suspect quickly with very low chances of mistaking the person for someone else. Similarly, it can be used to find a missing person. Even if a person, particularly a kid, gets lost, the kid can simply stand in front of a system camera and the camera will scan the person, send an alert and the person can be easily found and quickly. If the person that's lost hasn't been registered as lost yet, the system would just capture their face anyways and store it with location and time. When their face is searched in the database. The most recent image with time and location pops up, this would help the authorities to narrow down their search area and the person could be found quicker as the authorities would have some idea where the person could be. Another upcoming area are authenticating or registering emotions using facial recognition also known as Facial Emotion Recognition (FER) Facial Emotion Recognition (from real-time or static images) is

the process of mapping facial expressions to identify emotions such as disgust, joy, anger, surprise, fear, or sadness - or compound emotion such as sadly angry - on a human face with image processing software. Facial emotion detection's popularity comes from the vast areas of potential applications. It's different from facial recognition, whose goal is to identify a person, not an emotion. Face expression may be represented by geometric or appearance features, parameters extracted from transformed images such as eigenfaces, dynamic models, and 3D models. Integrating the system with home security systems can be an excellent addition to automation and security. Imagine the owner forgets the key or pin, they can simply open the door with their face. If the owner is expecting a delivery call from amazon, a system could be put in place that sends a picture of the delivery person to the home security system. The system then creates a one-time access pass for the person to enter the house, keep the item and exit. This could be timed for further security. The person has to exit in 2 mins or the system sounds alarms. This would also come into effect if the owner is expecting company. The owner can register them into the system and the door will open for them upon recognition. This can be configured as one time, multi time, multi time during a specific time period, etc.

10. References

1. Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on (Vol. 1, pp. I-511). IEEE.
2. NirmalyaKar, MrinalKantiDebbarma, AshimSaha, and DwijenRudra Pal, "Implementation of Automated Attendance System using Face Recognition", International Journal of Computer and Communication Engineering, Vol. 1, No. 2, July 2012.
3. Soundrapandiyan Rajkumar, J. Prakash, "Automated attendance using Raspberry pi", International Journal of Pharmacy and Technology, Sep 2016.
4. *An Integrated Approach to Software Engineering* by Pankaj Jalote, 3rd Ed.
5. https://www.academia.edu/52121795/Face_Recognition_Attendance_System_using_Raspberry_Pi - (PDF) Face Recognition Attendance System using Raspberry Pi | IJRASET Publication.
6. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3851342 - Face Recognition System Attendance System using Raspberry Pi by Evanjalín A.B., Christy D., Karthika N., Reshma R.S.: SSRN.
7. "Facial Recognition Technology (FERET)." National Institute of Standards and Technology. <https://www.nist.gov/programs-projects/facial-recognition-technology-feret>
8. "Facial Recognition Technology: Current Status and Potential Applications." National Academies of Sciences, Engineering, and Medicine. <https://www.nap.edu/catalog/24634/facial-recognition-technology-current-status-and-potential-applications>
9. "Facial Recognition Technology: A Survey of Policy and Implementation Issues." Congressional Research Service. <https://fas.org/sgp/crs/misc/R46534.pdf>
10. "Facial Recognition Technology: The Need for Regulation to Protect Privacy and Promote Accuracy." Centre for Democracy and Technology. <https://cdt.org/insight/facial-recognition-technology-the-need-for-regulation-to-protect-privacy-and-promote-accuracy/>
11. "Facial Recognition Technology: Key Considerations for Policymakers." Brookings Institution. <https://www.brookings.edu/research/facial-recognition-technology-key-considerations-for-policymakers/>

12. "Facial recognition software is not ready for use by law enforcement." MIT Technology Review. <https://www.technologyreview.com/2019/04/17/135237/facial-recognition-software-is-not-ready-for-use-by-law-enforcement/>
13. "The Global Expansion of AI Surveillance." Carnegie Endowment for International Peace. <https://carnegieendowment.org/2019/09/17/global-expansion-of-ai-surveillance-pub-79847>