# **Employee Productivity Monitoring System – MVP SRS**

|  |  |
| --- | --- |
| **Team Name** | **Kairo** |
| **Team Leader** | **Nishvan Kumar. M** |
| **Team Member** | **Sanjay Prakash. G** |
| **Team Member** | **Sai Raja Rajan. J. K** |
| **Team Member** | **Santhosh. S. K** |

## **Problem Statement**

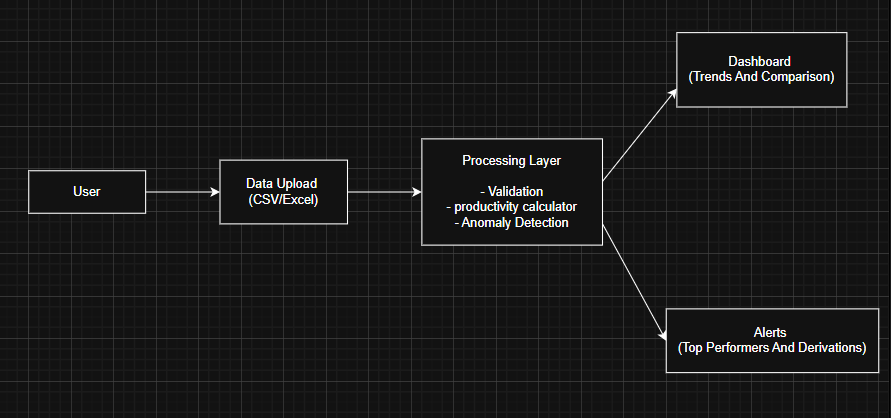
Organizations currently depend on manual or Excel-based methods for tracking employee productivity. This leads to inefficiency, errors, and lack of real-time insights. Managers struggle to quickly identify top performers or detect sudden drops in productivity.

The system aims to automate productivity tracking, highlight top performers, and detect unusual performance changes using a simple ML model, all through a secure web-based dashboard.

## **2. Requirement Analysis**

* Functional Requirements (MVP Scope)
  + Upload employee productivity data (CSV/Excel).
  + Calculate productivity scores using configurable formulas.
  + Provide time-based comparisons (weekly, monthly, quarterly).
  + Identify top 5 performers and send automated alerts.
  + Detect sudden performance deviations using ML-based anomaly detection.
  + Provide a secure web dashboard for visualization and reporting
* Non-Functional Requirements (MVP Scope)
  + Dashboard loads within 3 seconds for < 5,000 records.
  + File uploads supported up to 10MB.
  + System supports ~200 employees.
  + Login-based authentication with encrypted password storage.

**3. Data Flow Diagram (MVP)**

****

Input Layer:

User uploads CSV/Excel productivity data.

Processing Layer:

Data validation → Productivity score calculation → ML anomaly detection.

Output Layer:

Dashboard (visual trends, comparisons).

Alerts (Top 5 performers + anomaly notifications via email).

## **4. Model Selection & Justification**

Predictive Models Used: Linear Regression & Random Forest

**1. Linear Regression**

Type: Supervised Machine Learning algorithm.

Purpose: Predict Achieved output based on Target (numeric prediction).

Key Points:

Shows expected performance according to the Target.

Helps calculate performance gaps (Achieved − Predicted).

Works best when there’s a linear relationship between Target and Achieved.

**2. Random Forest**

Type: Supervised Machine Learning algorithm (ensemble of decision trees).

Purpose: Predict Achieved output like Linear Regression, but captures non-linear relationships.

Advantages:

Handles complex patterns better than Linear Regression.

Less sensitive to outliers.

Can provide feature importance, useful if multiple features are used.

### **5. Data Preprocessing**

1. **Data Collection**

Gather employee metrics: tasks completed, hours worked, quality scores, attendance.

1. **Data Cleaning**

Handle missing values, remove duplicates, fix inconsistencies.

1. **Feature Engineering**

Create derived features like productivity ratio, consistency score, trends.

1. **Data Transformation**

Normalize numerical data (Z-score), encode categorical variables.

1. **Outlier Check**

Remove extreme noise using simple statistical methods.

### **6. Model Training & Optimization**

**Models used:**

Isolation Forest (ML) and Z-score (basic stats).

**Training:**

Isolation Forest learns what normal productivity looks like.

Z-score checks if performance is much higher or lower than usual.

**Tuning:**

Adjust Isolation Forest settings (number of trees, expected anomalies).

Change Z-score threshold to catch more or fewer unusual cases.

**Check results:**

Look at flagged anomalies to see if they make sense.

### **Performance Evaluation & Metrics**

1. **Purpose:** Check how well the models detect unusual employee productivity.
2. **Metrics:**

**Precision:** How many flagged anomalies are actually real issues.

**Recall:** How many real anomalies were successfully detected.

**F1-Score:** Balance between precision and recall.

**Accuracy (optional):** Overall correctness if labeled data exists.

1. **Evaluation Steps:**

Compare model predictions with known anomalies (if available).

For unsupervised cases, review flagged cases manually.

Adjust model thresholds for better detection.