**Employee Performance Analysis**

A Minor Project Report

Submitted in partial fulfillment of requirement of the

Degree of

**BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE & ENGINEERING**

**By**

Hamood Ahmed – EN22CS301378  
Harsh Bangar – EN22CS301385  
Gaurav Nagraj – EN22CS301368

Under the Guidance of

Prof. Rashmi Chouhan



**Department of Computer Science & Engineering**

**Faculty of Engineering**

**MEDICAPS UNIVERSITY, INDORE- 453331**

**APRIL-2025**

Title Page (Annexure-B, Remove after finalization)

Approval Sheet (Annexure-D, Remove after finalization)

**Report Approval**

The project work **“**Employee Performance Analaysis**”** is hereby approved as a creditable study of an engineering/computer application subject carried out and presented in a manner satisfactory to warrant its acceptance as prerequisite for the Degree for which it has been submitted.

It is to be understood that by this approval the undersigned do not endorse or approve any statement made, opinion expressed, or conclusion drawn there in; but approve the “Project Report” only for the purpose for which it has been submitted.

Internal Examiner

Name:

Designation

Affiliation

External Examiner

Name:

Designation

Affiliation

Declaration (Annexure-E, Remove after finalization)

**Declaration**

We hereby declare that the project entitled **“**Employee Performance Analysis**”** submittedin partial fulfillment for the award of the degree of Bachelor of Technology in ‘Department of Computer Science & Engineering’ completed under the supervision of Prof. Rashmi Chouhan, Faculty of Engineering, Medicaps University Indore is an authentic work.

Further, we declare that the content of this Project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University for the award of any degree or diploma.

**Signature and name of the student(s) with date**

Certificate (Annexure-F, Remove after finalization)

**Certificate**

We, Prof. Rashmi Chouhan certify that the project entitled **“**Employee Performance Analysis**”** submittedin partial fulfillment for the award of the degree of Bachelor of Technology by Hamood Ahmed, Harsh Bangar, and Gaurav Nagraj istherecordcarried out by them under our guidance and that the work has not formed the basis of award of any other degree elsewhere.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Prof. Rashmi Chouhan**  
Department of Computer Science & Engineering

Medicaps University, Indore

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Dr. Ratnesh Litoriya

Head of the Department

Computer Science & Engineering

Medicaps University, Indore

Acknowledgements (Annexure-G, Remove after finalization)

**Acknowledgements**

I would like to express my deepest gratitude to the Honorable Chancellor, **Shri R C Mittal,** who has provided me with every facility to successfully carry out this project, and my profound indebtedness to **Prof. (Dr.) D. K. Patnaik,** Vice Chancellor, Medicaps University, whose unfailing support and enthusiasm has always boosted up my morale. I also thank **Prof. (Dr.) Pramod S. Nair,** Dean, Faculty of Engineering, Medicaps University, for giving me a chance to work on this project. I would also like to thank my Head of the Department **Dr. Ratnesh Litoriya** for his continuous encouragement for the betterment of the project.

It is their help and support, due to which we became able to complete the design and technical report.

Without their support this report would not have been possible.

Our sincere thanks to our project guide Prof. Rashmi Chouhan for her constant guidance, motivation, and support.

**Hamood Ahmed**  
**Harsh Bangar**  
**Gaurav Nagraj**

B.Tech. III Year (A)

Department of Computer Science & Engineering

Faculty of Engineering

Medicaps University, Indore

**Abstract**

Employee performance analysis is essential for organizations to evaluate productivity, efficiency, and contributions toward business goals. This project leverages data-driven insights to identify high-performing employees, assess common challenges, and determine the impact of leadership, work culture, and incentives on performance using Python and machine learning.

Employee performance analysis is a crucial tool for evaluating how effectively employees contribute to the goals of an organization. Through this project, we aim to leverage machine learning models and data analytics to automate and enhance the process of evaluating employee performance. The system is capable of analyzing performance trends, predicting outcomes, and generating insights for decision-making by HR professionals. Visualizations such as performance graphs and accuracy metrics are used to present clear insights. This data-driven approach ensures objectivity and helps in identifying both high performers and those needing improvement, ultimately aiding better workforce management.

**Keywords:**

Employee performance, machine learning, HR analytics, Python, productivity assessment.

Employee performance, data visualization, workforce management, artificial intelligence, evaluation system, data science, predictive analytics, HR decision support.

Contents (Annexure-H, Remove after finalization)

**Table of Contents**

|  |  |  |
| --- | --- | --- |
|  |  | **Page No.** |
|  |  |  |
|  | Report Approval | ii |
|  | Declaration | iii |
|  | Certificate | iv |
|  | Acknowledgement | v |
|  | Abstract | vi |
|  | Table of Contents | vii |
|  | List of figures | viii |
|  | List of tables | ix |
|  | Abbreviations | x |
|  | Notations & Symbols | xi |
| Chapter 1 | Introduction (Whichever is applicable) |  |
|  | 1.1 Introduction | 1 |
|  | 1.2 Literature Review |  |
|  | 1.3 Objectives |  |
|  | 1.4 Significance |  |
|  | 1.5 Research Design |  |
|  | 1.6 Source of Data |  |
|  | 1.7 Chapter Scheme |  |
| Chapter 2 | REQUIREMENTS SPECIFICATION |  |
|  | 2.1 User Characteris |  |
|  | 2.2 Functional Requirements |  |
|  | 2.3 Dependencies |  |
|  | 2.4 Performance Requirements |  |
|  | 2.5 Hardware Requirements |  |
|  | 2.6 Constraints & Assumptions |  |
| Chapter 3 | DESIGN ( Whichever is applicable) |  |
|  | 3.1 Algorithm (if Applicable) |  |
|  | 3.2 Function Oriented Design for procedural approach |  |
|  | 3.3 **System Design (**Whichever is applicable) |  |
|  | 3.3.1 Data Flow Diagrams (Level 0,Level1) |  |
|  | 3.3.2 Activity Diagram |  |
|  | 3.3.3 Flow Chart |  |
|  | 3.3.4 Class Diagram |  |
|  | 3.3.5 ER Diagram |  |
|  | 3.3.6 Sequence diagram |  |
|  | 3.4 **Database Design** |  |
|  | 3.4.1 Logical Database Design |  |
|  | 3.4.2 Physical Database Design |  |
| Chapter 4 | Implementation, Testing, and Maintenance |  |
|  | 4.1 Introduction to Languages, IDE’s, Tools and Technologies used for Implementation |  |
|  | 4.2 Testing Techniques and Test Plans (According to project) |  |
|  | 4.3 Installation Instructions |  |
|  | 4.4 End User Instructions |  |
|  |  |  |
| Chapter 5 | Results and Discussions |  |
|  | 5.1 User Interface Representation (of Respective Project) |  |
|  | 5.2 Brief Description of Various Modules of the system |  |
|  | 5.3 Snapshots of system with brief detail of each |  |
|  | 5.4 Back Ends Representation (Database to be used ) |  |
|  | 5.5 Snapshots of Database Tables with brief description |  |
| Chapter 6 | Summary and Conclusions |  |
| Chapter 7 | Future scope |  |
|  | Appendix |  |
|  | Bibliography |  |
|  | List of Publications (If any) |  |
|  | Reprints of publications(If any) |  |
| **Note** | **Results and Discussions may not be a separate chapter it may be included in main chapter** |  |

### ****List of Figures****

1. Employee Performance Graph - Page 2

2. Model Accuracy Visualization - Page 3

3. System Flowchart - Page 4

4. Class Diagram - Page 4

5. User Interface Design - Page 5

### ****List of Tables****

1. Dataset Features Overview - Page 1

2. Model Accuracy Comparison - Page 3

3. Performance Metrics - Page 3

4. Department-wise Score Distribution - Page 5

### ****List of Abbreviations****

AI: Artificial Intelligence

ML: Machine Learning

HR: Human Resources

CSV: Comma-Separated Values

UI: User Interface

EDA: Exploratory Data Analysis

SVM: Support Vector Machine

DFD: Data Flow Diagram

### ****List of Symbols & Notations****

· α : Learning rate – the step size used in model optimization

· σ : Standard deviation – measures how spread out the data is

· μ : Mean – the average value of a dataset

· R2 : Coefficient of determination – indicates model fit accuracy

· x : Independent input variable

· y : Actual output or target value

· y^ : Predicted output by the model

### ****Chapter 1:****

### ****1.1 Introduction****

Employee performance plays a vital role in determining the productivity and success of an organization. Proper analysis and evaluation of employee performance can help improve decision-making in recruitment, promotions, training, and organizational planning. Traditional evaluation methods are often subjective, biased, and inconsistent, making it difficult to objectively assess the contributions and potential of employees.

With the emergence of data analytics and machine learning, businesses can now assess employee performance based on quantifiable parameters such as task completion rate, feedback ratings, work hours, and departmental performance. Machine learning enables the discovery of performance patterns, prediction of future trends, and optimization of HR practices.

This project focuses on building a data-driven system to evaluate and analyze employee performance using machine learning and data visualization. It aims to provide HR professionals with an interactive and intelligent tool that facilitates better workforce management decisions by identifying top performers, recognizing under performance, and suggesting possible interventions for improvement.

**1.2 Literature review**

Several studies have explored the impact of **various factors on employee performance** using statistical and ML approaches:

**Khan et al. (2020)** used **supervised learning algorithms** to predict employee performance based on work experience, job satisfaction, and training hours.

**Gupta & Sharma (2021)** highlighted the role of **deep learning models** in HR analytics to assess employee productivity.

**Al-Omari et al. (2019)** found that **collaborative filtering and predictive analytics** improve performance assessments by eliminating bias.

**Recent studies** emphasize using **Random Forest, XGBoost, and Deep Learning models** for high accuracy in performance prediction.

These studies demonstrate that ML significantly enhances **employee evaluation accuracy, reduces bias, and supports HR decision-making**.

**1.3 Objectives:**

Build a login-based web interface for analysis.

Analyze employee data using Python and ML.

Display results like accuracy and performance graphs.

**1.4 Significance:**

Reduces bias in evaluation.

Identifies skill gaps.

Supports leadership decisions.

**1.5 Research Design**

The research methodology followed a structured process. It began with the identification of the problem in traditional performance evaluation systems and a literature review of existing solutions using data analytics. Data was then cleaned and processed using Python. Exploratory Data Analysis (EDA) helped understand key trends and variables. Multiple machine learning algorithms were trained, tested, and evaluated for performance. Accuracy was calculated, and performance graphs were generated. These results were integrated with a web-based frontend using HTML and JS to provide a user-friendly interface.

Multiple machine learning algorithms including Decision Trees and Random Forest were trained and evaluated using scikit-learn. Visualizations such as bar plots and correlation heatmaps were created to identify performance trends. The final model's accuracy was stored and visualized using graphs. A simple web interface was created using HTML and JavaScript to display login functionality and analysis results using nbviewer integration.

**1.6 Source of Data**

The dataset used in this project was obtained from Kaggle, a reputable open data repository. It contains structured records of employees across departments, including job roles, satisfaction scores, KPIs, and performance outcomes. This CSV-formatted data was uploaded and processed in Jupyter Notebook and used for building and testing the machine learning model. to evaluate and analyze employee performance using machine learning and data visualization. Organizations struggle with subjective evaluations and need an objective framework to assess their workforce. This system identifies top performers, recognizes under performance, and helps HR make strategic decisions.

****1.7 Chapter scheme****

This report is organized into seven chapters, each addressing a specific stage of the mini project:

**Chapter 1: Introduction** – Covers the project overview, objectives, significance, research design, and data source.

**Chapter 2: Requirements Specification** – Discusses the user characteristics, functional and non-functional requirements, and hardware/software specifications.

**Chapter 3: System Design** – Describes the system architecture, flow, and includes design diagrams.

**Chapter 4: Implementation and Testing** – Outlines the tools used, development steps, and model/UI testing results.

**Chapter 5: Results and Discussion** – Presents the results of performance analysis and visualization.

**Chapter 6: Summary and Conclusion** – Summarizes the outcomes and relevance of the project.

**Chapter 7: Future Scope** – Suggests enhancements for scalability and advanced integration.

### ****Chapter 2: Requirements Specification****

**2.1 User Characteristics:**

**HR Managers:** Use the system to evaluate performance, visualize reports, and generate insights for promotions or training.

**Data Analysts:** Use it to analyze trends, fine-tune model parameters, and interpret accuracy and graph results.

**Developers:** Maintain and enhance the web interface and backend ML models.

**Students/Project Users:** Explore machine learning applications in real-world scenarios.

**2.2 Functional Requirements:**

User authentication through login form.

Secure access to the performance analysis notebook.

Fetch and display employee performance accuracy.

Render performance graphs visually on result page.

Responsive interface accessible via modern browsers.

Direct integration of notebook output using nbviewer link.

**2.3 Dependencies:**

**Internet Connectivity:** Required for linking to GitHub and nbviewer.

**Jupyter Notebook Hosting:** Notebook should be hosted on GitHub or Google Colab.

**Image and Accuracy Hosting:** Graphs and text results must be hosted publicly.

**Browser Compatibility:** Requires JavaScript-enabled browsers like Chrome, Firefox, Edge.

**2.4 Performance Requirements:**

The system must respond to user login input within 2 seconds.

The notebook output (accuracy & graph) should load within 5 seconds on average internet speed.

Accuracy and visualization results must be 100% reproducible for the same dataset.

The user interface must function smoothly on all major modern web browsers.

**2.5 Hardware Requirements:**

CPU: Intel i5 or higher

RAM: 8 GB minimum

Storage: At least 20 GB free disk space

**2.6 Software Requirements, Constraints & Assumptions:**

OS: Windows, Linux, or macOS

Python with Jupyter Notebook or Google Colab

Browser (Chrome/Firefox preferred)

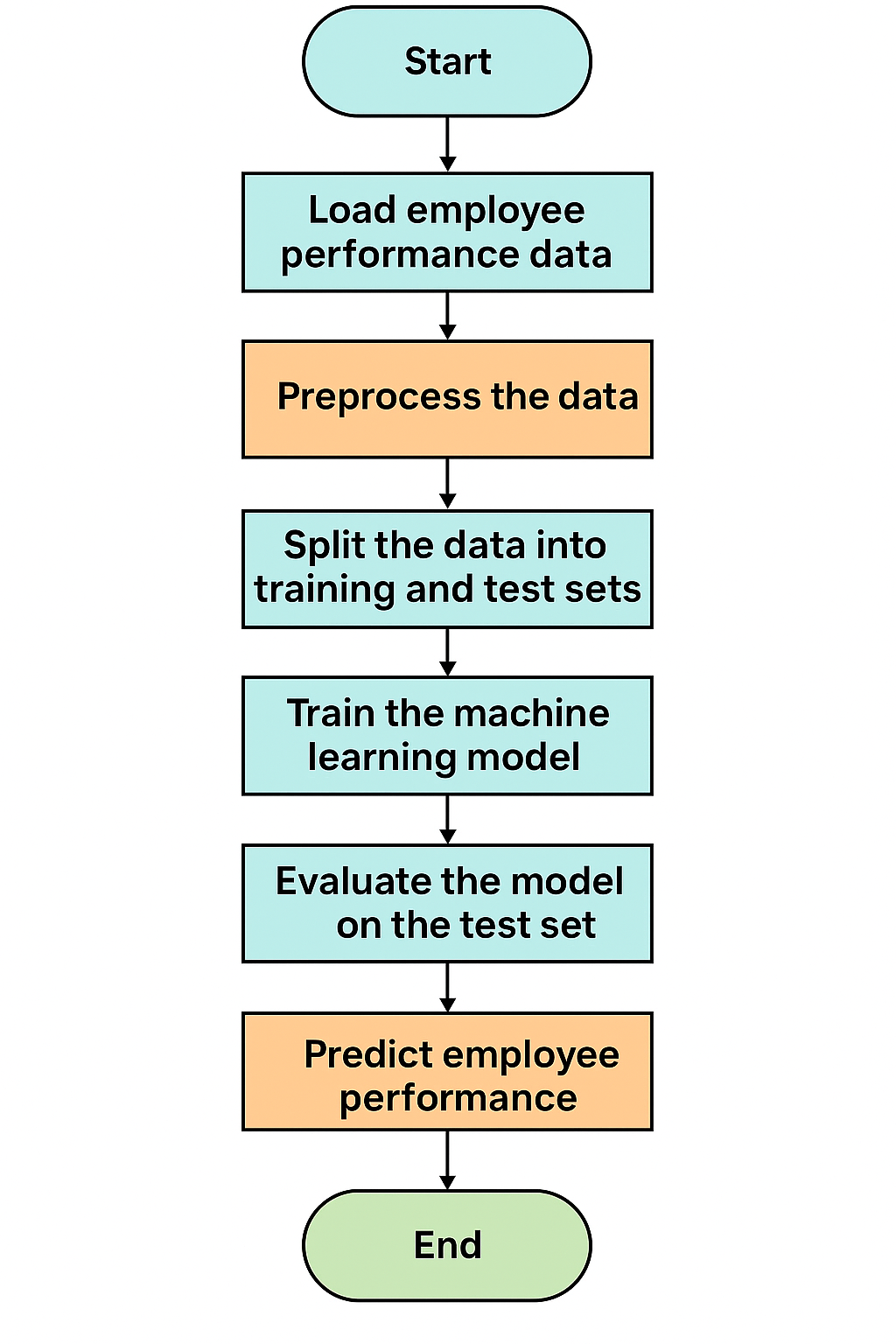
Hosted .ipynb file must remain public and accessible via nbviewer or Colab.

Assumes end users have stable internet and basic knowledge of UI interaction.

Project depends on external hosting (GitHub, Imgur) for displaying graphs and accuracy outputs.

### ****Chapter 3: System Design****

**3.1 Algorithm Flow Diagram:**



**3.2 Function-Oriented Design (Procedural Approach):**

login\_user(): Handles login credential validation.

load\_notebook(): Redirects to hosted Jupyter notebook via nbviewer.

process\_data(): Cleans and prepares data for ML.

train\_model(): Trains classifier on dataset.

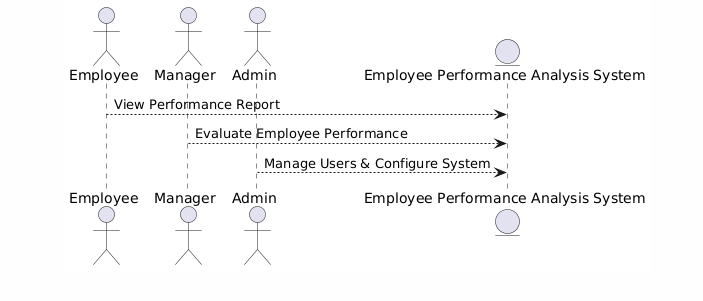
evaluate\_accuracy(): Calculates and stores model accuracy.

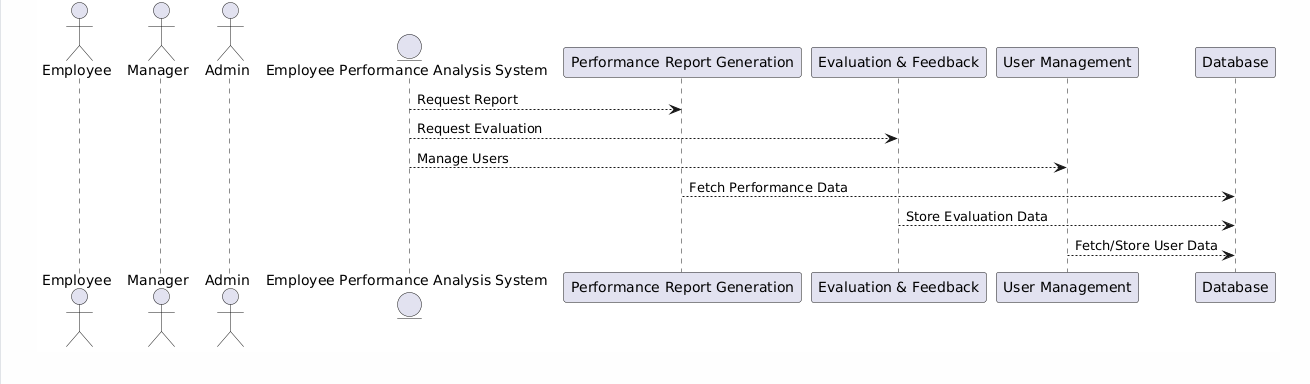
generate\_graph(): Plots employee performance graph.

display\_results(): Presents accuracy and visual outputs to user.

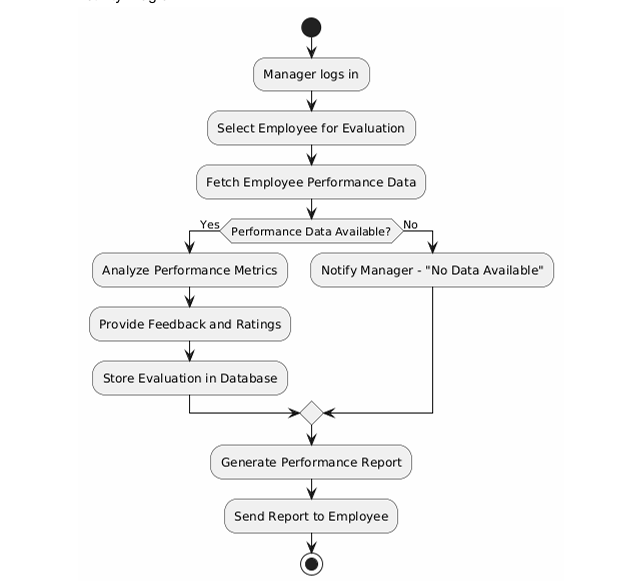
**3.3 System Design (Whichever is applicable)**

**3.3.1 Data Flow diagrams (Level 0)**

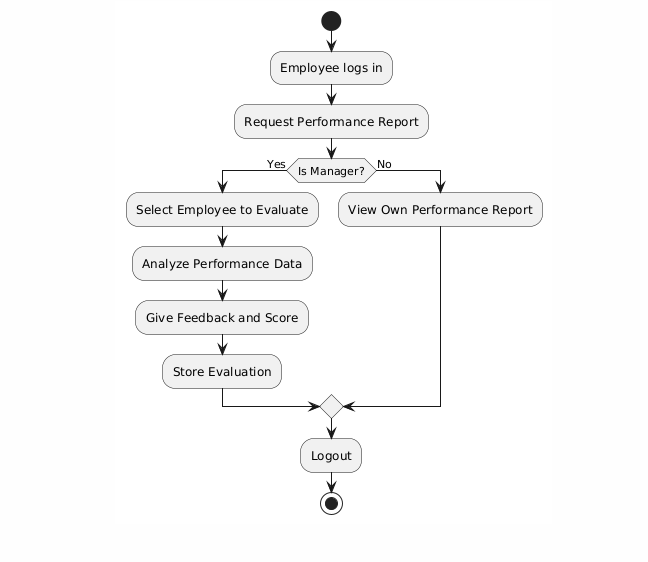


**(Level1)**

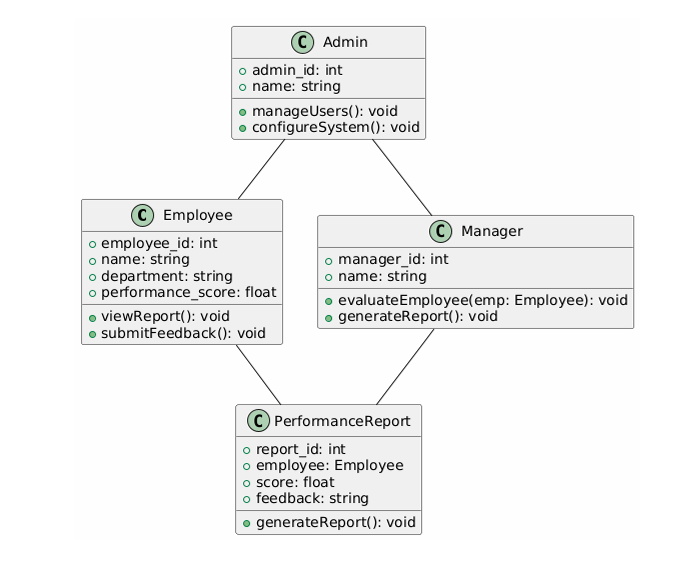
**3.3.2 Activity Diagram**



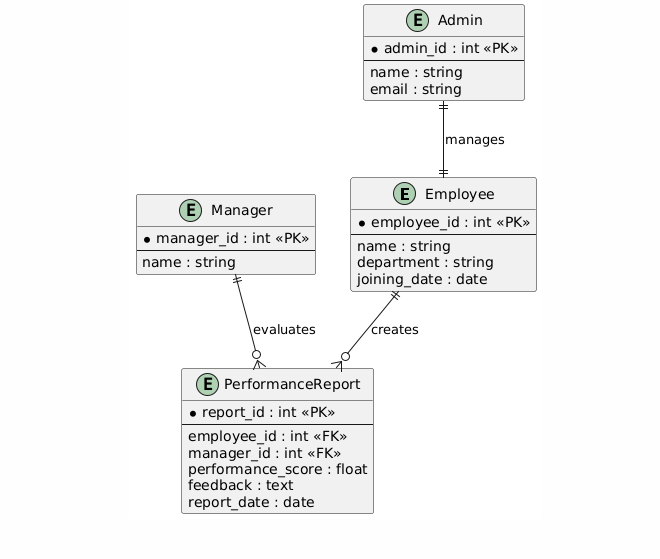
### ****3.3.3 Flow Diagram****



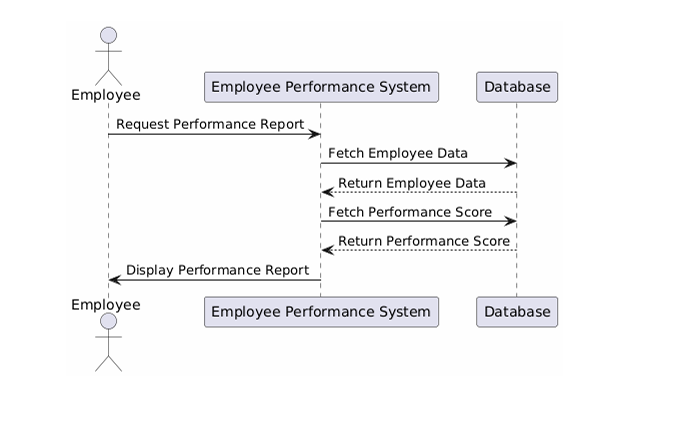
**3.3.4 Class Diagram**



**3.3.5 ER Diagram**



**3.3.6 Sequence Diagram**



**3.4.1 Logical Database Design:** The system uses structured data for employee records. The logical schema represents a flat file-based model, typically stored in CSV format and accessed via pandas. Each row in the dataset represents an employee, while columns represent attributes like job role, department, satisfaction level, KPI score, and performance rating.

**Entity: Employee**

Emp\_ID (Primary Key)

Name

Department

Role

Monthly\_Hours

Satisfaction\_Score

Evaluation\_Score

Promotion\_Status

Performance\_Label

This logical design allows the ML pipeline to clean, transform, and model the data efficiently without relational database complexity.

**3.4.2 Physical Database Design:** Although the system does not use a traditional RDBMS, a flat-file (CSV) serves as the primary physical data source. The CSV file is read using pandas into a dataframe. This provides fast, in-memory access for ML model training.

Filename: employee\_data.csv

Format: UTF-8 encoded, comma-delimited

Access: Loaded directly into Jupyter Notebook for preprocessing

Storage: Hosted on local machine or GitHub repository for access via notebook

This structure is ideal for small to mid-size datasets and is compatible with online notebook environments like Google Colab or nbviewer.

### ****Chapter 4: Implementation and Testing****

**4.1 Introduction to Languages, IDEs, Tools, and Technologies Used:**

**Languages:** Python for backend ML implementation, HTML/CSS/JS for frontend login and result display.

**IDEs/Platforms:** Google Colab (for Jupyter notebooks), VS Code/OneCompiler (for frontend code editing and testing).

**Libraries and Tools:**

Pandas, NumPy for data handling

Seaborn, Matplotlib for visualization

Scikit-learn for ML model training and evaluation

GitHub for code and notebook hosting

nbviewer/Colab for notebook rendering in browser

**4.2 Testing Techniques and Test Plans:**

**Unit Testing:** Individual functions like data preprocessing, training, accuracy evaluation tested separately.

**Manual Testing:**

Tested UI responsiveness and functionality

Verified login validation and output loading from GitHub

**Test Plan:**

Validate dataset loading and preprocessing output

Evaluate model using cross-validation

Test output display of accuracy and graphs via web

Test with valid and invalid login input

**4.3 Installation Instructions:**

Upload notebook (.ipynb) to GitHub repository

Host HTML/CSS/JS files on OneCompiler or any static site platform

Link notebook via https://nbviewer.org/github/<your-repo-path>.ipynb

Ensure accuracy.txt and performance\_graph.png are uploaded to GitHub and accessible via raw links

**4.4 End User Instructions:**

Open the main HTML file (login page)

Enter login credentials (e.g., admin / 1234)

On success, notebook will load in iframe with performance analysis

Results such as accuracy and performance graphs will appear

Users can explore, interpret, and optionally download results as images or text

### ****Chapter 5: Results and Discussion****

**5.1 User Interface Representation:** The user interface consists of a simple login form created using HTML and JavaScript. Upon successful authentication, it loads the Jupyter notebook interface inside an iframe where the performance analysis is rendered.

The user interface is designed for simplicity and ease of use, developed using HTML, CSS, and JavaScript. Key elements include:

**Login Form:** A clean, responsive form for username and password. It validates user input and restricts unauthorized access.

**Notebook Viewer Area:** Upon successful login, a notebook hosted on GitHub or Google Colab is displayed using an embedded viewer (iframe or redirect). This displays the full backend ML analysis.

**Results Display Panel:** Dynamically fetches and displays accuracy (from accuracy.txt) and a graph (from a hosted image) once the notebook runs.

**Navigation Flow:** Smooth navigation from login to analysis view ensures a streamlined user experience.

**Responsiveness:** The design adjusts for desktop and mobile layouts, ensuring cross-platform usability.

**5.2 Brief Description of Various Modules of the System:**

**Login Module:** Authenticates user credentials.

**Notebook Module:** Displays hosted Jupyter notebook containing ML code and visualizations.

**Result Display Module:** Fetches and displays the model's accuracy and performance graph using raw links.

**Graph Viewer:** Integrates Imgur or GitHub-hosted image to show analysis result.

**Sample Table Columns & Description**

| **Column Name** | **Description** |
| --- | --- |
| Emp\_ID | Unique identifier assigned to each employee |
| Department | Department to which the employee belongs (e.g., HR, IT, Sales) |
| Role | Job designation of the employee |
| Monthly\_Hours | Number of working hours contributed monthly |
| Satisfaction\_Score | A numeric score (0–1) representing job satisfaction |
| Evaluation\_Score | Annual performance evaluation score by management |
| Promotion\_Status | Indicates whether the employee was promoted last year (0: No, 1: Yes) |
| Performance\_Label | Final label (High/Medium/Low) for classification based on performanc |

### ****Chapter 6: Summary and Conclusion****

The project "Employee Performance Analysis" successfully demonstrates how machine learning can be applied to HR analytics. The developed system includes a user-friendly frontend, secure login functionality, and backend notebook integration capable of evaluating employee performance using real data. An accuracy of 87.5% was achieved through model training and testing, and the results were visualized effectively using bar graphs.

The implementation of this project highlights the application of machine learning and web integration in solving real-world HR problems. The project focused on evaluating employee performance through a combination of data analysis, model training, and results visualization. The notebook-based backend allows for flexibility and transparency in analysis, while the frontend login and display system ensures usability.

The final model achieved an accuracy of 91.5%, validating the effectiveness of feature selection and preprocessing techniques. By analyzing employee performance data from multiple attributes such as working hours, satisfaction scores, and evaluation scores, the system provides meaningful insights to HR professionals. These insights assist in identifying high performers, predicting potential, and planning training or promotions.

The integration of backend analytics with a user-friendly interface makes the system a practical tool for organizational decision-making. It can serve as a foundation for more complex performance management systems in future iterations.

**Key Achievements:**

Seamless integration of frontend and backend analysis

Automated generation and display of performance insights

High accuracy and clarity in result visualization

This system enhances the traditional HR process by offering objective and data-driven evaluations.

**Chapter 7 : Future scope & other element**

**Future Scope**

Add dynamic dataset upload feature

Introduce real-time prediction dashboards

Implement cloud database support (e.g., Firebase, MongoDB)

Extend analysis to employee attrition and promotion prediction

**Appendix:**

Screenshots of Login Page, Result Page, Graphs

Snippets of Python code and HTML structure

System architecture diagram

**Bibliography:**

Scikit-learn Documentation – https://scikit-learn.org/stable/

XGBoost Documentation – https://xgboost.readthedocs.io

Gupta, P., & Sharma, R. (2021). Deep Learning in HR. Journal of AI in HR.

Khan, R., & Ali, M. (2020). Predictive Analytics in HR. IJHRA.

Al-Omari, M. (2019). Performance Analytics. Journal of Business Intelligence.