Design and Build a Teacher Attendance Application Using Geolocation and Face Recognition

Arjuna Satya Rizaldy[1], Denny Kurniadi[2], Ahmaddul Hadi[3], Ika Parma Dewi[4]

[#]*Department of Electronics Engineering, Padang State University, Jl. Prof. Dr. Hamka, Padang, Indonesia*

** Corresponding Author: arjunasatya53@gmail.com*

Abstract **-Technological advancements have facilitated the automation of various processes, including attendance recording in educational institutions. This study aims to design and develop an Android-based Teacher Attendance Application utilizing Geolocation and Face Recognition technology with the Convolutional Neural Network (CNN) method. The system was developed to replace the manual attendance method still in use at SMKN 1 Lintau Buo, which is prone to data falsification and delays in attendance recapitulation. The proposed system integrates geolocation to ensure attendance is recorded within the designated school area, while CNN-based face recognition is used for automatic teacher identity verification. The testing phase demonstrated high recognition accuracy, achieving successful face detection at distances of 20 cm, 40 cm, 60 cm, 80 cm, and up to 1 meter from the smartphone camera. In terms of facial orientation, recognition was successful when facing forward, left, right, and downward, but failed when facing upward. The overall accuracy of face detection was calculated at 83.3%. Compared to manual attendance, this system enhances efficiency by eliminating human errors, reducing fraudulent attendance records, and enabling real-time reporting. The system is built using the Flutter framework for mobile applications and Flask for the backend, connected to a MySQL database. Implementation of this system is expected to improve the accuracy, efficiency, and transparency of teacher attendance recording in educational environments.**

Keywords**— Digital Attendance, Geolocation, Face Recognition, Convolutional Neural Network, Android.**

I. INTRODUCTION

The advancement of information technology has significantly impacted various fields, including education. One critical aspect of education management is the teacher attendance recording system. In many institutions, manual attendance remains the primary method, despite its numerous limitations. These include susceptibility to data falsification, recording errors, and delays in attendance processing [1]. In an environment where timely and accurate attendance tracking is essential, adopting a more reliable and automated attendance system is crucial.

DOI:
10.24036/int.j.emerg.technol.eng.educ..v1i1.35

Corresponding author: Arjuna Satya Rizaldy,
Universitas Negeri Padang
Email: arjunasatya53@gmail.com,

Received: 04-02-2025
Revised: 10-020205
Accepted: 15-02-2025
Published: 16-02-2025

At SMKN 1 Lintau Buo, a vocational high school in Indonesia, the current manual attendance system lacks efficiency and accuracy. The school has 96 teachers, yet frequent errors in attendance reporting have been observed. Based on observations, approximately 15% of teachers experience discrepancies in attendance records each month, often due to misreporting or delayed manual input. Additionally, attendance recapitulation can take up to two days, delaying administrative processes. Furthermore, the system is prone to manipulation, where unauthorized personnel may sign in for absent teachers. Given these challenges, a modern attendance system integrating face recognition and geolocation is necessary to ensure data integrity and streamline attendance management.

Face recognition-based attendance has become a modern solution in the attendance recording system, because it has a high level of accuracy and is able to reduce data manipulation. Face recognition is a biometric technology that allows the system to automatically recognize the user's face through digital images [2]. With this technology, the system can identify individuals based on the unique facial characteristics of each person, so that the validity of attendance data can be more guaranteed.

One of the most common methods used in face recognition technology is the Convolutional Neural Network (CNN). CNN is a deep learning algorithm that is capable of automatically extracting facial features and providing very accurate classification results [3]. Previous research has shown that the application of CNN in an attendance system can increase the accuracy of face detection by up to 99% with proper dataset processing [4].

Research conducted by [5] developed a face recognition-based attendance system using CNN, achieving an 80% accuracy rate when tested on 300 facial images. The study highlighted that factors such as lighting and distance significantly affect recognition performance. Similarly, [6] implemented an R-CNN-based attendance system for webinars, which successfully identified all registered participants with optimized parameters. These studies emphasize the potential of CNN-based face recognition in attendance tracking, though improvements are still needed to enhance robustness against environmental variations.

In addition to face recognition, modern attendance systems can also be equipped with Geolocation. This technology allows the device to track the user's location by utilizing GPS coordinates, so that the system can ensure that attendance is carried out in a predetermined area [7]. The implementation of geolocation in an attendance system is very important to prevent fraud, such as recording attendance from an inappropriate location.

This research proposes the development of an Android-based Teacher Attendance Application using Flutter for the front-end interface and Flask as the backend, with MySQL for database management. The system enables automatic attendance recording, real-time data storage, and easy access for school administrators to generate reports. By leveraging modern technologies, this study contributes to the advancement of automated attendance systems in educational institutions, ensuring efficiency, accuracy, and transparency.

II. Literature Review

A. Application

An application is a software that functions as a front end in a system that is used to process various data so that it becomes useful information for its users and also related systems [8]. An application is a software that consists of several attribute components that are adjusted to user needs to help users in processing data so that it produces the desired input and output [9]. An application in the context of digital attendance systems serves as an interface that allows users to interact with the system efficiently. Mobile applications enable real-time attendance recording, improve data accuracy, and integrate advanced technologies such as geolocation and face recognition for identity verification

B. Android

Android is an operating system for Linux-based mobile devices that includes an operating system, middleware and applications [10]. Android is an open-source mobile operating system widely used for developing applications that require seamless integration with hardware features such as cameras and GPS. Its flexibility and extensive library support make it an ideal platform for implementing face recognition-based attendance systems, ensuring efficient identity verification and geolocation tracking for accurate attendance recording.

C. Absence

Attendance is a form of recording the presence of a person or employee in an institution, recorded as a report, which contains well-organized and easy-to-find attendance status data, and can be used when needed by interested parties [11].

D. Geolocation

Geolocation is the ability to track the location of a device using GPS, cell phone towers, WIFI access points, or a combination of these. Geolocation or location-based services is a general term used to describe the technology used to determine the location of a device being used. Presence can only be done in areas with a limited scope scale at the test location. Geolocation in the application functions to create an area where only the specified location can be used to do presence using a smartphone.

E. Face Recognition

Face recognition is a combination of machine learning and biometric techniques, where this technology not only looks at the level of accuracy but also its reliability in operation [12]. Face recognition has emerged as a promising alternative for enhancing the security and accuracy of attendance systems [13]. Accuracy in attendance systems is crucial to ensure real-time and precise recording of employee presence. A good facial recognition system is a system with a facial database that has been processed through the extraction of certain features, this technology is usually used for authentication, validation, authorization, and identification.

F. Machine Learning

Machine Learning (ML) is a branch of artificial intelligence (artificial intelligence) is a discipline that includes the design and development of algorithms that allow computers to develop behavior based on empirical data. Machine Learning can also be interpreted as a discipline that tasks computers to learn and act like humans, and improve their learning abilities over time automatically, by supplying data and information as a form of real-world experience and interaction. While Mitchell explains that Machine Learning is the ability of computers to learn from experience E to tasks T that are charged with measurable performance P [14].

G. Convolutional Neural Network

Basically, the CNN algorithm is a more effective artificial neural network architecture for image classification. The main concept of CNN itself is in the convolution operation it has, where an image will be extracted for each feature to form several patterns that will be easier to classify.

A previous study titled "Design of a Facial Recognition Attendance System Using CNN and Liveness Detector at BPR Central Dana Mandiri," conducted in April 2022 by [15], implemented the Convolutional Neural Network (CNN) algorithm for face detection along with Eye Aspect Ratio (EAR) calculations as a Liveness Detector. The results of this study demonstrated that the system successfully recognized all faces stored in the system’s database, achieving an accuracy rate of 100%.

Another prior study also addressed this issue using a face recognition algorithm based on CNN. The research, titled "Student Attendance System Using Face Recognition with CNN Algorithm," conducted in March 2024 by [16], utilized 60 images divided into four classes. The evaluation, performed using a confusion matrix on a modified Convolutional Neural Network (CNN) algorithm, showed that the model training process was significantly accelerated,

completing in 1.23 hours. The final model achieved an accuracy of 99.85%.

H. Front end

1) Flutter

The flutter framework is an application development template that has high performance and an attractive appearance. Applications built using the flutter framework can be easily developed across platforms such as Android, iOS, and the web. Flutter is also dynamic, meaning that the coding components can be customized so that the appearance is user friendly and not rigid. In the development structure, the author also uses state management rules that can improve the performance of application use.

2) Dart

Dart is a programming language developed by Google and first introduced in 2011. Dart is designed as an efficient, fast, and easy-to-use language for building modern applications, especially for mobile, web, desktop, and backend application development. Dart became popular because it is used in the Flutter framework, which is widely used to create cross-platform mobile applications (Android and iOS) with a single code base.

3) Library or Package

a) http

http is a library used for sending HTTP requests in Flutter. Its primary function is to send requests to the Flask back-end API, such as login requests, retrieving attendance data, or requesting reports in PDF format.

b) Shared_preferences

Shared_preferences is a library for storing local data in key-value format on a device. Its primary function is to store important information such as username, role, and attendance session data, ensuring that the application can retain user data even after it is closed.

c) Camera

Camera is a library that provides direct access to a device's camera and enables real-time preview. It serves as a key component in the face recognition process, allowing users to interact seamlessly with the attendance system and facilitating an automated, real-time experience.

d) Geolocator

Geolocator is a library that retrieves the GPS location of a device. Its primary function is to obtain user location coordinates and validate whether the location falls within the designated office area, for instance, using a predefined radius.

e) Path_provider

Path_provider is a library that grants access to common device directories. Its primary function is to store downloaded attendance reports (PDF files) from the back-end, making them accessible for users to open or save.

f) Open_file

Open_file is a library used to open files with the device's default applications. Its primary function is to automatically open downloaded attendance reports (PDF format), allowing users to view or print them.

I. Back-end

1) Flask

Flask is a type of microframework that allows developers to build well-structured and organized web applications[17]. With Flask, developers can manage the behavior of web applications more easily, without having to be tied to certain rigidities or complexities.

2) Python

Python is a multipurpose interpretive programming language with a design philosophy that focuses on code readability [18]. Python is claimed to be a language that combines capabilities, capabilities, with very clear code syntax, and is equipped with a large and comprehensive standard library functionality. Python is also supported by a large community.

3) Library or Package

a) Flask-cors

Flask-cors is a Flask extension library that enables Cross-Origin Resource Sharing (CORS), a mechanism that allows web applications to interact with resources from different domains. Its primary function is to permit front-end applications (Flutter), running on a different domain/port, to access the back-end API without being blocked by the same-origin policy.

b) mysql-connector-python

Mysql-connector-python is a library used to connect and communicate with a MySQL database. Its primary function is to facilitate CRUD (Create, Read, Update, Delete) operations on the database.

c) Pdfkit

Pdfkit is a Python library that serves as a wrapper for wkhtmltopdf, which converts HTML into PDF format. Its primary function is to generate attendance reports in PDF format by converting the HTML-based reports created on the back-end.

d) Wkhtmltopdf

Wkhtmltopdf is an external program library (binary) that converts HTML into PDF format. It is required by Pdfkit to transform attendance data generated in HTML format into a PDF file.

e) MTCNN

MTCNN (Multi-task Cascaded Convolutional Networks) is a library for face detection. Its primary function is to detect faces in images more accurately as part of the face recognition module.

f) OpenCV

OpenCV is a Python library for image processing. Its primary functions include cropping faces, resizing images, and converting color formats (e.g., BGR to RGB) before feeding the data into a CNN model.

g) Numpy

Numpy is a Python library for numerical computing and array manipulation. Its primary function is to process image data (pixel arrays) and

perform necessary numerical operations for preprocessing and model inference.

h) Tensorflow

TensorFlow is a machine learning and deep learning library. Its primary function is to load and execute a pre-trained CNN model for face recognition.

J. *MySQL*

MySQL can be used to create and manage databases and their contents. SQL itself is a language used to retrieve data from relational databases or structured databases. Therefore, MySQL is a database management system that uses the SQL language as a communication language between application software and database servers.

K. *System Development Methods*

1) Waterfall

The Waterfall method is one of the most widely used software development methods [19]. This model takes a systematic and sequential approach. It is called waterfall because each stage that is passed must wait for the completion of the previous stage and run sequentially. The Waterfall method is one of the most straight-forward system development methodologies. The structure of this model is 6 stages consisting of: Requirements Gathering and Analysis, Systems Development, Systems Implementation and Coding, Testing, Deployment and Systems Operation and Maintenance.

2) UML

UML (Unified Modeling Language) is a language benchmark for writing detailed frameworks of software [20]. UML is used to visualize, specify, build, and document a software system. UML is a tool for visualizing and documenting the results of analysis and design that contains syntax for modeling systems visually.

III. METHOD

The design of the teacher attendance application using geolocation and face recognition was carried out using a number of tools and materials as well as a number of system development steps. The method used is the Waterfall method. The Waterfall method is one of the most straight-forward system development methodologies. In this chapter, the author takes steps in the waterfall method consisting of: Requirements Gathering and Analysis, Systems Development, Systems Implementation and Coding, Testing, Deployment and Systems Operation and Maintenance.

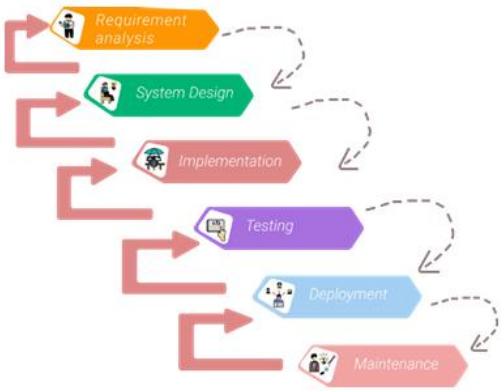

Fig1. Waterfall Method

Requirements

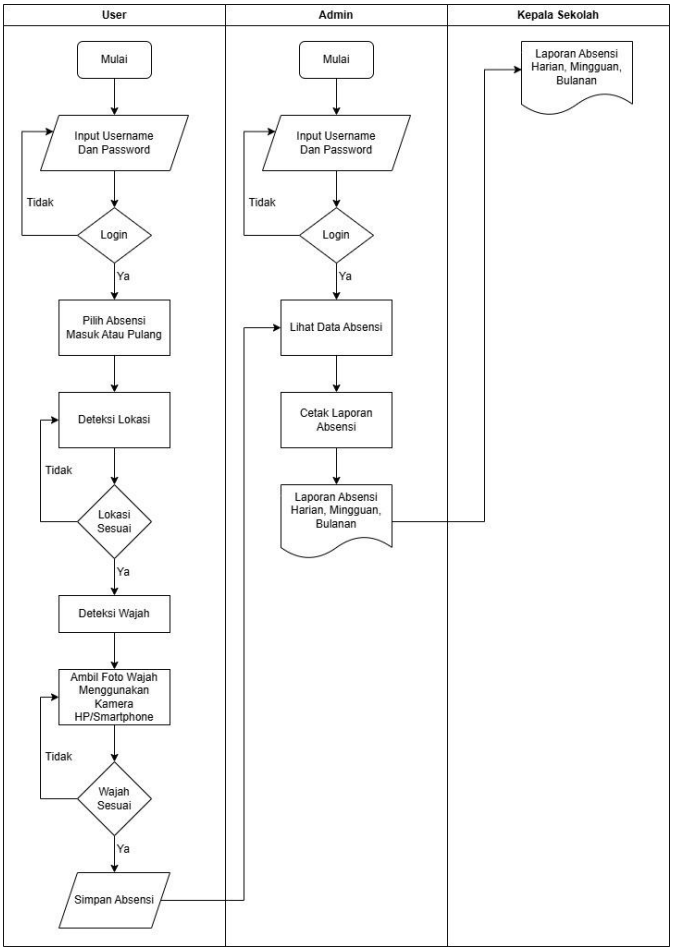

Fig2. Proposed System Analysis Flowmap

Unified Modelling Language (UML)

A. *Class Diagram*

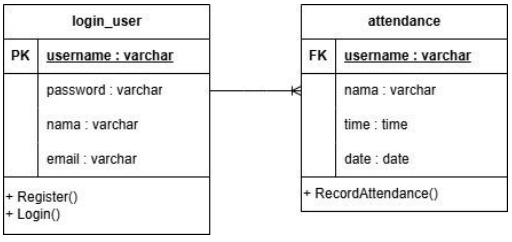

Fig3. Class Diagram

B. *Use Case Diagram*

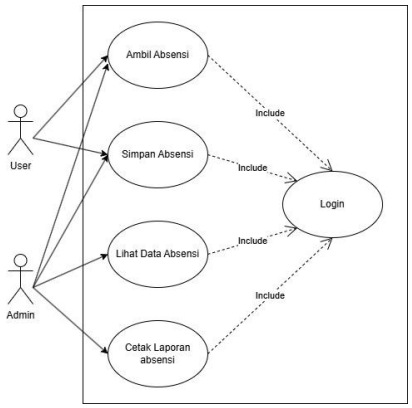

Fig4. Use Case Diagram

C. *Activity Diagram*

1) Activity Diagram Admin

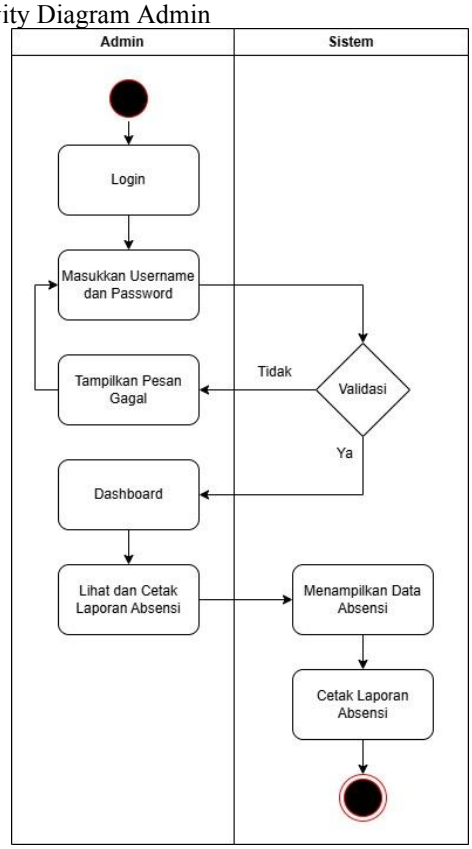

Fig5. Activity Diagram Admin

2) Activity Diagram User

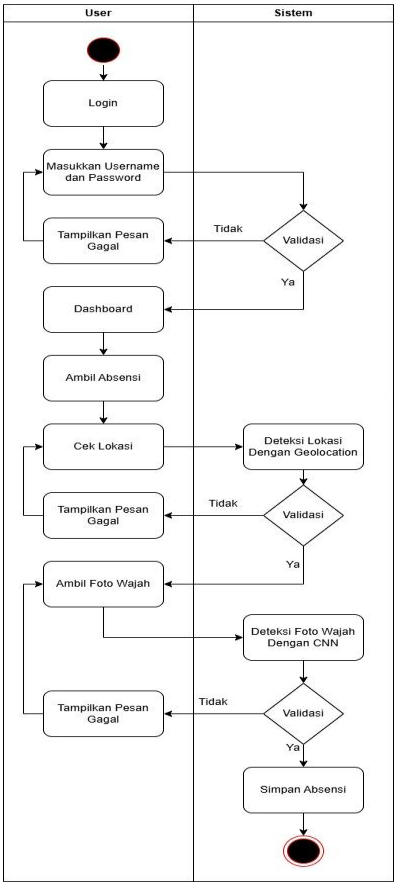

Fig6. Activity Diagram User

D. *Sequence Diagram*

1) Sequence Diagram Admin

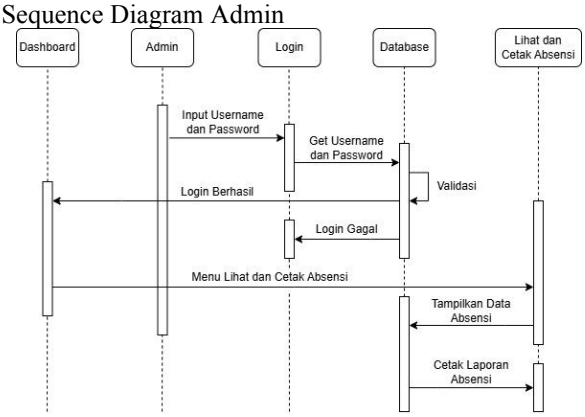

Fig7. Sequence Diagram Admin

2) Sequence Diagram User

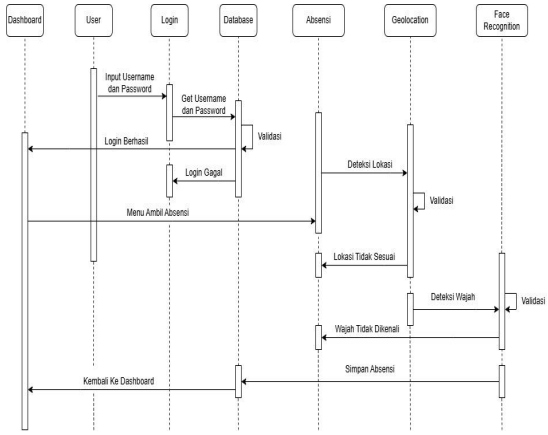

Fig8. Sequence Diagram User

E. Flow chart

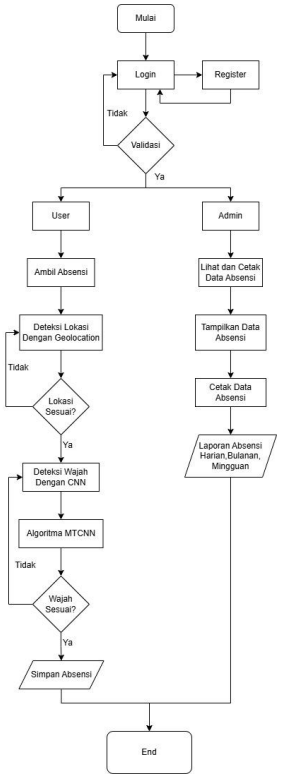

Fig9. Flow chart

F. Entity Relationship Diagram (ERD)

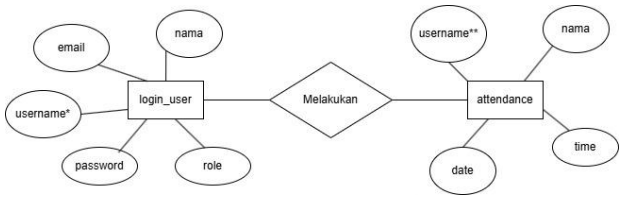

Fig10. Entity Relationship Diagram

Dataset Creation Workflow

The Teacher Attendance application employs face recognition using the CNN method, specifically implementing the MTCNN (Multi-Task Cascaded Convolutional Network) algorithm. MTCNN is an advanced model derived from CNN that consists of three network layers (P-Net, R-Net, and O-Net), utilizing candidate selection and classification to achieve efficient and fast face detection. The following outlines the dataset creation workflow for face detection in the Teacher Attendance application.

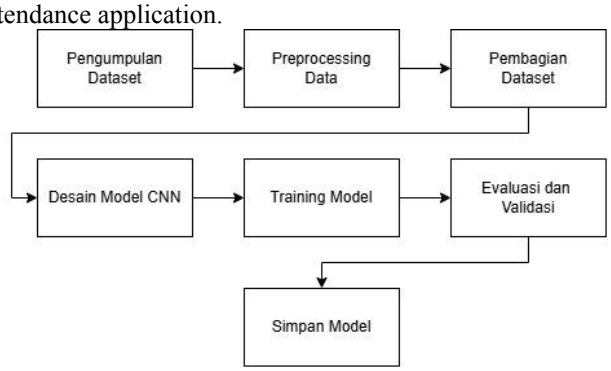

Fig 11. Dataset Creation Workflow

A. Dataset Collection
   1) Collect facial images of each individual to be recognized.
   2) Capture 64 facial images per individual for training purposes.
   3) Ensure the dataset includes variations in expression, angles, lighting, and background to enhance model robustness.

B. Data Preprocessing
   1) Face Detection: Utilize MTCNN to detect faces and obtain bounding boxes.
   2) Cropping: Crop the images to retain only the relevant facial region.
   3) Resizing: Resize the cropped images to a consistent dimension, such as 128 × 128 pixels.
   4) Normalization: Scale pixel values (e.g., dividing by 255) to ensure a range between 0 and 1 for better model performance.
   5) Data Augmentation (Optional): Apply augmentation techniques such as rotation, flipping, zooming, and brightness adjustment to improve dataset diversity.

C. Dataset Splitting
   Divide the preprocessed dataset into training and validation sets to evaluate model performance during training.

D. CNN Model Design
   1) Design the CNN architecture by incorporating convolutional layers, pooling layers, and fully connected layers.
   2) Use a softmax layer at the final stage to generate class predictions.

E. Model Training
   1) Train the model using the training dataset with proper hyperparameter tuning (e.g., learning rate, batch size, number of epochs).
   2) Monitor metrics such as loss and accuracy on the validation data to ensure the model does not overfit and achieves good convergence.

F. Model Evaluation and Validation
   1) Evaluate model performance using the validation dataset.
   2) Perform hyperparameter tuning and refinements if necessary to enhance accuracy.

G. Model Saving

Once the model achieves satisfactory performance, save it into a file (e.g., cnn_face_model.h5) for future inference and deployment.

Implementation

A. Login Page All Users

The login page is the page that appears when the application is first run. The login page is used as an intermediary to enter the application for users who have been given access rights. On this login page there is the application name and a form to enter email and password and a button to login for users who will use this application. The appearance of the login page is as follows:

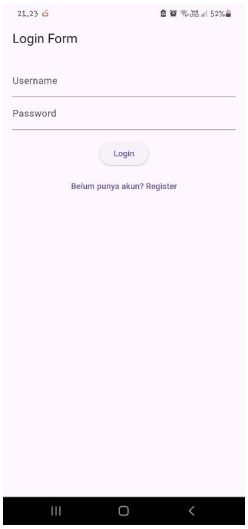

Fig12. Login Page View

B. Register Page

The register page is a page to register new users if they do not have a username and password to enter the attendance application. Here is a display of the register page:

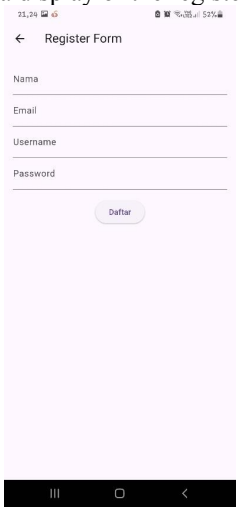

Fig13. Register Page View

C. Dashboard Page

The dashboard page is a page of the attendance application after we have successfully logged into the application. In this attendance application there are two dashboard pages, namely the dashboard page for users with the user role and users with the admin role.

1) Admin Dashboard

Fig14. Admin Dashboard

2) User Dashboard

Fig15. User Dashboard

D. Attendance Page

The attendance page is a page for all users to take attendance. The following is a display of the attendance page before detecting the location:

Fig16. Attendance Page Before Location Detection

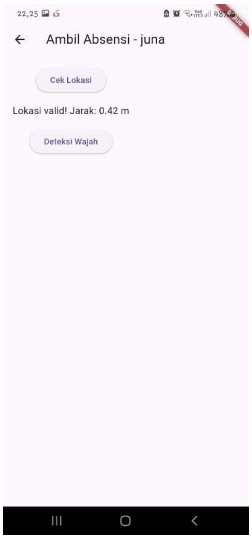

Fig17. Attendance Page After Location Detection

E. Face Recognition Page

The face detection page is a page to detect user faces using the MTCNN algorithm. The following is a display of the face detection page if the face matches:

Fig18. Face Recognition Page If the Face Matches

Fig19. Face Recognition Page If No Face Is Detected Or The Face Does Not Match

F. View and Print Attendance Page

The view and print attendance page can only be seen by the admin. This page is a page to view the attendance data of users who have taken attendance. All attendance data that has been stored in the database will be displayed and can be printed into daily, weekly, and monthly attendance reports. Here is the display of the view and print attendance page:

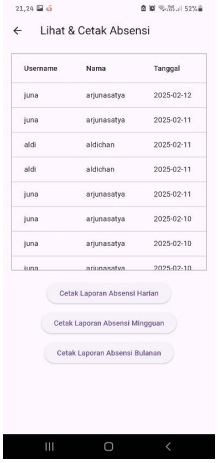

Fig20. View and Print Attendance Page

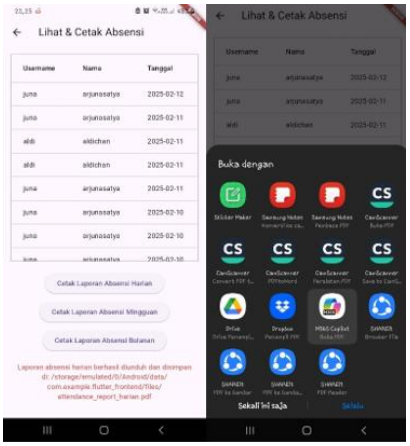

Fig21. View Saved Report And Print Attendance

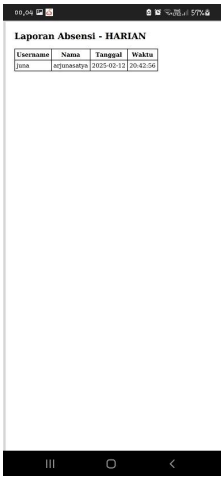

Fig22. Daily Attendance Report View

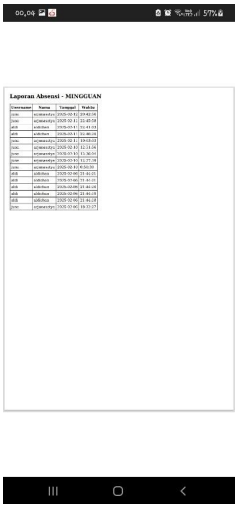

Fig23. Weekly Attendance Report View

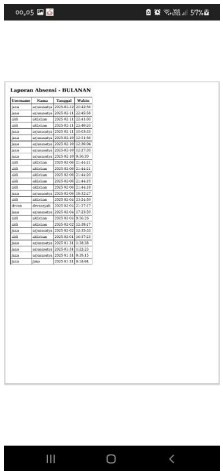

Fig24. Monthly Attendance Report View

Testing

Table1. Black Box Testing

No.	Teacher Attendance Application Functional	Functioning	Does not work
1.	Ensure that users can register and log in with valid data, and that the system handles cases of invalid data.	Functioning	
2.	Ensure that the appearance and access of features on the dashboard differ according to the user's role.	Functioning	
3.	Test whether the application can retrieve GPS data and verify that the user's location is within the specified area. Ensure that attendance cannot be performed if the location is not appropriate.	Functioning	
4.	Test whether the application can correctly detect faces from the live camera, display bounding boxes with appropriate labels, and reject attendance if the face does not match the logged-in user.	Functioning	
5.	Test whether the attendance is successfully recorded to the server database when the conditions are met and check whether the stored data is correct.	Functioning	
6.	Verify that any invalid input results in a proper error response from the backend.	Functioning	
7.	Testing the live camera can provide a smooth and responsive preview, and check whether the face detection bounding box is displayed.	Functioning	
8.	Test the attendance report feature by generating daily, weekly, and monthly reports. Ensure that the PDF files are generated correctly and contain accurate attendance data.	Functioning	
9.	Test communication between the Flutter app and the backend API. Make sure requests and responses work properly without any timeouts or communication errors.	Functioning	
10.	Testing application access from different devices and different networks to ensure the application is publicly accessible.	Functioning	

IV. CONCLUSIONS

This research has successfully designed and implemented an Android-based Teacher Attendance Application with Geolocation and Face Recognition features. This application allows users to register, login, detect location, and verify faces in the attendance process. This system is supported by a Flask API-based backend developed using the Python programming language, as well as a database stored on a VPS Server. The test results using the Black Box method show that the application has run as expected.

REFERENCES

[1] Wijaya, N., Febriyanti, A. R., & Wibowo, A. (2020). Web-Based Employee Data Management Application at PT. Pelayaran Sakti Inti Makmur Palembang. Jurnal Sisfokom (Sistem Informasi dan Komputer), 9(1), 42-50.

[2] Nuryanto, W. J. (2017). Face Recognition Using the SURF (Speeded Up Robust Features) Method. Surakarta.

[3] Anggraini, D., Arsal, M., & Wardijono, B. A. (2021). Face Recognition for Bank Employee Access Using Deep Learning with CNN Method. Jurnal Nasional Teknologi dan Sistem Informasi, 1, 55–63. https://doi.org/10.1109/UBMK52708.2021.9559031

[4] Firmansyah, A., Itsnan, A. F., Apip, A., Mullia, R. T., & Rosyani, P. (2024). Sistem Absensi Mahasiswa Menggunakan Face Recognition dengan Algoritma CNN. AI dan SPK: Jurnal Artificial Intelligent dan Sistem Penunjang Keputusan, 1(4).

[5] Zahrah, S., & Azhar, A. (2022). Sistem Deteksi Wajah Untuk Pencatatan Kehadiran Mahasiswa di Kelas Menggunakan Metode Convolutional Neural Network. Jurnal Artificial Intelligence dan Software Engineering (J-AISE), 2(1).

[6] Putra, A. T., Usman, K., & Saidah, S. (2021). Webinar Student Presence System Based on Regional Convolutional Neural Network Using Face Recognition. Jurnal Teknik Informatika (JUTIF), 2(2), 109-118.

[7] Franenfield, J. (2021). Geolocation. Investopedia. Retrieved from https://www.investopedia.com/terms/g/geolocation.asp.

[8] Alfiah, A., & Damayanti, D. (2020). E-Marketplace Application for Selling Catfish Harvest Results (Case Study: Pringsewu Regency, Pagelaran District). Jurnal Teknologi dan Sistem Informasi, 1(1), 111-117.

[9] Kinaswara, T. A., Hidayati, N. R., & Nugrahanti, F. (2019). Design and Construction of a Website-Based Inventory Application in Bantengan Village. Prosiding Seminar Nasional Teknologi dan Komunikasi (SENATIK), 2(1), 71–75.

[10] Supardi, S., & Yuniar, Y. (2014). Everyone Can Be an Android Programmer. Jakarta: PT Elex Media Komputindo.

[11] Wahyu, F., & Agung, W. (2022). Digitalization of Student Attendance at SMA Negeri 1 Petanahan. Jurnal Sinkronisasi (Sistem Informasi, Informatika dan Komputer), 2, 28–31.

[12] Isum, R., Maryati, S., & Tryadmojo, B. (2019). Accuracy of Face Recognition System Using OpenCV Face Recognition System with Haar Cascade Method on Raspberry Pi. Jurnal Scientific Informatics (JIF), Cv, 12790.

[13] Isputrawan, M. F., & Suriyanti, S. (2023). Pengembangan Aplikasi Absensi Berbasis Web Menggunakan Face Recognition. Jurnal Teknoinfo, 17(1), 55-65. https://doi.org/10.33365/jti.v17i1.2243.

[14] Fahrizal, F., Reynaldi, F. O., & Hikmah, N. (2020). Implementation of Machine Learning in PETS Identification System Using Python Based on Ubuntu. Jurnal Sistem Informasi, Informatika, dan Komputasi, 4(1), 86–91.

[15] Sari, K. (2022). Perancangan Sistem Absensi Facial Recognition Menggunakan CNN dan Liveness Detector pada BPR Central Dana Mandiri. Jurnal Informatika dan Rekayasa Komputer (JAKAKOM), 2(1), 70-80. https://doi.org/10.33998/jakakom.2022.2.1.63.

[16] Firmansyah, A., Itsnan, A. F., Apip, A., Mullia, R. T., & Rosyani, P. (2024). Sistem Absensi Mahasiswa Menggunakan Face Recognition dengan Algoritma CNN. AI dan SPK: Jurnal Artificial Intelligent dan Sistem Penunjang Keputusan, 1(4).

[17] Izzathohir, K. M., & Yulianton, H. (2024). Web-Based Palm Sugar Sales Application System Using Flask Framework. Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi), 8(1), 163-169.

[18] Setia, B., & Komang, I. (2018). Application for Computer Operation by Detecting Motion Using OpenCV Python. Prosiding SINTAK, 190–191.

[19] Ruspandi, R., Rivardi, B., Sherwin, S., & Feisy, K. (2018). Implementation of Azure Cognitive Service for Speech Recognition Applications. Jurnal Teknik Informatika, 13(4), 1–10.

[20] Sa'ad, S., & Muhammad, I. (2020). Autodidactic Web Programming: Creating an Edutainment Website. Jakarta: Gramedia.