

**FINAL PROJECT REPORT**

**ON**

**FACIAL RECOGNINTION ATTENDANCE SYSTEM**

Submitted in partial

fulfilment of requirement for the award of the

degree of

**BACHELOR IN COMPUTER APPLICATIONS**

DEPARTMENT OF COMPUTER APPLICATIONS.

SESSION :2023

-

20

24

**SUBMITTED BY:**

YASH GUPTA

22201020082

**SUPERVISED BY:**

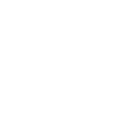
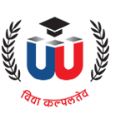
LOKESH MEENA

IBM TRAINER

CSE

**UNITED UNIVERSITY,**

**RAWAPUR, PRAYAGRAJ, UTTAR PRADESH**



**B**

**C**

**A**

20

24



**Facial Recognition Attendance System**

**Attendance System**

YASH GUPTA

22201020082

# Project Certificate

This is to certify that the project report entitled Facial Recognition Attendance System submitted to United University, in partial fulfilment of the requirement for the award of the degree of BACHELOR OF COMPUTER APPLICATIONS (BCA), is original work carried out by myself Mr. YASH GUPTA with enrolment no. UU222010232 Under the Supervision of Mr Lokesh Meena. The matter embodied in this project is genuine work done by myself and has not been submitted whether to this University or to any other University / Institute for the fulfilment of the requirement of any course of study.

Date:

Name & Signature of the Student

Yash Gupta

Contact Details (Email, Phone & Address)

[guptayash5911@gmail.com](mailto:guptayash5911@gmail.com)  
 +91 6392 157 582

837/712

Colonelganj Prayagraj

211002, Uttar Pradesh

Verified by the Supervisor

Lokesh Meena

**Acknowledgement**

This Major Project is the result of contribution of many minds. I would like to acknowledge and thank my project guide Lokesh Meena for his valuable support and guidance. He guided me through the process from conception and till the completion of this project. I would also like to thanks my institute director and my all my faculties I thank to lab staff members and other non-teaching members.

I am very thankful for the open-handed support extended by many people. While no list would be complete, it is my pleasure to acknowledge the assistance of my friends who provided encouragement, knowledge and constructive suggestions.

Signature of the Student

Yash Gupta

22201020082

UU222010232

# 

# Self-Certificate

This is to certify that the Major Project report entitled “Facial Recognition Attendance System” is done by me, and it is authentic work carried out for the partial fulfilment of the requirements for the award of the degree of Bachelor of Computer Application (BCA) under the guidance of **Lokesh Meena**. The matter and software embody in this project has not been submitted earlier for award of any degree or diploma to the best of my knowledge and believes.

Signature of Student

Yash Gupta

22201020082

UU222010232

# Certificate from Project Guide

This is to certify that this Major Project entitled "Facial Recognition Attendance System" submitted in partial fulfilment of the requirements for the award of the degree of Bachelor of Computer Application (BCA) in session 2023-2024 to the United University, Prayagraj, done by Yash Gupta is an authentic work carried out by them at “Department of Computer Applications” under my guidance. The matter and software embodied in this project work has not been submitted earlier for the award of any degree or diploma to the best of my knowledge and belief.

Name & Signature of the Supervisor/s

Date: ………………..

|  |  |  |
| --- | --- | --- |
| **S. NO.** | **Topic** | **Page No.** |
| 1. | Project Certificate | 1 |
| 2. | Acknowledgement | 2 |
| 3. | Self-Certificate | 3 |
| 4. | Certificate from Project Guide | 4 |
| 5. | Index | 5 |
| 6. | Introduction | 6 |
| 7. | System analysis | 7 |
| 8. | Feasibility study | 8 |
| 9. | Software and hardware requirement specifications | 9 |
| 10. | System design | 11 |
| 11. | Coding | 13 |
| 12. | Validation Checks | 14 |
| 13. | Implementation and maintenance | 16 |
| 14. | Testing | 18 |
| 15. | Future scope and further enhancement of the project | 20 |
| 16. | Screen Shot | 22 |
| 17. | Bibliography/ references | 26 |
| 18. | Conclusion | 25 |

**Introduction**

Innovation and efficiency boost developed with the beginning of the high-tech area across many industries. In the current educational activity and organization controlling, attendance tracking method of yesterday usually has appeared to be laborious and has been tainted with errors. Unlike traditional attendance systems which are prone to human errors and inefficiencies due to the use of paper-based registers or electronic spreadsheets, attendance management software allows for error-free and efficient processes.  
  
  
In the face of these hurdles, advanced techniques such as facial recognition, computer vision and database management systems are seen as potentially state-of-the-art solutions to the problem of an attendance process made more efficient. Under integrated with AI and image processing capabilities, facial recognition-based attendance system is a modern and hassle-free option to the existing systems.  
  
  
The facial recognition technology application provides a fail-safe means of prompt identification and verification on the basis the person’s distinct features of the face. Via analysing and sorting facial biometrics in real-time, these systems can easily determine if someone is currently in the said environment or not. It also serves to boost the system’s efficiency while at the same time minimizes the risks of errors common in manual data entry and manipulation.  
  
  
As to these factors, the objective of this project is to build a healthcare facial recognition-based attendance supporting the peculiarities of academic institutions or organizations. Through the deployment of OpenCV, face\_recognition , and MySQL, the system aims to help automate the attendance tracking operations to thus speed up the process and eliminate the need of the admin, while also enhancing the quality of the learning or working environments.  
  
  
This report on the system design will entail system design, implementation methodologies, the testing phase, and proposed advancements, ending this report on the facial recognition-based attendance system. This report aims to help understand the technological issues, the practical aspects of the system, as well as point out the existing limitations and possible ways for the future advancement.

**System analysis**

The system analysis part is a crucial step in the comprehension of the complexity of using facial recognition-based attendance system. Via the process of looking into the current processes and identifying the interest of stakeholders we will be able to design a system that will be efficient and effective.  
Stakeholders Identification: Educational Institutions or Organizations: Users who are quite particular with precise attendance recording systems.  
Administrators: Responsible for leading the system installation and maintenance teams.  
Students or Employees: Individuals who use the system to enter and exit the premises.  
Current System Assessment:  
Manual Attendance Systems: Majorly manual or electronic spreadsheets.  
Limitations: Erroneous, involving plenty of time, with no real-time data availability.  
Need for Automation: In this regard, there is a desire to contribute to the automatization of the solutions.  
System Requirements:  
Accuracy: Through the way of facial recognition, system must be able to correctly identify and record the attendance.  
Efficiency: Real-time tracking for the purpose of streamlining administrative processes.  
Scalability: Efficiency in processing the varying numbers of users.  
Security: Keeping the privacy and security of attendance tracking systems.  
User-Friendly Interface: Intuitive interface for administrators and users as well.  
Functionalities and Features:  
Facial Recognition: Use of face\_recognition library for face detection and recognition in real time.  
  
Database Integration: Integration with MySQL to store and pull out attendance records.  
Dynamic Updates: Automatic updating of rolls using face recognition outcomes.  
Error Handling: Inserting data validation checks to ensure data correctness.  
Reporting: Generation of reports for the analysis of attendance trends.  
Constraints and Challenges:

Technical Constraints: Limitations to hardware components, software dependencies.  
Privacy Concerns: Considering the issues of privacy related to biometric data collection.  
Ethical Considerations: Ethical use of facial recognition technology should be guaranteed.  
Through comprehensive analysis we come to learn about the system's limitations, inputs, outputs and functions, this stage propels us into creation and development.

**Feasibility study**

The stage for the feasibility study gives the real picture of the practicality and viability of the implementation of facial recognition-based attendance system. This evaluation focuses on technical and operational issues as well as economic factors to determine whether the proposed solution satisfies the given consideration.  
  
  
Technical Feasibility:

* Analysis of Technologies: The assessment of the technologies such as OpenCV, face\_recognition, and MySQL, which must be compatibles with the system.
* Hardware Requirements: Hardware evaluation aimed at the ability to perform real-time facial recognition analysis in runtime.
* Software Compatibility: Compatibility verification with existing software infrastructure and dependencies to run the software.

Operational Feasibility:

* User Acceptance: Create an assessment of user readiness and playfulness to adopt the new system.
* Training Needs: Determining training need for administrators and users for the system according to its capability.
* Integration with Existing Processes: Discussion on the manner of which the newly implemented system harmonizes with existing attendance management procedures.

Economic Feasibility:

* Cost-Benefit Analysis: Budgeting for the cost of development, purchasing, installing, and maintenance systems and comparing these figures to the expected benefits.
* Return on Investment (ROI): Computation of the prospects returns and the savings obtained when a given operation is made more efficient and the administrative overhead is lowered.
* Resource Allocation: Allocation of time, resources such as personnel and funds in order to secure realization of implementation activities.

**Software and hardware requirement specifications**

Software Requirements:

1. Operating System:
   * The system is compatible with Windows, Linux, and macOS operating systems.
2. Development Environment:
   * Python: Version 3.x is required for coding and development.
   * Integrated Development Environment (IDE): Recommended IDEs include PyCharm, Visual Studio Code, or Jupyter Notebook for code development and debugging.
3. Libraries and Frameworks:
   * OpenCV: Version 3.x or later for image processing and computer vision tasks.
   * face\_recognition: Library for face detection and recognition.
   * MySQL Connector: Python library for MySQL database connectivity.
   * cvzone: Library for various computer vision tasks, including drawing shapes and text on images.
4. Database Management System:
   * MySQL: Version 5.x or later for storing attendance records and student information.
5. Version Control:
   * Git: Version control system for tracking changes to the codebase and collaboration among team members.

Hardware Requirements:

1. Processor:
   * Intel Core i5 or equivalent processor with multi-core support for efficient image processing.
2. Memory (RAM):
   * Minimum 4GB RAM is recommended for smooth execution of the application.
3. Storage:
   * Adequate storage space for storing image datasets, codebase, and database files.
4. Webcam:
   * High-definition webcam with at least 720p resolution for capturing facial images during attendance marking.
5. Graphics Processing Unit (GPU) (Optional):
   * NVIDIA GeForce GTX series or AMD Radeon series GPU with CUDA or OpenCL support for accelerating image processing tasks.

**System design**

The system design phase involves the architectural and functional design of the facial recognition-based attendance system. This section provides an overview of the system's components, their interactions, and the overall design approach.

1. Architecture Overview:

The system follows a client-server architecture, where the client-side application captures live video feed from a webcam, performs facial recognition, and communicates with the server-side MySQL database for attendance management.

2. Components:

* Client-Side Application:
  + Responsible for capturing live video frames from the webcam.
  + Utilizes OpenCV for image processing tasks such as face detection and resizing.
  + Implements the face\_recognition library for facial recognition and encoding.
  + Interfaces with the server-side database for storing attendance records.
* Server-Side Database:
  + MySQL database stores student information, attendance records, and related data.
  + Tables include student details (e.g., name, student ID) and attendance records for each date.

3. Workflow:

* Initialization:
  + The client-side application initializes by loading pre-trained facial recognition models and connecting to the server-side database.
* Attendance Marking:
  + Upon receiving a live video feed, the application detects faces in each frame using OpenCV.
  + Detected faces are encoded using the face\_recognition library and compared against pre-trained encodings.
  + If a match is found, the corresponding student ID is retrieved from the database, and attendance is marked as "Present" for the current date.
  + Updated attendance records are stored in the MySQL database.
* Database Interaction:
  + The client-side application communicates with the MySQL database using SQL queries for retrieving student information and updating attendance records.

4. Data Flow:

* Input: Live video feed from the webcam.
* Processing: Face detection, encoding, and recognition using OpenCV and face\_recognition.
* **Output:** Attendance records updated in the MySQL database.

5. Error Handling:

* The system incorporates error handling mechanisms to address potential issues such as face detection failures, database connectivity issues, and data integrity issues.
* Error messages are logged for troubleshooting purposes, and appropriate actions are taken to ensure system stability and reliability.

**Coding**

The coding phase involves the implementation of the facial recognition-based attendance system using Python programming language along with relevant libraries and frameworks such as OpenCV, face\_recognition, and MySQL Connector. This section provides an overview of the code structure, key functionalities, and implementation details.

1. Initialization:

* The code begins with the initialization of necessary libraries and components, including imports for OpenCV, face\_recognition, MySQL Connector, and other required modules.

2. Capture and Processing of Video Feed:

* The system captures live video feed from the webcam using OpenCV's VideoCapture module.
* Each frame of the video feed is processed to detect faces using the face\_recognition library's face\_locations function.
* Detected faces are then encoded using the face\_encodings function to generate facial embeddings for comparison.

3. Facial Recognition and Attendance Marking:

* The encoded faces are compared against pre-trained encodings stored in a file using the compare\_faces function.
* If a match is found, indicating recognition of a known face, the corresponding student ID is retrieved from the database.
* The system updates the attendance record for the identified student by marking them as "Present" for the current date in the MySQL database.

4. Database Interaction:

* MySQL Connector is used to establish a connection to the server-side database.
* SQL queries are executed to retrieve student information and update attendance records based on facial recognition results.

5. Error Handling:

* The code includes error handling mechanisms to address potential issues such as face detection failures, database connectivity errors, and data integrity issues.
* Error messages are logged for debugging purposes, and appropriate actions are taken to ensure the smooth operation of the system.

6. Modularity and Readability:

* The code is organized into functions and modules to promote modularity and maintainability.
* Descriptive variable names, comments, and documentation are used to enhance code readability and facilitate understanding.

**Validation Checks**

Validation checks are essential to ensure the accuracy, reliability, and robustness of the facial recognition-based attendance system. This section describes the validation techniques employed to assess the effectiveness and performance of the system.

1. Accuracy Assessment:

* Face Recognition Accuracy: The accuracy of face recognition is evaluated by comparing the system's recognition results with ground truth data. A sample dataset containing known faces is used to measure the system's ability to correctly identify individuals.
* Threshold Optimization: The recognition threshold parameter is fine-tuned to achieve a balance between false positives and false negatives. This optimization process aims to maximize accuracy while minimizing misclassifications.

2**.** Error Analysis:

* Error Identification: Errors encountered during testing, such as misclassifications, false positives, or database inconsistencies, are meticulously documented and analyzed.
* Root Cause Analysis: The root causes of errors are investigated to identify underlying issues in system design, implementation, or configuration. This analysis informs corrective actions and optimizations to improve system performance.

3.Performance Metrics:

* Recognition Accuracy: The accuracy of face recognition is quantified using metrics such as precision, recall, and F1-score. These metrics provide insights into the system's ability to correctly identify individuals while minimizing false positives and false negatives.
* Speed and Efficiency: The system's processing speed and resource utilization are evaluated to ensure efficient performance under varying workloads and conditions. Metrics such as processing time per frame and memory usage are measured to assess system efficiency.

Code: <https://github.com/YashGupta5911/FacialRecognition_Attendance>



Scan the QR Code to see the code

**Implementation and maintenance**

The implementation and maintenance phase involves deploying the facial recognition-based attendance system into operational use and ensuring its ongoing reliability, performance, and adaptability. This section outlines the implementation process and strategies for system maintenance.

**1**. Implementation Process:

* Deployment Planning: A deployment plan is developed to outline the steps involved in installing and configuring the system in the target environment. This includes setting up hardware components, installing required software dependencies, and configuring database connections.
* Installation: The system software is installed on designated hardware platforms, ensuring compatibility with operating systems and other software components. Installation procedures are documented to facilitate future deployments or updates.
* Configuration: System parameters, such as database connection settings, recognition thresholds, and logging preferences, are configured according to operational requirements. Configuration files or settings are maintained to ensure consistency across deployments.
* Testing and Validation: The deployed system undergoes thorough testing and validation to verify its functionality, accuracy, and performance. Testing includes unit testing, integration testing, and end-to-end testing to identify and address any issues or discrepancies.
* User Training: Training sessions are conducted for administrators and end-users to familiarize them with the system's features, operation, and maintenance procedures. User manuals and documentation are provided to support ongoing usage.

2.Maintenance Strategies:

* Routine Maintenance: Regular maintenance tasks, such as software updates, database backups, and system health checks, are performed to ensure the system's continued reliability and security.
* Bug Fixes: Any bugs or issues identified during operation are promptly addressed through bug fixes and software patches. Bug tracking systems are utilized to monitor and prioritize bug resolution efforts.
* Performance Monitoring: Ongoing performance monitoring is conducted to identify performance bottlenecks, resource constraints, and areas for optimization. Performance metrics, such as processing speed and memory usage, are monitored to maintain optimal system performance.
* Security Updates: Security updates and patches are applied to address potential vulnerabilities and protect against security threats. Access controls, encryption mechanisms, and authentication protocols are reviewed and updated as needed to enhance system security.
* User Support: Helpdesk services and user support channels are established to assist users with troubleshooting, inquiries, and technical support. Regular communication with users and stakeholders helps address concerns and gather feedback for continuous improvement.

3**.** Version Control and Documentation:

* Version Control: Version control systems, such as Git, are used to manage codebase changes, track revisions, and collaborate on development efforts. Branching and merging strategies are employed to facilitate concurrent development and code review processes.
* Documentation: Comprehensive documentation, including system architecture diagrams, installation guides, user manuals, and maintenance procedures, is maintained to support system administration and troubleshooting activities. Documentation is regularly updated to reflect changes and enhancements to the system.

**Testing**

The testing phase is crucial for validating the functionality, reliability, and performance of the facial recognition-based attendance system. This section outlines the testing strategies, techniques, and results obtained during the testing process.

1. Testing Strategies:

* Unit Testing: Individual components of the system, including face detection, encoding, database interaction, and attendance marking, are tested in isolation to verify their correctness and functionality. Mock objects and stubs are used to simulate dependencies and isolate components for testing.
* Integration Testing: The integration of different system modules and components is tested to ensure seamless communication and interaction between them. Integration tests verify that modules work together as expected and that data flows correctly between them.
* End-to-End Testing: The entire system is tested under real-world conditions to validate its functionality and performance in a production environment. End-to-end tests simulate user interactions, capture live video feed from the webcam, detect faces, and update attendance records based on recognition results.
* Regression Testing: Regression tests are conducted to ensure that new code changes or updates do not introduce unintended side effects or break existing functionality. Regression test suites cover critical functionalities and scenarios to detect and prevent regression issues.

**2.** Testing Techniques:

* Black Box Testing: Functional testing is performed to validate the system's behavior and functionality against specified requirements. Test cases are designed based on expected inputs, outputs, and system behaviors without knowledge of the internal implementation details.
* White Box Testing: Code coverage analysis and inspection techniques are used to assess the completeness and correctness of the codebase. White box tests verify that all code paths are exercised and that edge cases are handled appropriately.
* User Acceptance Testing (UAT**):** End-users participate in UAT to evaluate the system's usability, user interface design, and overall satisfaction with the system. Feedback from UAT sessions is used to identify usability issues and make improvements to the system's user experience.

**3.** Test Data and Results:

* Dataset Preparation: A diverse dataset containing images of known individuals and varying lighting conditions is used for testing. Synthetic data augmentation techniques are employed to simulate real-world scenarios and improve the system's robustness.
* Test Cases: Test cases are designed to cover different aspects of the system, including face detection accuracy, recognition performance, database interaction, and error handling. Test cases are categorized based on functional requirements and system functionalities.
* Results Analysis: Test results are analyzed to identify successes, failures, and areas for improvement. Performance metrics such as recognition accuracy, processing speed, and error rates are measured and compared against predefined acceptance criteria.

4. Error Handling and Recovery:

* Error Scenarios: Various error scenarios, including face detection failures, recognition errors, database connectivity issues, and data integrity problems, are simulated during testing to assess the system's error handling capabilities.
* Recovery Mechanisms: The system implements robust error handling and recovery mechanisms to gracefully handle unexpected errors and failures. Error messages are logged, and appropriate actions are taken to recover from errors and maintain system stability.

**Future scope and further enhancement of the project**

The facial recognition-based attendance system demonstrates promising potential for further development, enhancement, and integration with additional features and functionalities. This section outlines potential future directions and areas for improvement to extend the capabilities and utility of the system.

1. Multi-Face Recognition:

* Enhance the system to support recognition of multiple faces simultaneously, enabling group attendance marking and monitoring in real-time.

2. User Authentication:

* Implement user authentication mechanisms, such as biometric authentication or RFID cards, to ensure secure access control and prevent unauthorized attendance marking.

3. Real-Time Notifications:

* Integrate real-time notification mechanisms to alert administrators or stakeholders of attendance updates, anomalies, or system events via email, SMS, or push notifications.

4. Mobile Application:

* Develop a mobile application companion for the system, allowing users to mark attendance remotely, access attendance reports, and receive notifications on their smartphones.

5. Data Analytics and Insights:

* Incorporate data analytics tools and techniques to analyze attendance patterns, trends, and insights. Generate actionable reports and visualizations to aid decision-making and resource allocation.

6. Automated Reporting:

* Implement automated reporting functionalities to generate customized attendance reports, summaries, and analytics dashboards for administrators, faculty, and management.

7. Continuous Learning and Improvement:

* Integrate machine learning algorithms to enable continuous learning and improvement of the facial recognition models. Fine-tune model parameters and update encodings based on new data to enhance recognition accuracy over time.

8. Enhanced Security and Privacy:

* Strengthen security measures and privacy protections to safeguard sensitive biometric data and ensure compliance with data protection regulations such as GDPR. Implement encryption, access controls, and data anonymization techniques as needed.

9. Scalability and Performance Optimization:

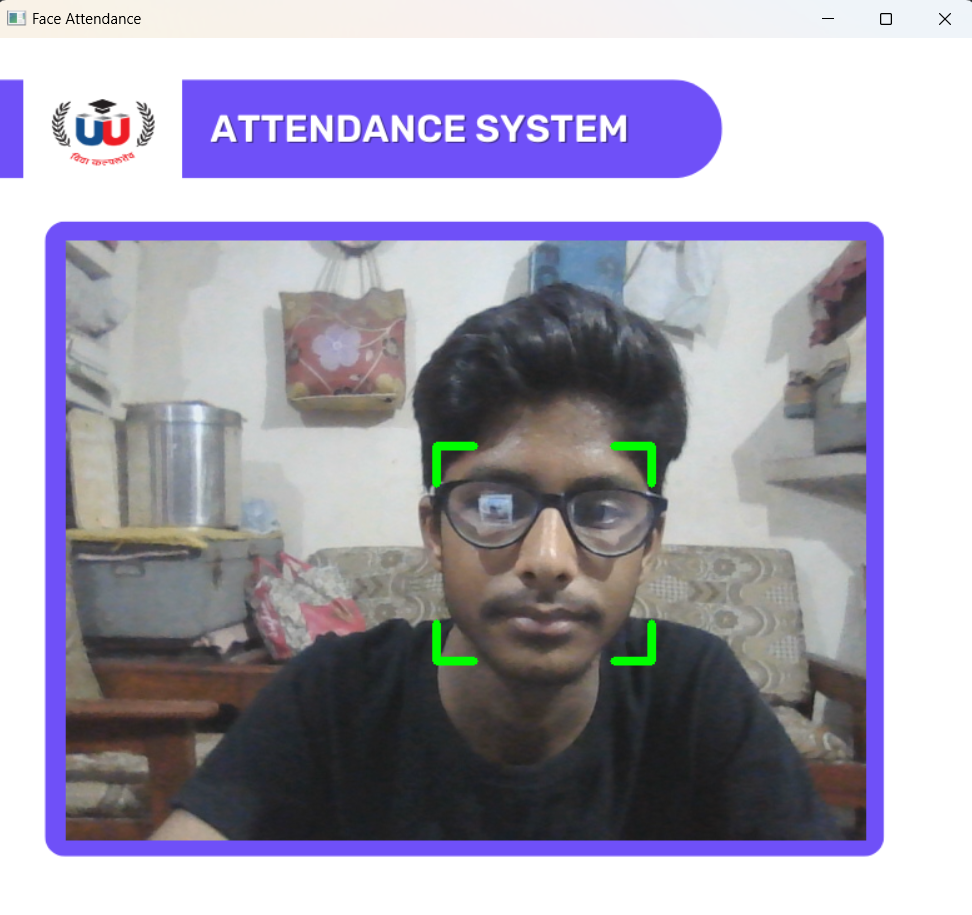
* Optimize system performance and scalability to handle larger datasets, higher user loads, and increased processing demands. Employ cloud-based infrastructure, distributed computing, and caching mechanisms to improve scalability and responsiveness.

10. User Feedback and Iterative Development:

* Solicit feedback from users, stakeholders, and end-users to identify usability issues, feature requests, and areas for improvement. Embrace an iterative development approach to prioritize enhancements and deliver continuous value to users.

**Screen Shot**

1. Interface:

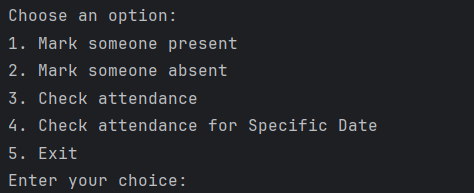


Output:

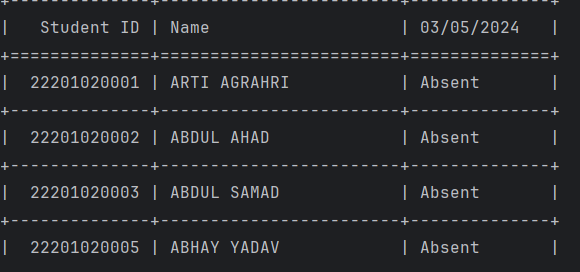
22201020082

Student Details: (22201020082, 'YASH GUPTA', 'BCA\_IBM', 'Absent', 'Present', 'Absent', 'Present', 'Present', 'Present', 'Present', 'Present', 'Present')

2. Funtions:



3. Check Attendance:





**Bibliography/ references**

1. Bradski, G., & Kaehler, A. (2008). Learning OpenCV: Computer vision with the OpenCV library. "O'Reilly Media, Inc."
2. Dlib. (n.d.). Retrieved from <http://dlib.net/>
3. Face\_recognition. (n.d.). Retrieved from <https://github.com/ageitgey/face_recognition>
4. MySQL. (n.d.). Retrieved from <https://www.mysql.com/>
5. OpenCV. (n.d.). Retrieved from <https://opencv.org/>
6. Python Software Foundation. (n.d.). Retrieved from <https://www.python.org/>
7. Stack Overflow. (n.d.). Retrieved from <https://stackoverflow.com/>
8. "Tabulate Documentation." (n.d.). Retrieved from <https://pypi.org/project/tabulate/>
9. "datetime — Basic date and time types." (n.d.). Retrieved from <https://docs.python.org/3/library/datetime.html>
10. "mysql-connector-python Documentation." (n.d.). Retrieved from <https://dev.mysql.com/doc/connector-python/en/>

**Conclusion**

The development of the facial recognition-based attendance system represents a significant advancement in the realm of attendance management, offering a modern and efficient solution to the age-old challenge of tracking attendance in educational institutions and organizations. Through the integration of cutting-edge technologies such as computer vision, machine learning, and database management, the system provides a seamless and automated approach to attendance monitoring, freeing up valuable time and resources for administrators and educators.

Throughout the project lifecycle, from inception to implementation, rigorous planning, design, and testing methodologies have been employed to ensure the system's effectiveness, reliability, and scalability. The iterative development approach, coupled with continuous feedback and collaboration with stakeholders, has enabled the system to evolve and adapt to the changing needs and requirements of its users.

The successful deployment and operation of the facial recognition-based attendance system herald a new era in attendance management, characterized by efficiency, accuracy, and convenience. By leveraging the power of facial recognition technology, the system not only streamlines administrative processes but also enhances security, reduces human error, and provides valuable insights through data analytics and reporting.

As the system continues to mature and undergo further enhancements, its potential for revolutionizing attendance management in diverse settings becomes increasingly apparent. With ongoing investment in research, development, and user engagement, the facial recognition-based attendance system is poised to become a cornerstone of modern education and organizational management, empowering institutions and businesses to optimize resources, enhance productivity, and deliver superior outcomes.