DECLARATION

I undersigned here by declared that the project report **“ROI – Face Recognition based Attendance and Alert System”** submitted for partial fulfilment of the requirements for the award of degree of Master of Computer Applications of the APJ Abdul Kalam Technological University, Kerala. This submission represents my idea in my own words and, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact of source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University.

Place: Trivandrum DON J VELLATHOTTAM

Date: __/__/2022

ACKNOWLEDGEMENT

An endeavour over a long time can be successful only with advice and support of many well-wishers. I wish to place on record my profound indebtedness and gratitude to all those who have contributed directly or indirectly to make this project work a success.

At the very onset, I express my gratitude to God Almighty who sheltered me under his protective wings and showered on innumerable blessings throughout the period of this Master of Computer Application.

It is a great pleasure to express my sincere gratitude to Rev.Fr.Bejoy Arackal, Director and Dr.Beshiba Wilson, Principal Lourdes Matha College of Science and Technology for permitting to do this project with the fullest spirit.

I am highly obliged to Prof.Bismi K charleys Head of the Department of Computer Applications of Lourdes Matha College of Science and Technology, for being the source of inspiration throughout the course and for her valuable guidance.

With heart full of thanks, I would like to take up this opportunity to wish my Internal guide Prof.Bismi K charleys, Assistant Professor and all staffs of Department of Computer Applications for their endless support, encouragements and suggestions in various stages of the development of this project.

With immense love and gratitude, I thank every unknown member of numerous amounts of open-source communities for all the selfless works and contributions they’ve made. Without them and their help, I wouldn’t have made it here. Finally, I wish to express my sincere gratitude to all our friends who directly or indirectly contributed in this venture.

CONTENTS

Content Page No

ABSTRACT 1

CHAPTER 1 2

INTRODUCTION 2

1.1 General Introduction 3

1.2 Goal of the Project 3

CHAPTER 2 4

LITERATURE SURVEY 4

2.1 Study of Similar Work 5

2.1.1 Existing System 5

2.1.2 Drawbacks of Existing System 5

CHAPTER 3 6

OVERALL DESCRIPTION 6

3.1 Proposed System 7

3.2 Features of Proposed System 7

3.3 Functions of Proposed System 7

3.4 Requirements Specification 7

3.5 Feasibility Analysis 8

3.5.1 Technical Feasibility 8

3.5.2 Operational Feasibility 8

3.5.3 Economical Feasibility 8

3.5.4 Behavioural Feasibility 9

CHAPTER 4 10

OPERATING ENVIRONMENT 10

4.1 Hardware Requirements 11

4.2 Software Requirements 11

4.3 Tools and Platforms 11

4.3.1 Python 3.9 11

4.3.2 OpenCV 11

4.3.3 Face Recognition 11

4.3.4 MongoDB 12

4.3.5 Visual Studio 12

CHAPTER 5 13

DESIGN 13

5.1 System Design 14

5.2 DFD 15

5.3 Input Design 17

5.4 Output Design 18

5.5 Program Design 19

5.6 Database Design 19

5.7 Table Design 20

CHAPTER 6 21

FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS 21

6.1 Functional Requirements 22

6.2 Non-Functional Requirements 22

CHAPTER 7 23

TESTING 23

7.1 System Testing 24

7.2 Unit Testing 24

7.3 Integration Testing 24

7.4 Black Box Testing 24

7.5 Validation Testing 24

7.6 Output Testing 25

7.7 User Acceptance Testing 25

7.8 White Box Testing 25

7.9 Test Data and Results 26

7.10 Test Cases 26

CHAPTER 8 27

RESULTS AND DISCUSSION 27

8.1 Results 28

8.2 Screenshots 29

CHAPTER 9 32

CONCLUSION 32

9.1 System Implementation 33

9.2 Future Enhancement 34

9.3 Conclusion 34

CHAPTER 10 36

BIBLIOGRAPHY 36

10.1 Books 37

10.2 Websites 37

10.3 Journals and Publications 37

APPENDICES 38

GIT HISTORY 39

ABSTRACT

In this project, we propose a system that takes the attendance of students for classroom lecture. The proposed system marks the attendance naturally using facial identification and recognition. The system uses a camera that is mounted in the classroom, which is used to monitor the classroom and students. The feed from the camera is captured by the system and is broken down into frames. Using these frames, the faces of the children that are present in the class are recognized and their attendance is marked as present. Later the system alerts the parents of the absent children about the situation. The current system is of physical method. Teachers mark the attendance physically which causes a lot of time to be wasted. Also, it is very difficult in the current system to warn the parents about their children being absent in the classes. The proposed system ensures that both the attendance marking and the alerting process happens seamlessly and efficiently.

To make the system operable we use OpenCV and Face-Recognition libraries in python. There are three different main modules in this system. First module is used to register the faces of students and store their facial data in a pickle file. Second module uses the data from the said pickle file and compare it with the frames captured through OpenCV to recognize students and mark their attendance. Third module sends alert to parents of absent children.

The main features of this project are:

- Makes attendance system easier and efficient.
- Hassle free and time saving.
- Alert system helps increase safety of children.

CHAPTER 1

INTRODUCTION

1.1 GENERAL INTRODUCTION

In India average school time a day is eight hours and it consists of 8 periods each of duration 45 min or lower. On an average a teacher spends about 5 to 10 minutes each period to take attendance, which means nearly an hour a day is completely wasted for the sole purpose of taking attendance. By making the process automated we can save this lost time. The system also has an alert feature to make parents aware about the absence of their children in real-time. The alert system helps increase the safety and security of children.

Proposed system uses OpenCV, Face-Recognition, MongoDB and WhatsApp for functioning. System is divided into three different modules and handed out to users accordingly. Office, Teachers and Parents are the three categories of users. Even though students are part of it they really do not enjoy any user privileges

1.2 GOAL OF THE PROJECT

The main goal of the project is to minimalize the time wasted in the process of attendance taking and make the process more efficient. I am trying to create a system that benefits teachers by saving them time, benefits students by reducing the time wasted and thus giving them more time to learn, benefits the parents by letting them know if their children are safe or not.

CHAPTER 2

LITERATURE SURVEY

2.1 STUDY OF SIMILAR WORK

There are several systems for attendance purpose, like traditional methods of data, have drawbacks and hard to use that list, a biometric presence. There is a lack of human error in the system like fingerprint scan is not accepted because of wet conditions Fingers, dirty, very dry, or peeled fingers. The paper reviews relatable works. Attendance management system, NFC, face authority area, Microcomputers and Cloud area.

2.1.1 EXISTING SYSTEM

Existing online attendance systems uses mobile apps to record attendance. This takes the same time as of pen and paper-based attendance system. The teachers have to manually input the attendance status of each child into the app. Network issues can affect the functionality and they have to wait until the process is fully finished. In effect the existing system has the same problem as the pen and paper-based system. It just saves paper and has no other merits soever over the traditional system

2.1.2 DRAWBACKS OF EXISTING SYSTEM

- Wastage of time
- No alert facility
- Complicated to add a new student

CHAPTER 3
OVERALL DESCRIPTION

3.1 PROPOSED SYSTEM

The proposed system works in three different steps. First one is to add the facial data of the students, which can be done at the time of their admission. The stored data can be modified at any time and can add the faces of new students as they enrol. Second one is to mark attendance the teacher can run this action at any time as they please, so they can add latecomers. The final module sends the message to parents.

3.2 FEATURES OF PROPOSED SYSTEM

The proposed system makes the process more efficient and easier. It also helps you know whether the students are where they are supposed to be or not. The system is easy to use and efficient than the existing systems.

3.3 FUNCTIONS OF PROPOSED SYSTEM

There are basically four main functions for this proposed system

- Image Capturing
  Python OpenCV library is used to capture the faces of students. Facial data of all students are already stored along with their names in the system memory. Three different angles of every student’s face are captured earlier to make the dataset efficient.

- Processing the images
  The captured images are processed to obtain face encodings and locations, then they are compared with the existing data to recognize the students who are present in the class.

- Marking attendance
  The database is updated with the attendance status of students using the data received after the processing of images.

- Message alert
  Parents of absent children are notified via WhatsApp messages.

3.4 REQUIREMENTS SPECIFICATION

1. Accuracy: Accuracy in functioning and the nature of user-friendly should be maintained by the system.

2. Speed: The system must be capable of offering speed.

3. Flexible: The system should be flexible to new updates and patches in the future.

4. Good UX: The system should provide good user experience.

3.5 FEASIBILITY ANALYSIS

Feasibility study is a test of system proposed regarding its workability, impact on the organization, ability to meet the needs and effective use of resources. Thus, when a new project is proposed, it normally goes through a feasibility study before it is approved for development. A feasibility study is made to see if the project on completion will serve the purpose of the organization for the amount of work, effort and the time that is spend on it. Feasibility study lets the developer foresee the future of the project and its usefulness. All the projects are feasible given unlimited resources and infinite time. Unfortunately, the development of the computer-based system is more likely to be played by security of resources and difficulty delivery dates. Feasibility and risk analysis are related in many ways. If project risk is great, the feasibility of producing the quality software is reduced.

3.5.1 TECHNICAL FEASIBILITY

Technical Feasibility focuses on the existing computer system (hardware, software, etc) and to what extend it can support the proposed addition. For example, if the current computer is operating at 80 percent capacity, an arbitrary ceiling, then running another application could over load the system or require additional hardware. This involves financial considerations to accommodate technical enhancements. If the budget is a serious constraint, then the project is judged not feasible.

3.5.2 OPERATIONAL FEASIBILITY

The main problem faced during development of a new system is getting acceptance from the user. People are inherently resistant to changes and computers have been known to facilitate change. It is mainly related to human organizational and political aspects. The points to be considered are:

- What changes will be brought with the system?
- What new skills will be required? Do the existing staff members have these skills? If not, can they be trained due course of time?

Generally, project will not be rejected simply because of operational feasibility but such considerations are likely to critically affect the nature and scope of the eventual recommendations. This feasibility study is carried out by a small group of people who are familiar with information system techniques, who understand the parts of the business that are relevant to the project and are skilled in skilled analysis and design process.

3.5.3 ECONOMICAL FEASIBILITY

Economic Feasibility is the most frequently used method for evaluating the effectiveness of the candidate system. More commonly known as cost/benefit analysis, the procedure is to determine the benefits and savings that are expected from a candidate system and compare them with costs. If benefits outweigh costs, then the decision is made to design and implement the system. Otherwise, further justifications or alterations in the proposed system will have to be made if it is having a chance of being approved. This is an ongoing effort that improves in accuracy at each phase of the system life cycle.

3.5.4 BEHAVIOURAL FEASIBILITY

The proposed system is behaviourally feasible. Since it is behaviourally feasible because our system can be accessed and used even for a normal user. The user can easily understand working of the system. The proposed system is planned in such a way that every user can easily operate the system without difficulty. Hence it is behaviourally feasible. Behavioural Feasibility mainly includes how strong the reaction of staff will be towards the development of the proposed system that involves computer's use in their daily work. So resistant to change is identified.

CHAPTER 4

OPERATING ENVIRONMENT

4.1 HARDWARE REQUIREMENTS

1. Processor : Dual Core 1.60 GHz or higher.

2. Hard disk : 500 GB.

3. RAM : 4GB.

4. Monitor : 17” Color Monitor.

5. Mouse : Microsoft.

6. Keyboard : Microsoft multimedia keyboard.

4.2 SOFTWARE REQUIREMENTS

1. Operating System : Windows 8 or higher.

2. Framework : Microsoft .Net Framework.

3. Environment : Visual Studio 2012.

5. Language : Python 3.9, OpenCV.

6. Database : MongoDB.

7. Documentation : Microsoft Word 2010 or higher.

4.3 TOOLS AND PLATFORMS

4.3.1 Python 3.9

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

4.3.2 Open-CV

Open CV (Open-Source Computer Vision Library) is a library of programming functions mainly aimed at real-time computer vision. Originally developed by Intel, it was later supported by Willow Garage then Itseez (which was later acquired by Intel). The library is cross-platform and free for use under the open-source Apache 2 License. Starting with 2011, OpenCV features GPU acceleration for real-time operations.

4.3.3 Face Recognition

Recognize and manipulate faces from Python or from the command line with the world’s simplest face recognition library; built using dlib’s state-of-the-art face recognition. Built with deep learning, the model has an accuracy of 99.38% on the Labelled Faces in the

Wild benchmark. This also provides a simple face recognition command line tool that lets you do face recognition on a folder of images from the command line.

4.3.4 MongoDB

MongoDB is a source-available cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas. MongoDB is developed by MongoDB Inc. and licensed under the Server-Side Public License (SSPL) which is deemed non-free by several distributions.

4.3.5 Visual Studio

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code. A fully-featured, extensible, free IDE for creating modern applications for Android, iOS, Windows, as well as web applications and cloud services. Visual Studio does not support any programming language, solution or tool intrinsically; instead, it allows the plugging of functionality coded as a VSPackage. When installed, the functionality is available as a Service. The IDE provides three services: SVsSolution, which provides the ability to enumerate projects and solutions; SVsUIShell, which provides windowing and UI functionality (including tabs, toolbars, and tool windows); and SVsShell, which deals with registration of VSPackages. In addition, the IDE is also responsible for coordinating and enabling communication between services. All editors, designers, project types and other tools are implemented as VSPackages. Visual Studio uses COM to access the VSPackages. The Visual Studio SDK also includes the Managed Package Framework (MPF), which is a set of managed wrappers around the COM-interfaces that allow the Packages to be written in any CLI compliant language. However, MPF does not provide all the functionality exposed by the Visual Studio COM interfaces. The services can then be consumed for creation of other packages, which add functionality to the Visual Studio IDE.

CHAPTER 5
DESIGN

5.1 SYSTEM DESIGN

System Design involves translating system requirements and conceptual design into technical specifications and general flow of processing. After the system requirements have been identified, information has been gathered to verify the problem and after evaluating the existing system, a new system is proposed.

System Design is the process of planning of new system or to replace or complement an existing system .It must be thoroughly understood about the old system and determine how computers can be used to make its operations more effective.

System design sits at technical the kernel of system development. Once system requirements have been analysed and specified system design is the first of the technical activities-design, code generation and test- that required build and verifying the software. System design is the most creative and challenging phases of the system life cycle. The term design describes the final system and the process by which it is to be developed.

System design is the high-level strategy for solving the problem and building a solution. System design includes decisions about the organization of the system into subsystems, the allocation of subsystems to hardware and software components and major conceptual and policy decision that forms the framework for detailed design.

There are two levels of system design:

- Logical design.
- Physical design.

In the logical design, the designer produces a specification of the major features of the system which meets the objectives. The delivered product of logical design includes current requirements of the following system components:

- Input design.
- Output design.
- Database design.

Physical design takes this logical design blue print and produces the program software, files and a working system. Design specifications instruct programmers about what the system should do. The programmers in turn write the programs that accept input from users, process data, produce reports, and store data in files.

Structured design is a data flow-based methodology that partitions a program into a hierarchy of modules organized top-down manner with details at the bottom. 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 independently of the physical components.

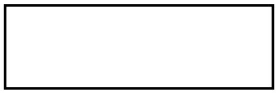
5.2 DATA FLOW DIAGRAM (DFD)

A data flow diagram is a graphical technique that depicts information flow and transforms that are applied as data move from input to output. The DFD is also known as Data Flow Graph or Bubble Chart. The DFD is used to represent increasing information flow and functional details. Also, DFD can be stated as the starting point of the design phase that functionally decomposes the requirements specifications down to the lowest level of detail. A Level 0 also called a fundamental system model or a context level DFD that represent the entire software elements as a single bubble with input and output data indicated by incoming and outgoing arrows, respectively. Additional process and information flow parts are represented in the next level, i.e., level 1 DFD. Each of the processes represented at level 1 are sub functions of overall system depicted in the context model. Any processes that are complex in level 1 will be further represented into sub functions in the next level, i.e., level 2. Data flow diagram is a means of representing a system at any level of detail with a graphic network of symbols showing data flows, data stores, data processes and data sources. The purpose of data flow diagram is to provide a semantic bridge between users and system developers. The diagram is the basis of structured system analysis. A DFD describes what data flows rather than how they are processed, so it does not depend on hardware, software, data structure or file organization.

Components of Data Flow Diagram

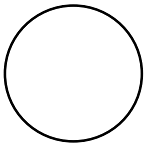
There are four symbols that are used in the drawing of Data Flow Diagrams:

- Entities



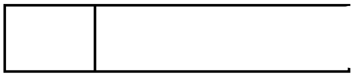
External entities represent the sources of data that enter the system or the recipients of data that leave the system.

- Process



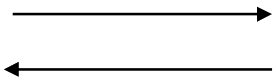
Processes represent activities in which data is manipulated by being stored or retrieved or transformed in some way. A circle represents it. The process will show the data transformation or change.

- Databases



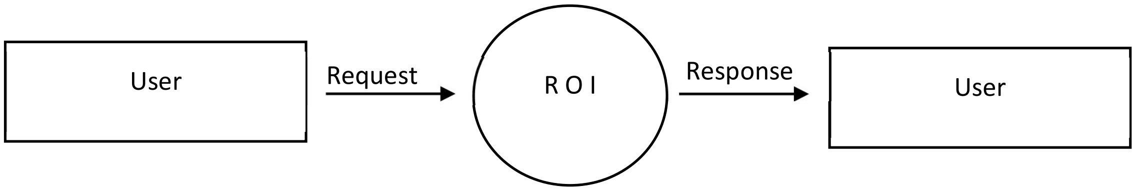
Databases represent storage of data within the system.

- Data Flow

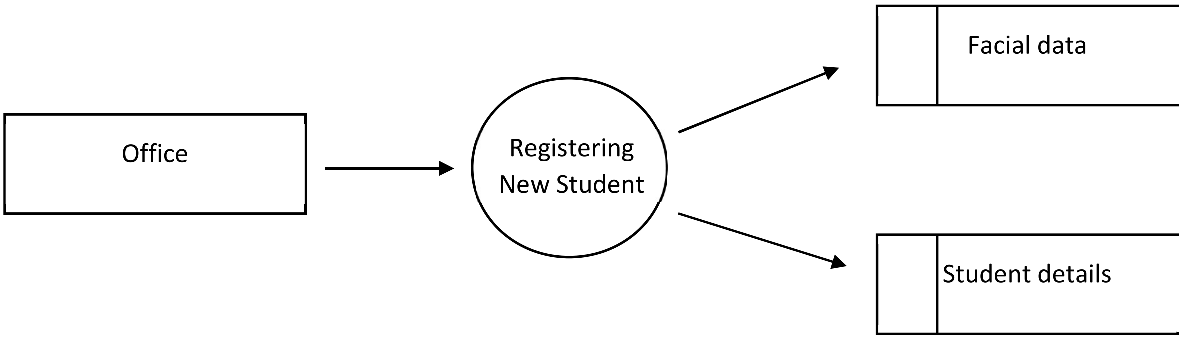


A data flow shows the flow of information from its source to its destination. A line.

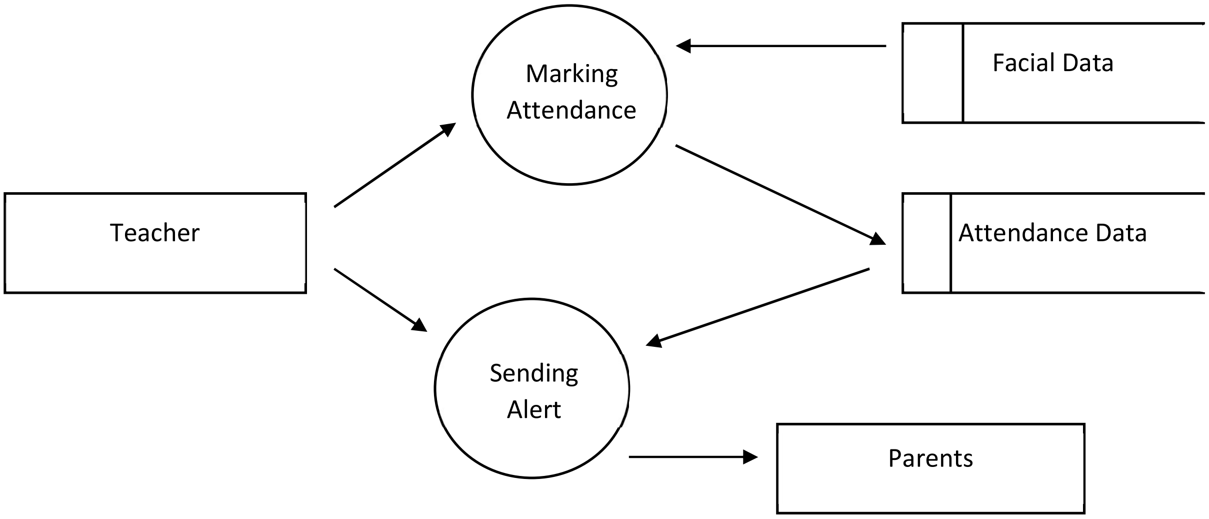
Level 0 Context Diagram



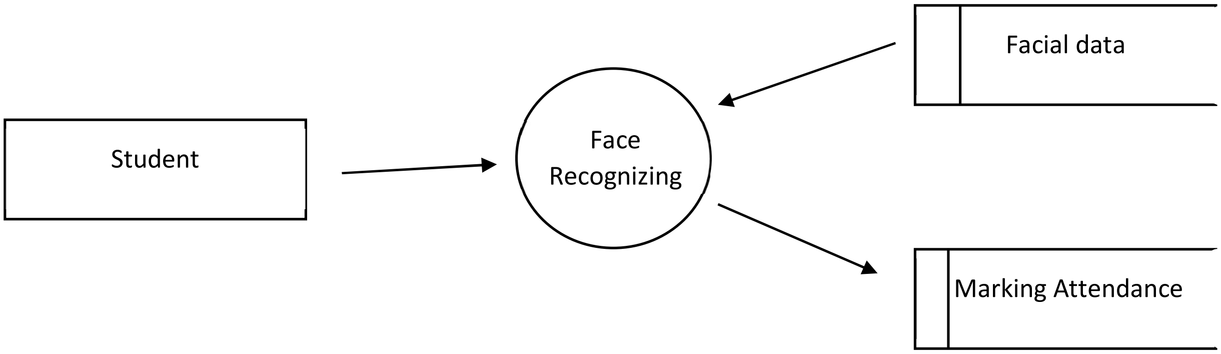
Level 1 Office



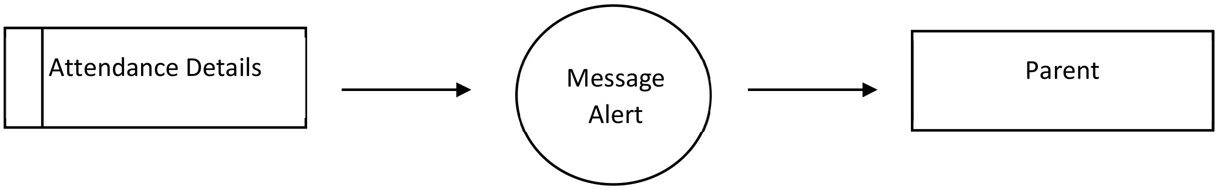
Level 1 Teacher



Level 1 Student



Level 1 Parent



5.3 INPUT DESIGN

The input design is the process of converting the user-oriented inputs in to the computer-based format. The goal of designing input data is to make automation as easy and free from errors as possible. The input design requirements such as user friendliness, consistent format and interactive dialogue for giving the right message and help for the user at right time are also considered for the development of the project.

The following points should be considered while designing the input:

- What data to input?
- What medium to use?
- How the data should be arranged or coded?
- The dialogue to guide users in providing input.
- Data items and transactions needing validation to detect errors.
- Methods for performing input validation and steps to follow when errors occur.

Inaccurate input data is the most common cause of error in processing data. Errors entered by the data entry operators can be controlled by the input design. The arrangement of messages as well as placement of data, headings and titles on display screens or source document is also a part of input design. The design of input also includes specifying the means by which end user and system operators direct the system what action to take. The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps that are necessary to put transaction data into a usable form for processing data entry.

The user interface design is very important for any application. The interface design defines how the software communicates within itself, to system that interpreted with it and with human who use it. The interface design is very good; the user will fall into an interactive software application.

Input design is the process of converting user-oriented inputs to a computer-based format. The data is fed into the system using simple interactive forms. The forms have been supplied with messages so that user can enter data without facing any difficulty. The data is validated wherever it requires in the project. This ensures that only the correct data have been incorporated into the system. Inaccurate processing of data is the most common cause of errors in data processing. Errors entered by data entry operators can be controlled by correct input design. This type of input design allows user to input only the required data into the processing units and also these input from check for validation of the input values, thus preventing errors.

The input design is made into user-friendly atmosphere where the user can perform the daily routine work without any ones help. The user-friendly environment created by the input design helps the end user to use the software in a more flexible way and even the wrong entries by the user is correctly pointed out to the user.

The goal of designing input data is to make the automation easy and free from errors as possible. For providing a good input design for the application, easy data input and selection features are adopted.

5.4 OUTPUT DESIGN

Output generally refers to the results and information that are generated by the system. When designing output, system analyst must accomplish the following:

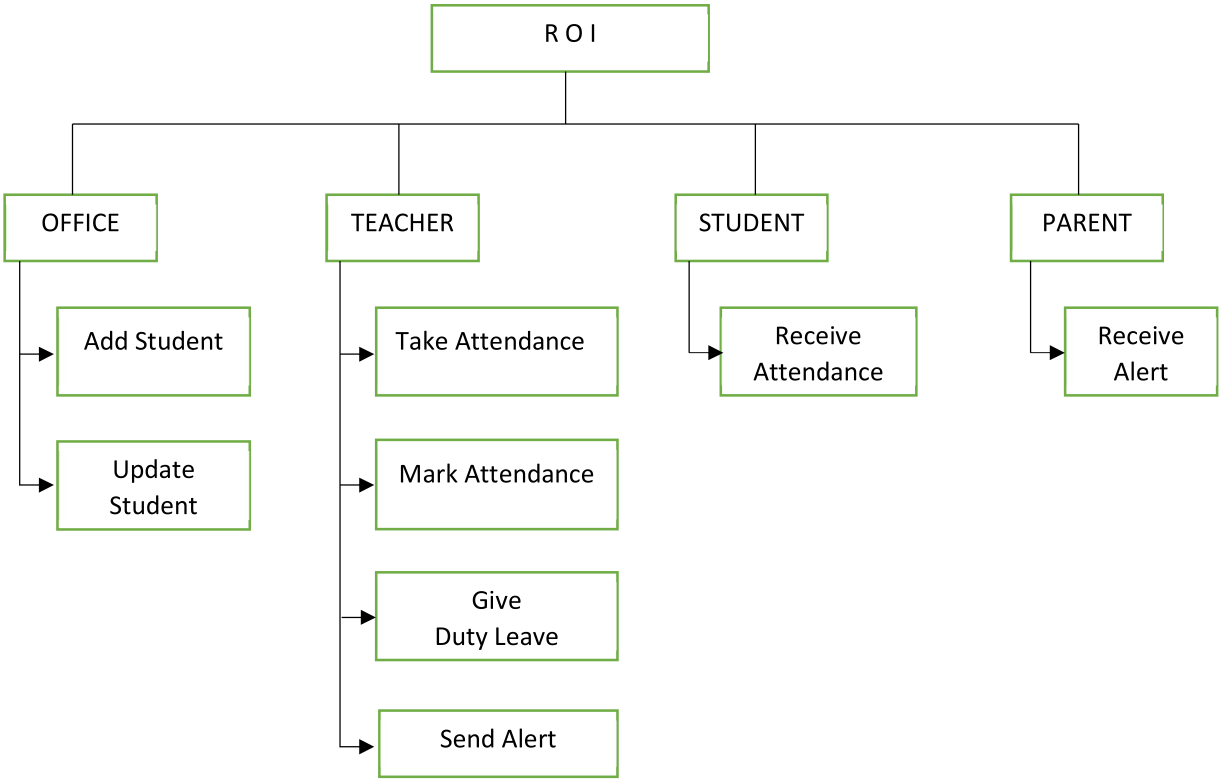
- Determine what information to present.
- Decide whether to display, print the information and select the output medium.
- Arrange the presentation of information in an acceptable format.
- Decide how to distribute the output to intended recipients.

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any systems, results of processing are communicated to the user and to other systems through outputs. In the output design, it is determined how the information is to be displayed for immediate need.

The major idea of output is to convey information so its layout and design need careful consideration. Efficient, intelligible output design improves the system relationship with the users and help in making decisions. The output designs decide how well the implementation of the system has been useful to the user. The output design should be understandable to the user and it must offer great convenience. The one who look into the reports or output will get the impression of how well the system performs.

The objective of the output design is to convey the information of all the past activities, current status and emphasize important events. The output generally refers to the results and information that is generated from the system. Outputs from the computers are required primarily to communicate the result of processing to the users. They are also used to provide a permanent copy of these results for later consideration.

5.5 PROGRAM DESIGN



5.6 DATABASE DESIGN

The most important aspect of building software systems is database design. The highest level in the hierarchy is the database. It is a set of inter-related files for real time processing. It contains the necessary data for problem solving and can be used by several users accessing data concurrently. The general objective of database design is to make the data access easy, inexpensive and flexible to the user.

Database design is used to define and then specify the structure of business used in the client/server system. A business object is nothing but information that is visible to the users of the system. The database must be a normalized one.

Database management system (DBMS) allows the data to be protected and organized separately from other resources like hardware, software and programs. DBMS is a software package, which contains components that are not found in other data management packages. The significance of DBMS is the separation of data as seen by the programs and data as stored on the direct access storage devices, i.e., the difference between logical and physical data.

In my project, I have used MongoDB as the database to implement the data store part. The most important part in the database design is the identification of tables to be used.

5.7 TABLE DESIGN

Database design activity deals with the design of the physical database. A key is to determine how the access paths are to be implemented. A physical path is derived from a logical path. Pointers, chains or other mechanisms may implement it.

Unlike conventional databases MongoDB is not an SQL database, it is a NoSQL database. Hence, the data is not stored in table like structures which eliminates the need of multiple tables. MongoDB uses collections of objects of same type and stores data in each object's attributes. Here each student is an object of the collection attendance data and their data is stored in their own attributes.

The attributes of student and their types are:

Table name: Student Attributes

Field	**Type**
Student ID	Integer(_id field)
Student Name	String
Student Mobile	Integer
Attendance Status	Object

CHAPTER 6

FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

6.1 FUNCTIONAL REQUIREMENTS

The functional requirements represent the intended behaviour of the system. The proposed system consists of

- Register students
- Webcam access
- Face detection
- Generate token
- Save file
- Face video detection
- Harr algorithm implementation
- Detect face
- Match face and identify name
- Mark attendance
- Send alert message

6.2 NON-FUNCTIONAL REQUIREMENTS

- Accuracy: Accuracy in functioning and the nature of user-friendly should be maintained by the system.
- Speed: The system must be capable of offering speed.
- Graphical User Interface: This proposed system is highly user friendly by generating GUI using Python Tkinter.

CHAPTER 7
TESTING

7.1 SYSTEM TESTING

System Testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently as expected before live operation commences. It certifies that the whole set of program hang together. System testing requires a test plan that consists of several keys, activities and steps to run program, string, system and user acceptance testing. The implementation of newly designed package is important in adopting a successful new system.

Testing Objectives

- Testing is the process of correcting a program with intend of finding an error.
- A good test is one that has a high probability of finding a yet undiscovered error.
- A successful test is one that uncovers an undiscovered error.

There are different types of testing methods available:

7.2 UNIT TESTING

In this testing we test each module individually and integrate the overall system. Unit testing focuses verification efforts on the smaller unit of software design in the module. This is also known as 'module' testing. The modules of the system are tested separately. The testing is carried out during programming stage itself. In this testing step each module is found to work satisfactory as regard to the expected output from the module. There are some validation checks for verifying the data input given by the user. It is very easy to find error and debug the system.

7.3 INTEGRATION TESTING

Data can be lost across an interface; one module can have an adverse effect on the other sub functions when combined by May not produce the desired major functions. Integrated testing is the systematic testing for constructing the uncover errors within the interface. This testing was done with sample data. The need for integrated test is to find the overall system performance.

7.4 BLACK BOX TESTING

This testing attempts to find errors in the following areas or categories: Incorrect or missing functions, interface errors, errors in data structures, external database access, performance errors and initialization and termination errors.

7.5 VALIDATION TESTING

At the culmination of Black Box testing, software is completely assembled as a package, interface errors have been uncovered and corrected and final series of software tests, validation tests begins. Validation testing can be defined in many ways but a simple definition is that validation succeeds when the software functions in a manner that can be reasonably accepted by the customer.

After validation test has been conducted one of the two possible conditions exists.

- The function or performance characteristics confirm to specification and are accepted.
- A deviation from specification is uncovered and a deficiency list is created.

7.6 OUTPUT TESTING

After performing the validation testing, the next step is output testing of the proposed system since no system could be useful if it doesn't produce the required data in the specific format. The output displayed or generated by the system under consideration is tested by, asking the user about the format displayed. The output format on the screen is found to be correct as the format was designed in the system according to the user needs. Hence the output testing doesn't result in any correction of the system

7.7 USER ACCEPTANCE TESTING

User acceptance of the system is the key factor for the success of the system. The system under consideration is tested for user acceptance by constantly keeping in touch with prospective system at the time of developing and making change wherever required. This is done with regard to the following points:

- Output Screen design.
- Input Screen design.
- Menu driven system.

7.8 WHITE BOX TESTING

White box testing is a testing case design method that uses the control structure of the procedural design to derive the test cases. The entire independent path in a module is exercised at least once. All the logical decisions are exercised at least once. Executing all the loops at boundaries and within their operational bounds exercise internal data structure to ensure their validity.

In our project testing was conducted at every step. Initially each module was tested separately to check whether they gave the desired output for the given input. The forms used to enter data by user were validated and appropriate error messages were displayed if incorrect data was entered. Once the data was entered correctly, the processing was done and testing was done to check whether the correct output was obtained. Once the test cases were conducted successfully for each module, the modules were integrated together as a single system. After integration, the test cases were again applied to check whether the entire system as a whole produced the desired output. At times, the test cases failed and the shortcomings were noted down and appropriate corrections were done. Once the integration testing was performed correctly, output testing was done and it did not result in any change or correction in the system. Black box testing and white box testing was also conducted successfully. All the loops, decisions, relations were executed at least once before giving it to the users for testing. In black box testing, it was checked whether the data in the proper format was stored in the database or not. Also, it was checked whether the interfaces were working properly or not. On successful completion of these tests, the system was then given to undergo user acceptance testing where the users entered test data to check whether the correct output was obtained. The users were satisfied with the output and thus the testing phase was completed successfully.

7.9 TEST DATA AND RESULTS

The primary goal of software implementation is the production of source code that is easy to read and understand. Clarification of source code helps in easier debugging, testing and modification. Source code clarification is enhanced by structural coding techniques, by good coding style, by appropriate supporting documents, by good internal comments and by the features provided in the modern programming language.

In our implementation phase, source code contains both global and formal variables. It contains predefined functions as well as the user defined functions. The result of the new system is compared with old system and supposes if the result is wrong the error must be debugged.

After the acceptance of the system by the user, the existing system should be replaced by this system. Any user handles this package very easily. It does not require any intensive training for the user. Procedures and functions involved in this system are very simple that anyone can understand and correspondingly act to the system with no difficulty.

7.10 TEST CASES

Test Case No	Test Case Name	Test Case Step	Expected Result	Status	Defects
1	Adding students	Adding new students	Facial data is recorded and database is updated	Pass	Nil
2	Taking Attendance	By using real time camera footage Students are detected	Students are recognized using camera with previously stored facial data	Pass	Nil
3	Marking Attendance	Recorded attendance is updated in database	Attendance record is updated in database after giving duty leaves	Pass	Nil
4	Sending Alert	Alert message is sent	Parents of absent children gets alert messages	Pass	Nil

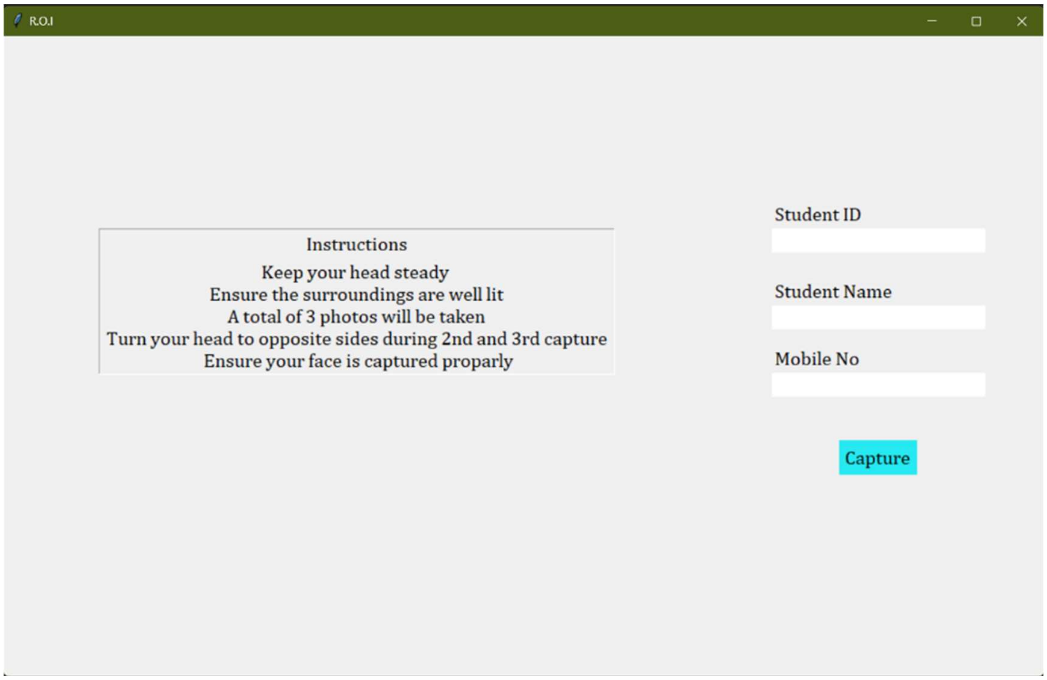
CHAPTER 8

RESULTS AND DISCUSSION

8.1 RESULTS

The system was tested on various conditions which include illumination, head movements, expressions, the distance of students from the camera. The system stands up to the expectations even when the image contains faces with beards and spectacles and without beard and spectacles. proposed system evinced to be magnificent to recognize faces having masks.

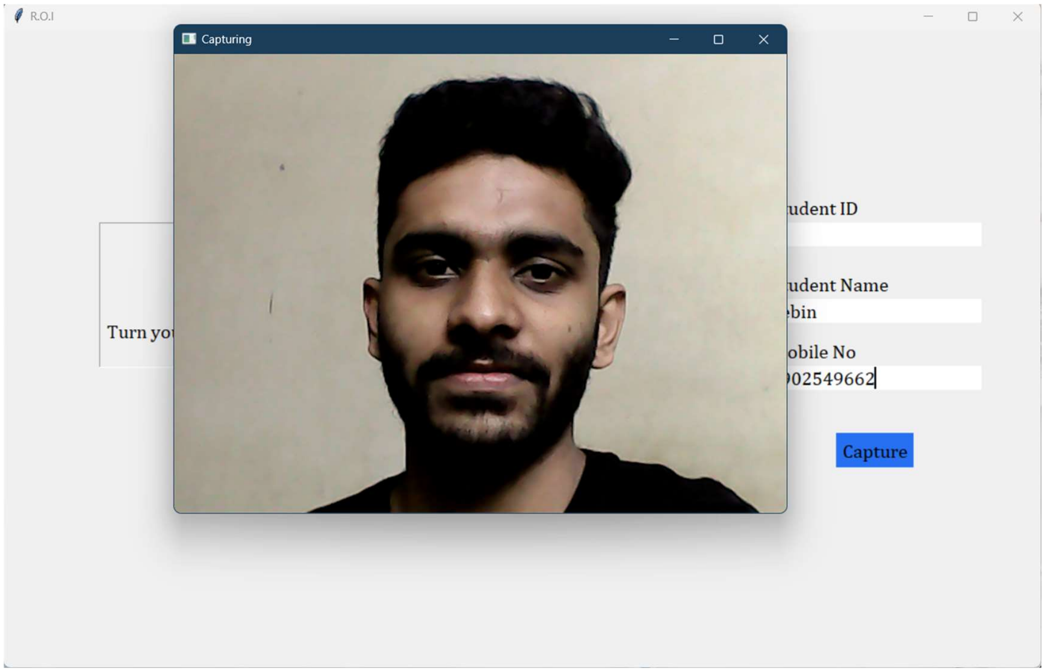
8.2 SCREENSHOTS



Instructions

Keep your head steady
Ensure the surroundings are well lit
A total of 3 photos will be taken
Turn your head to opposite sides during 2nd and 3rd capture
Ensure your face is captured proparly

Image 1: Office UI



udent Name
bin

obile No
02549662

Image 2: Adding Student

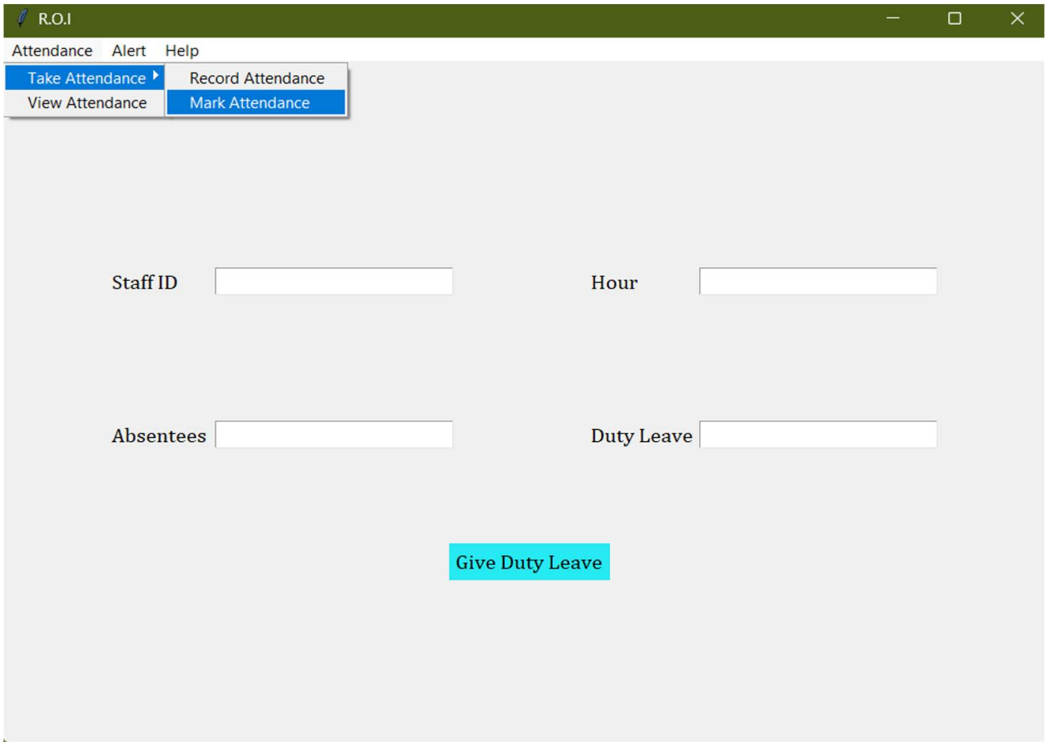


Image 3: Teacher UI

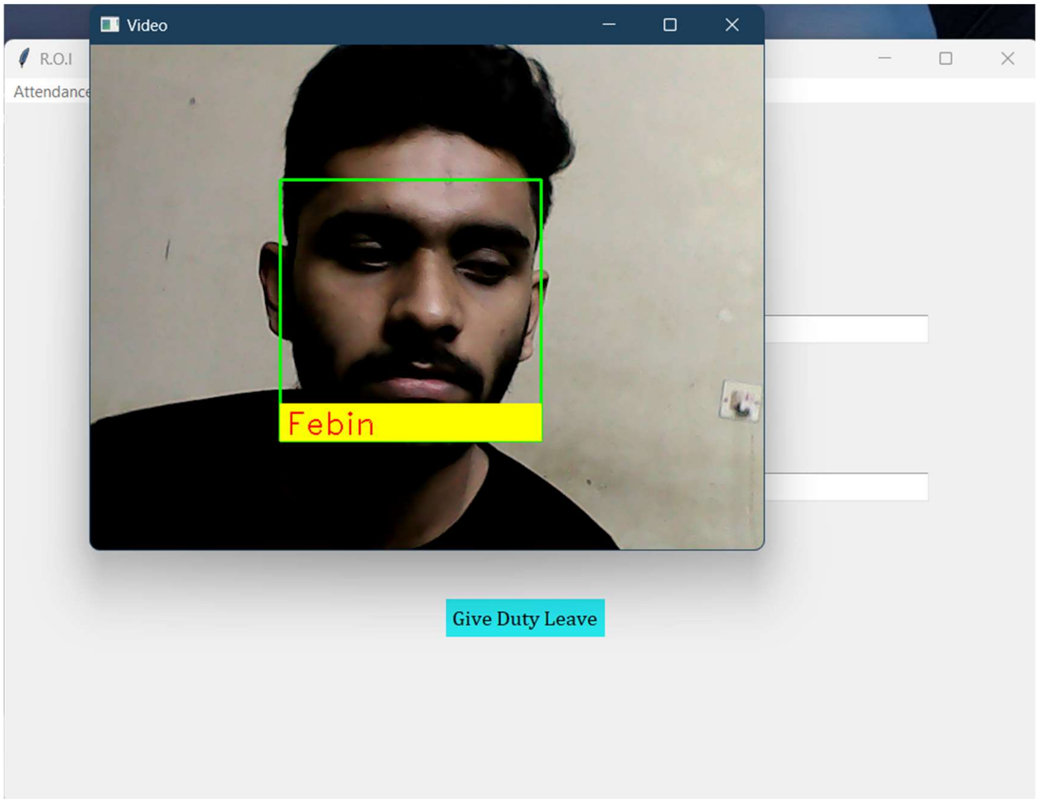
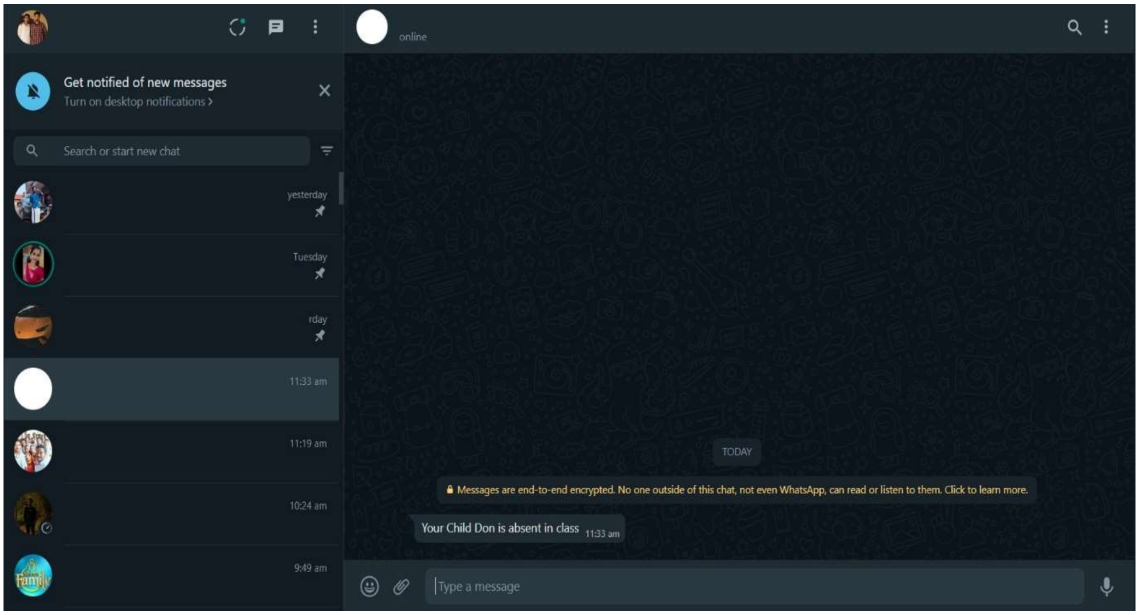


Image 4: Recording Attendance



Get notified of new messages
Turn on desktop notifications ›

TODAY

Messages are end-to-end encrypted. No one outside of this chat, not even WhatsApp, can read or listen to them. Click to learn more.

Your Child Don is absent in class 11:33 am

Image 5: Alert Message

CHAPTER 9

CONCLUSION

9.1 SYSTEM IMPLEMENTATION

Implementation is an activity that is contained throughout the development phase. It is the process of bringing a developed system into operational use and turning it over to the user. The new system and its components are to be tested in a structured and planned manner. A successful system should be delivered and users should have the confidence that the system should have work efficiently and effectively. The more complex system being implemented, the more will be the system analysis and design effort required just for implementation.

Implementation is the stage of the system when the theoretical design is turned into working system. The plan contains an overview of the system, a brief description of the major tasks involved in the implementation, the overall resources needed to support the implementation effort, and any site implementation requirements. The plan is developed during the design phase and is updated during the Development phase. The outline shows the implementation plan.

There are three types of implementations:

a. Implementation of a computer system for replacing the manual system. The problem encountered are converting files, training users, create accurate files.

b. Implementation of new computer system to replacing an existing one. This is usually a difficult conversion. If not properly planned, there can be many problems. Some larger computer systems have taken as long as a year to convert.

c. Implementation of modified application to replace an existing one using the same computer. This type of conversion relatively easy to handle, provided there are no major changes in file.

Implementation Plan Preparation

The implementation plan begins with preparing a plan for the implementation of the system. In this plan, discussion has been made regarding the equipment, resources and how to test the activities. The following information is acquired.

- What the task will accomplish?
- Resources required to accomplish the task.
- Key person(s) responsible for the task.

Examples of the major tasks are the following.

- Providing overall planning and coordination for the implementation.
- Providing appropriate training for the personnel.
- Ensuring that all the manual are applicable to the implementation.
- Providing all the technical requirements.

- Performing site surveys before implementation.
- Ensuring that all the prerequisites have been fulfilled before the implementation date.
- Providing personnel for the implementation team.
- Acquiring special software or hardware.
- Performing data conversion before loading data into the system.
- Preparing site facilities for implementation.

Implementation of Proposed System

After having user acceptance for the system developed, the implementation phase begins. Implementation is the stage of project during which theory is tuned into practice. During this phase, all the programs of the system are loaded into the user's computer. After loading the system training of the user starts. Such as type of training includes:

1. How to execute the package?
2. How to enter the data?
3. How to process the data (processing details)?
4. How to takeout the report?

The following two strategies are followed for running the system.

Parallel Run: In such run for a certain defined period, both the systems thereafter computerized and manual are executed in parallel. This strategy is helpful because of the following:

1. Manual result can be compared with the result of computerized system. For the care of demonstration of the success of this system, it was implemented with successfully running; manual systems and results are verified.
2. Failure of a computerized system at an early stage, do not affect the work of the organization, because the manual system continues to work as it used to do.

Pilot Run: In this type of run, some parts of the new system are installed first and executed successfully for the considerable time period. When the results are found satisfactory, only then the other parts are implemented. This strategy builds the confidence and errors are traced easily.

9.2 FUTURE ENHANCEMENT

In future Attendance monitoring system can develop using artificial intelligence. A database containing the facial images of the students in a particular class is constructed. Knowledge gained using Convolutional Neural Network (CNN) is reused in a perfect manner using transfer learning. In future this system is designed to improve the students' engagement time inside the classroom, to communicate to the parents frequently, to avoid proxy attendance and to generate detailed reports for future reference.

9.3 CONCLUSION

By studying various papers, we concluded that by using the color feature extraction it becomes easier to describe the image. Color feature extraction is one of the most important feature extraction techniques out of color, shape and texture. We have also used clustering algorithm to generate cluster of the similar images which would be useful for searching the similar image to that of query image very fast. By combining various algorithms such as feature extraction (Average RGB), for grouping similar images (clustering) and then finally using indexing can make the output of the project more accurate. The output of the project so obtained is not 100% accurate.

Even though the system is built as semi-automatic, security system in Client/Server architecture it can be easily expanded to a fully automatic security system. The system can be modified into a vigilante software. The face recognition algorithm can be used to locate and record the locations at which a person is spotted and the data can be used to predict when and where would someone be at. Even though ROI is developed as an attendance and alert system in future it can transform into an AI based vigilante.

CHAPTER 10
BIBLIOGRAPHY

10.1 BOOKS

- Claudia Schmuckli, “Beyond the Uncanny Valley: Being Human in the Age of AI”, Cameron Publications.
- Himanshu Singh, “Practical Machine Learning and Image Processing”, Apress Publications.
- Navin Kumar Manaswi, “Deep Learning with Applications Using Python”, Apress Publications.

10.2 WEBSITES

- scholar.google.com
- stackoverflow.com
- geeksforgeeks.org
- javapoint.com
- w3schools.com
- quora.com

10.3 JOURNALS AND PUBLICATIONS

- “Face Detection and Face Recognition in Python Programming Language”, Boris Kuster.
- “Face Detection and Recognition using OpenCV and Python”, Tejashree Dhawle.

APPENDICES

1. List of Tables

Table No	Table Name	Page No
01	Student Attributes	20
02	Test Cases	26

2. List of Figures

Figure No	Table Name	Page No
1	Components of DFD	15
2	Level 0 Context Diagram	16
3	Level 1 Office	16
4	Level 1 Teacher	16
5	Level 1 Student	17
6	Level 1 Parent	17
7	Program Design	19

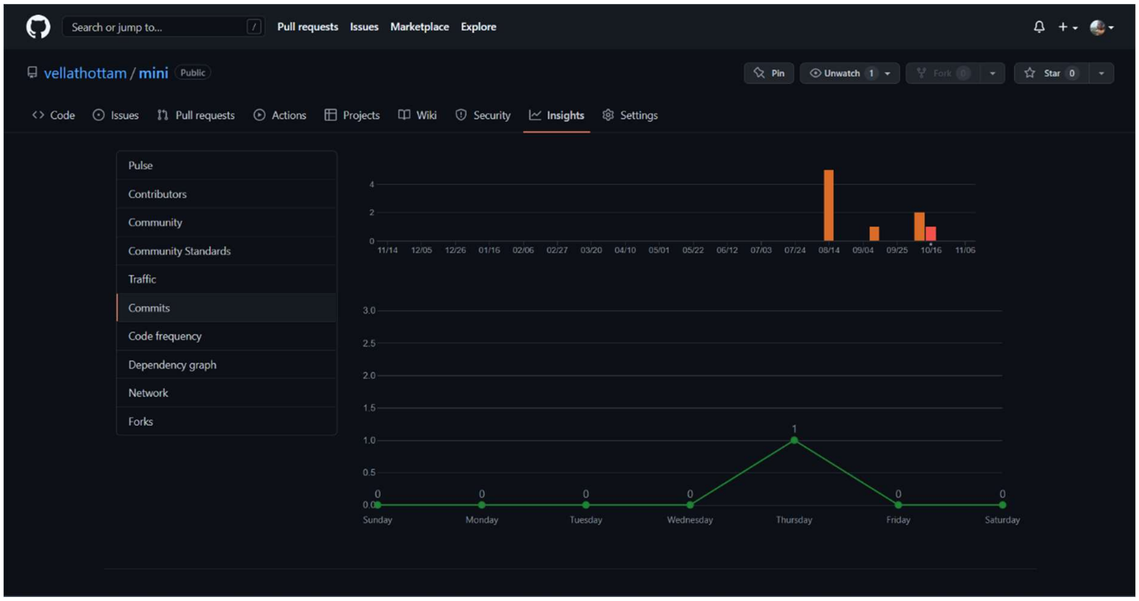
3. List of Images

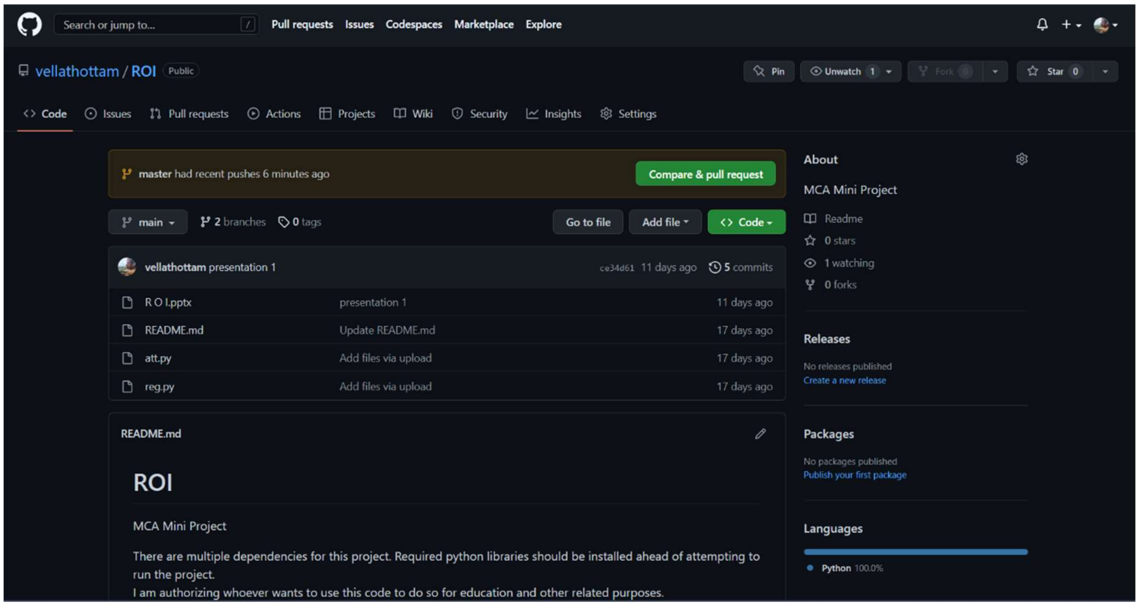
Image No	Image Name	Page No
1	Office UI	29
2	Adding Student	29
3	Teacher UI	30
4	Recording Attendance	30
5	Alert Message	31

Abbreviation and Notation

DFD (Data Flow Diagram): is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

GIT HISTORY





master had recent pushes 6 minutes ago

MCA Mini Project

vellathottam presentation 1 ce34d61 11 days ago 5 commits

R O I.pptx	presentation 1	11 days ago
README.md	Update README.md	17 days ago
att.py	Add files via upload	17 days ago
reg.py	Add files via upload	17 days ago

README.md

ROI

MCA Mini Project

There are multiple dependencies for this project. Required python libraries should be installed ahead of attempting to run the project.
I am authorizing whoever wants to use this code to do so for education and other related purposes.