# GURU TEGH BAHADUR INSTITUTE OF TECHNOLOGY

SYNOPSIS



ACADEMIC SESSION:2024-2025

BRANCH: ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

SUBJECT: PRACTICUM-INTEGRATED PROJECT

MENTOR: MS Shikha Bhalla TEAM MEMBERS:

1. Shreyas Jani (013)
2. Chirag Garg (022)
3. Gurnoor Singh Khurana (032)
4. Virat Chauhan (035)

# ACKNOWLEDGEMENT

We extend our heartfelt appreciation to our esteemed supervisor, Ms. Shikha Bhalla, whose unwavering support and invaluable suggestions have played a pivotal role in elevating this dissertation to its current standard. Without her guidance, achieving this level of excellence would have been an arduous task.

In addition, we seize this moment to express gratitude to all others who have contributed their support throughout the project and various facets of our academic journey at Guru Tegh Bahadur Institute of Technology.

Date:   
Mentor’s Name: Ms. Shikha Bhalla

Position: Assistant Teacher

Signature:

# INTRODUCTION

In today's educational landscape, streamlining administrative processes is crucial. Facial recognition attendance systems powered by machine learning (ML) offer a sophisticated solution for student attendance tracking. These systems leverage cutting-edge technology to automate attendance procedures, improving accuracy and efficiency.

The core technology lies in the ML model. During setup, the system is trained on a dataset of student photographs with their consent. Using complex algorithms, the ML model extracts unique facial features from each image, creating a mathematical representation for each student. When a student enters the classroom, a camera captures their face, and the system compares it to the stored representations. If there's a match, the student's attendance is automatically marked.

Facial recognition attendance systems deliver numerous advantages for educational institutions. They eliminate the time-consuming process of manual attendance rolls, allowing instructors to dedicate more time to teaching. Additionally, these systems minimize the potential for errors and buddy punching, ensuring accurate attendance records. Furthermore, the data collected can be used to identify attendance patterns and provide valuable insights for educators to improve student engagement and support.

# REQUIREMENT ANALYSIS

The System Requirements Specification (SRS) for the Student Performance Prediction system is a comprehensive document that delves into the intricacies of designing and implementing a robust predictive analytics platform for academic success. This endeavor begins with a thorough analysis of the educational landscape, taking into account the diverse stakeholders involved, including educators, administrators, students, and support staff.

One of the primary objectives of the SRS is to identify and understand the specific needs and challenges faced by these stakeholders. For educators and administrators, the focus lies on having a tool that can accurately forecast student performance, enabling them to intervene proactively and provide targeted support. Students, on the other hand, benefit from personalized learning experiences and timely interventions that cater to their individual strengths and weaknesses.

The SRS outlines both the functional and non-functional requirements of the system. Functional requirements encompass features such as data collection from various sources (e.g., attendance records, grades, study habits), data processing and analysis using machine learning algorithms, prediction of academic outcomes, generation of reports and alerts, and integration with existing educational systems.

Non-functional requirements cover aspects like scalability, reliability, security, usability, and performance metrics.

# SYSTEM DESIGN

The system design of the Student Performance Prediction model is a critical phase that involves architectural decisions and data modeling techniques to ensure the system's effectiveness and efficiency. One of the key architectural choices is the adoption of a client-server architecture, which facilitates seamless communication and data exchange between the user interface (client) and the backend processing and storage (server).

The client-server architecture enables users, including educators and administrators, to interact with the system through a user-friendly interface. This interface allows them to input data, access predictive analytics, view reports and recommendations, and initiate interventions as needed. On the server side, data processing, machine learning algorithms, and predictive modeling take place to generate insights and predictions regarding student performance.

An Entity-Relationship (ER) diagram is utilized to design the data model of the system, depicting the various entities (e.g., students, courses, grades, attendance records) and their relationships. This diagram serves as a blueprint for database design, ensuring data integrity, normalization, and efficient storage and retrieval mechanisms.

Furthermore, a Data Flow Diagram (DFD) is created to illustrate the flow of data within the system. The DFD delineates how data moves from input sources (e.g., student data sources, attendance tracking systems) through various processing stages (e.g., data cleansing, feature extraction, predictive modeling) to the output interfaces (e.g., reports, alerts, personalized recommendations). This diagram aids in understanding data transformations, processing logic, system components, and integration points.

# ALGORITHMS USED

Logistic Regression: This algorithm is often used for binary classification tasks, such as predicting whether a student will pass or fail based on various features like attendance, grades, study habits, etc.

Decision Trees: Decision tree algorithms, including Random Forest and Gradient Boosting Machines (GBM), are effective for handling both classification and regression tasks. They can capture complex relationships between input features and output labels, making them suitable for predicting student performance.

Support Vector Machines (SVM): SVM is a powerful algorithm for classification tasks. It works well for separating classes in high-dimensional spaces, making it useful when dealing with multiple features influencing student performance.

Neural Networks: Deep learning models like Artificial Neural Networks (ANNs) and Convolutional Neural Networks (CNNs) can be employed for complex prediction tasks. They can learn intricate patterns and relationships in data, providing high accuracy in student performance prediction.

K-Nearest Neighbors (KNN): KNN is a non-parametric algorithm used for both classification and regression tasks. It predicts the output label of a data point by considering the 'k' nearest neighbors based on feature similarity.

Clustering Algorithms: Unsupervised learning algorithms like K-Means clustering or Hierarchical clustering can be used to group students based on similar characteristics, which can then be used for targeted interventions and personalized support.

Reinforcement Learning: Although less commonly used in student performance prediction, reinforcement learning techniques can be applied in educational settings to optimize learning strategies and adapt interventions based on student feedback.

# REFERENCES

* 1. Jones, A., & Smith, B. (2021). Predictive Analytics in Education: Leveraging Machine Learning for Student Success. Journal of Educational Technology, 12(3), 45-60.
  2. Johnson, C., & Williams, D. (2022). Enhancing Learning Experiences through Personalized Interventions: A Case Study in Higher Education. International Conference on Educational Technologies Proceedings, 78-89.
  3. Anderson, E., et al. (2023). Designing Effective Student Performance Prediction Systems: Best Practices and Lessons Learned. ACM Transactions on Educational Technology, 5(2), 110-125.
  4. Patel, R., et al. (2024). Data-Driven Approaches to Student Support: A Review of Current Trends and Future Directions. IEEE Transactions on Learning Technologies, 7(1), 30-45.