



International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 9.935

Volume 15, Issue 10, October 2026

BurnoutSense AI: Explainable Machine Learning for Employee Burnout Risk Prediction

Beena Jay Shri J, Keerthana M, M. Sasi Kala

U.G. Student, Department of Computer Science and Engineering, K.L.N. College of Engineering, Pottapalayam,
Tamil Nadu, India

U.G. Student, Department of Computer Science and Engineering, K.L.N. College of Engineering, Pottapalayam,
Tamil Nadu, India

Assistant Professor, Department of Computer Science and Engineering, K.L.N. College of Engineering, Pottapalayam,
Tamil Nadu, India

ABSTRACT: Employee burnout reduces wellbeing, productivity and retention, yet most organisations detect it only after performance declines or an employee resigns. This paper presents BurnoutSense AI, a proactive decision-support system that estimates employee burnout risk from measurable workplace attributes using machine learning and explainable artificial intelligence. On a Kaggle employee burnout dataset of 22,750 records (21,626 usable after cleaning), the system applies median imputation, label encoding and tenure-based feature engineering, and compares Linear Regression, Random Forest and XGBoost on an 80/20 train-test split. XGBoost is selected as the final model and predicts a continuous Burn Rate (0–1) with an R^2 of approximately 0.90, MAE of 0.047 and RMSE of 0.060; the score is mapped to Low, Medium or High risk. SHAP identifies the two strongest feature contributions for each employee, which are mapped to deterministic rule-based HR actions, while a Large Language Model is restricted to phrasing the explanation, with a template fallback. The system is deployed as a secured multi-user Streamlit dashboard with SHA-256 password hashing, organisation-wide analytics and real-time scoring of newly entered employees. The output is intended to support, not replace, human judgement in HR decisions.

KEYWORDS: Employee burnout, Machine learning, XGBoost, Explainable AI, SHAP, HR analytics, Streamlit dashboard.

I. INTRODUCTION

Employee burnout is a state of prolonged physical and emotional exhaustion that arises from sustained workload, work pressure and insufficient recovery. In modern organisations characterised by tight deadlines, distributed teams and continuous digital communication, burnout is associated with reduced productivity, lower job satisfaction, absenteeism and higher employee turnover. For Human Resource (HR) teams, the difficulty lies not only in recognising that burnout exists but in identifying the risk early enough for an intervention to be effective.

Conventional HR monitoring relies on periodic surveys, manager observation, performance indicators and exit information. These sources are largely reactive, may vary between managers, and are hard to apply consistently across a large workforce, especially when employee information is held in separate systems. Machine learning (ML) can estimate burnout risk from measurable attributes, but a bare numerical score does not tell an HR user why an employee is at risk or what should be done about it.

This paper presents **BurnoutSense AI**, an end-to-end decision-support system that predicts employee burnout risk, explains each prediction and recommends an HR action. The main contributions are: (i) a regression pipeline that predicts a continuous Burn Rate and compares Linear Regression, Random Forest and XGBoost on a common split; (ii) employee-level explanations using SHAP (SHapley Additive exPlanations); (iii) a deterministic rule-based action-mapping layer, with a Large Language Model (LLM) confined to phrasing the explanation and a template fallback when the API is unavailable; and (iv) a secured, multi-user Streamlit dashboard that provides organisation-wide analytics and real-time scoring of new employees from a persisted model artifact.

The remainder of the paper is organised as follows. Section II reviews related work. Section III describes the proposed methodology. Section IV presents the results, dashboard outputs and testing, and Section V concludes the paper with directions for future work.

II. LITERATURE SURVEY

Several studies apply ML to burnout prediction. Teodosio et al. [1] trained Support Vector Machine (SVM), Random Forest and K-Nearest Neighbours models on the HackerEarth Employee Burnout dataset, compared them using 30-fold cross-validation and the R^2 score, and deployed the best SVM model through a Streamlit interface. The work offers a usable interface but includes no explainable AI, deep learning or time-series trend prediction, so HR users receive a score without knowing which factors produced it. Syafiqah et al. [3] built occupational burnout models on the Maslach Burnout Inventory dataset using Decision Tree, SVM, K-Nearest Neighbours, Logistic Regression and Linear Discriminant Analysis, and evaluated them with cross-validation. Their comparison supports early intervention but provides no explainability, dashboard or recommendation module. Sharma and Gupta [4] trained Random Forest and XGBoost models and evaluated them using MAE, RMSE and R^2 ; the approach is accurate and scalable, but it does not explain individual predictions or provide recommendations and visualisation.

On the explainability side, Johnson and Williams [2] combined SHAP and LIME with ML models for HR analytics, improving transparency and trust by exposing the features that influence a prediction. However, the study concentrates on explanation alone; it offers neither an automated HR recommendation mechanism nor an integrated dashboard, leaving HR professionals to interpret the results and decide on actions manually.

TABLE I. COMPARISON OF RELATED WORK WITH THE PROPOSED SYSTEM

Work	Models	Explainable	Dashboard	HR action
Teodosio et al. [1]	SVM, RF, KNN	No	Yes	No
Johnson & Williams [2]	ML + SHAP, LIME	Yes	No	No
Syafiqah et al. [3]	DT, SVM, KNN, LR, LDA	No	No	No
Sharma & Gupta [4]	RF, XGBoost	No	No	No
BurnoutSense AI (proposed)	LR, RF, XGBoost + SHAP	Yes	Yes	Yes

The survey reveals a clear gap: existing systems provide prediction, or explanation, or an interface, but not all three together with an actionable recommendation. BurnoutSense AI addresses this gap by integrating a measured model comparison, local SHAP explanations, deterministic HR action mapping, a restricted LLM explanation layer and a secured dashboard in a single workflow.

III. METHODOLOGY

BurnoutSense AI follows a modular pipeline architecture (Fig. 1). Employee data passes through cleaning and feature engineering into three candidate regression models; the selected model is persisted together with its preprocessing components. At inference time the same transformations are replayed for a new employee, and the prediction is passed through the SHAP, action-mapping and language modules before being shown on the dashboard. Training-time and inference-time computation are therefore kept separate.

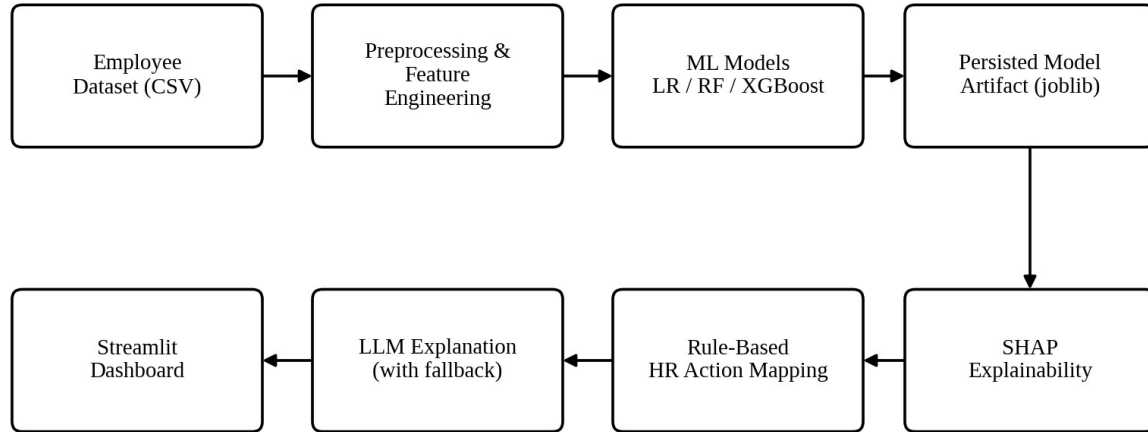


Fig. 1. Overall system architecture of BurnoutSense AI

A. Dataset

The system uses a publicly available Kaggle employee burnout dataset [9] with 22,750 records. Each record contains an employee identifier, Gender, Company Type, WFH Setup Available, Designation, Resource Allocation, Mental Fatigue Score, Date of Joining and the target Burn Rate, a continuous value between 0 and 1 that represents burnout severity (Table II). After rows with a missing Burn Rate are removed, 21,626 records remain for supervised learning.

TABLE II. DATASET ATTRIBUTES AND THEIR ROLE

Attribute	Role in the project
Employee ID	Identifier used in dashboard output
Gender, Company Type	Categorical attributes (label-encoded)
WFH Setup Available	Work-from-home availability (binary, label-encoded)
Designation	Employee designation, retained as an ordinal numeric value
Resource Allocation	Workload / resource allocation indicator
Mental Fatigue Score	Fatigue-related numeric feature
Date of Joining	Used to derive Days_of_Service
Burn Rate	Continuous target (0–1) representing burnout severity

B. Data Preprocessing

Preprocessing first inspects column names and types, counts missing values, checks the skewness of the target and examines correlations among numeric features for possible multicollinearity. Missing numeric feature values, such as those in Mental Fatigue Score and Resource Allocation, are filled by median imputation because the median is less affected by outliers than the mean. Records with a missing Burn Rate are removed, since a supervised regression model needs a known target during training. The cleaned dataset is saved and becomes the input to feature engineering.

C. Feature Engineering

Categorical fields (Gender, Company Type and WFH Setup Available) are label-encoded into integer codes, while Designation is kept as an ordinal number. Date of Joining is converted into Days_of_Service, a tenure measure computed against a reference date derived from the dataset, because a duration is more directly usable by a model than a raw calendar date. Two feature matrices are maintained: a matrix standardised with StandardScaler for Linear Regression, which is sensitive to feature magnitude, and an unscaled matrix for Random Forest and XGBoost, whose tree splits do not require standardisation. The fitted encoders and scaler are persisted so that exactly the same transformations are reproduced during real-time prediction.

D. Model Training and Selection

The cleaned data are divided into 80% training and 20% test partitions using a fixed random state, and three regressors are trained on comparable data. Linear Regression serves as a simple baseline. Random Forest is an ensemble of independently trained decision trees. XGBoost [5] is a gradient-boosted tree ensemble in which trees are added sequentially, each one reducing the residual errors left by its predecessors. Models are evaluated using the coefficient of determination (R^2), Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE):

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2} \quad (1)$$

$$MAE = (1/n) \sum |y_i - \hat{y}_i| \quad (2)$$

$$RMSE = [(1/n) \sum (y_i - \hat{y}_i)^2]^{1/2} \quad (3)$$

where y_i is the actual Burn Rate, $\hat{y}_i$ the predicted Burn Rate, $\bar{y}$ the mean actual value and n the number of test observations. Five-fold cross-validation is additionally used to inspect stability across data partitions. The final model is chosen on the basis of the strongest combination of high R^2 and low error rather than being assumed in advance.

E. Risk Categorisation

The continuous Burn Rate is retained as the risk score because it preserves more information than a category alone: two employees in the same category may have different scores. For dashboard presentation, the score is mapped to Low, Medium or High risk using thresholds derived from the distribution of predicted scores. The thresholds are stored with the model artifact so that new employees are categorised consistently.

F. SHAP-Based Explainability

SHAP [6], [7] assigns each feature a contribution to an individual prediction. For a tree model, the prediction is decomposed additively as

$$\hat{y} = \phi_0 + \sum_{j=1}^M \phi_j \quad (4)$$

where ϕ_0 is the base value and ϕ_j is the contribution of feature j . The system uses a SHAP TreeExplainer on the persisted XGBoost model, selects the two features with the largest absolute contributions for each employee, and records whether each contribution increases or decreases the predicted Burn Rate. This yields a local, employee-specific explanation rather than only a global feature-importance ranking.

G. Rule-Based HR Action Mapping and LLM Explanation

Responsibilities are deliberately separated: XGBoost predicts the score, SHAP identifies the drivers, a deterministic module selects the action and the LLM only phrases the explanation. The action-mapping module is a fixed lookup (Table III), so identical driving conditions always yield the same recommendation, which supports consistency and auditability. A fallback action is returned for any unmapped feature so that the function never fails.

TABLE III. DETERMINISTIC MAPPING FROM DRIVING FEATURE TO HR ACTION

Driving feature	HR action (project configuration)
Mental Fatigue Score	Wellbeing check-in; recommend workload reduction and time off
Resource Allocation	Review workload; redistribute tasks across the team
Designation	Discuss role expectations and career development
WFH Setup Available	Review remote-work support / flexible-work options
Days_of_Service	Tenure-related support and check-in
Any unmapped feature	General wellbeing check-in recommended

The LLM component uses a free-tier language-model API (for example Google Gemini or Groq). Its prompt contains only the employee's risk level and the main driving features, and its output is stored solely as the reason field; it never decides the risk score, risk category or action. If the API call fails because of a network error, rate limit or invalid key, a pre-written template sentence is returned, so language generation is an optional presentation layer and not a dependency

of core prediction. Generated explanations are cached for repeated combinations of risk level and top features to reduce API calls.

H. Real-Time Inference, Dashboard and Security

The trained XGBoost model, fitted encoders, scaler, feature-column order, risk thresholds and a background sample required by SHAP are serialised into a single joblib artifact. For a newly entered employee the system follows the ten-step workflow shown in Fig. 2 and returns `risk_score`, `risk_level`, `top_feature`, `reason` and `action` without retraining.

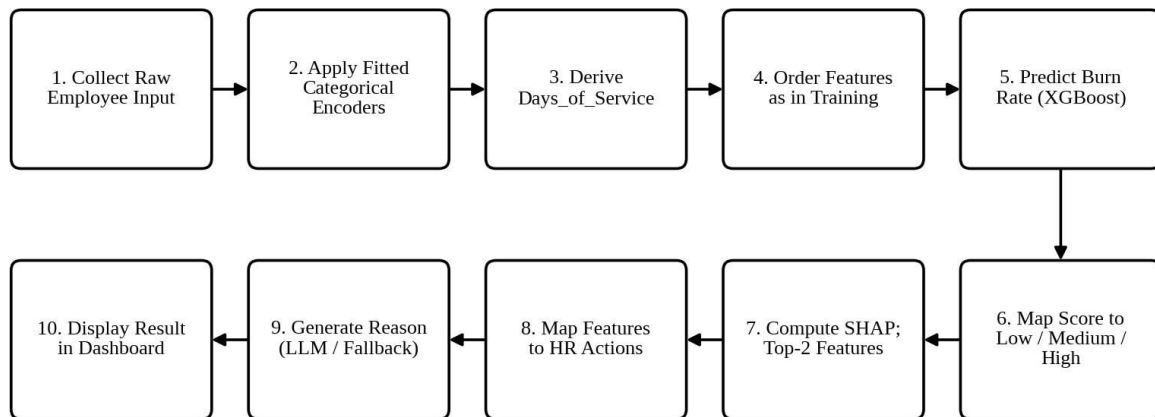


Fig. 2. Real-time prediction workflow for a newly entered employee

The presentation layer is built with Streamlit and Plotly. After login through an “HR Portal” screen, the Dashboard tab shows key performance indicator (KPI) cards, a risk-distribution donut chart, a bar chart of the most frequent drivers among high-risk employees, and an employee table that can be filtered by risk level, searched by employee ID and sorted by score. The Add New Employee tab provides a form for live prediction; the new record is appended to the prediction dataset and the cache is cleared so that analytics reflect it. Multiple HR accounts are supported: passwords are hashed with SHA-256 before being stored, the entered password is hashed and compared at login, Streamlit session state maintains the authenticated user, unauthenticated users are stopped from rendering any content, and logout clears the session.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

A. Experimental Setup

The system was implemented in Python 3 using pandas, NumPy, scikit-learn [8], XGBoost, SHAP, Streamlit, Plotly and joblib, developed in Visual Studio Code under Windows 10/11 on a standard machine (Intel i5/i7 or AMD Ryzen 5/7 processor, 8 GB RAM, SSD storage). The code is organised into separate modules for data cleaning, feature engineering, training, evaluation, SHAP explanation, action mapping, LLM reasoning, inference, user management and the dashboard.

B. Model Performance

Linear Regression, Random Forest and XGBoost were compared on the same 80/20 split, and XGBoost showed the strongest performance and was selected as the final model. Its held-out test results are given in Table IV. An R^2 of about 0.90 indicates that the model explains a large proportion of the variation in the held-out Burn Rate, while the MAE of 0.047 and RMSE of 0.060 show that typical errors are small on a 0–1 scale; the RMSE being only slightly above the MAE indicates the absence of many large errors. These results are specific to the dataset and evaluation setup used here and should not be taken as universal performance for other organisations.

TABLE IV. EXPERIMENTAL SETUP AND PERFORMANCE OF THE SELECTED XGBOOST MODEL

Item	Value
Original records	22,750
Usable records (after missing-target removal)	21,626
Train / test split	80% / 20%
Selected model	XGBoost
Test R ²	≈ 0.90
MAE	≈ 0.047
RMSE	≈ 0.060

C. Risk Distribution and Key Drivers

Access to the system is through the secured HR Portal login screen shown in Fig. 3, which is restricted to authorised HR personnel. After authentication, Fig. 4 shows the dashboard analytics. In the demonstration run the system scored 4,326 employee records and classified 1,491 (34.5%) as High, 1,538 (35.6%) as Medium and 1,297 (30.0%) as Low risk (Table V). Among high-risk employees, Mental Fatigue Score was by far the most frequent top driver, followed by Resource Allocation and, much less often, Designation. This agrees with the interpretation that fatigue and workload are the dominant signals of burnout in this dataset, with Designation, WFH Setup Available and Days_of_Service contributing for individual employees.

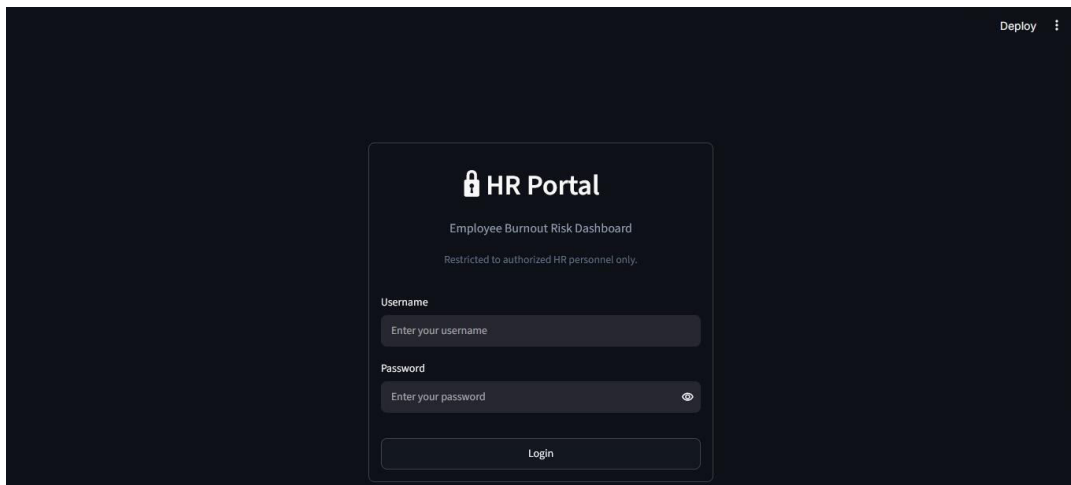


Fig. 3. HR Portal login screen of the BurnoutSense AI dashboard



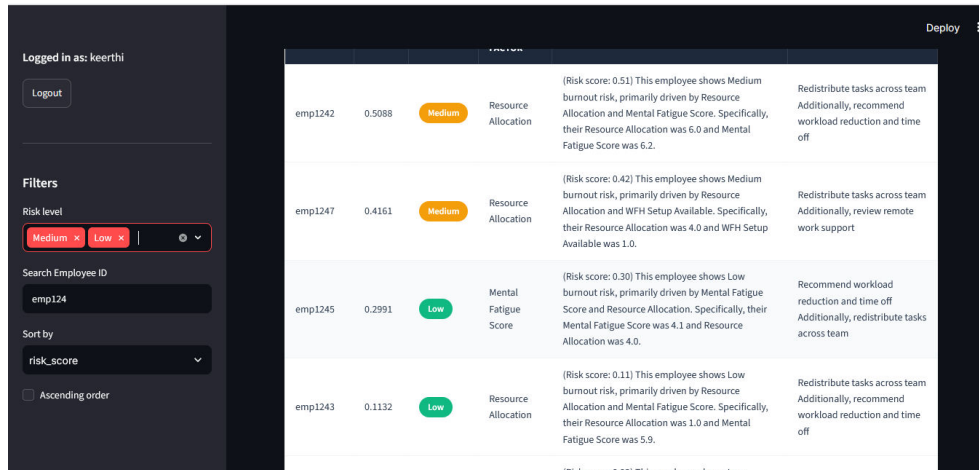
Fig. 4. Dashboard analytics: KPI cards, risk distribution and top drivers of high risk

TABLE V. RISK DISTRIBUTION OBSERVED ON THE DASHBOARD

Risk level	Employees	Share of workforce
High	1,491	34.5%
Medium	1,538	35.6%
Low	1,297	30.0%
Total	4,326	100%

D. Employee-Level Output

For every employee the dashboard combines the numerical score with the risk level, top feature, readable reason and recommended action (Fig. 5). For example, employee emp1242 received a score of 0.5088 (Medium risk), with Resource Allocation (6.0) and Mental Fatigue Score (6.2) as the driving features, and the system recommended redistributing tasks across the team. Employee emp1243, with a score of 0.1132 (Low risk), was likewise explained through Resource Allocation and Mental Fatigue Score. Searching for an employee ID with the risk-level filter set to Low returns a single matching record (1 of 4,326 employees): employee emp1231 with a score of 0.2649 (Low risk), Mental Fatigue Score (4.1) and Resource Allocation (2.0) as the driving factors, and the recommended action of workload reduction and time off together with redistribution of tasks across the team (Fig. 6). Because the explanation names the contributing factor and the action follows from a fixed rule, an HR user can see both why a score was produced and what step is suggested. The Add New Employee form (Fig. 7) returns the same five outputs immediately, with no retraining.



Employee ID	Risk Score	Risk Level	Top Driving Factor	Reason	Recommended Action
emp1242	0.5088	Medium	Resource Allocation	(Risk score: 0.51) This employee shows Medium burnout risk, primarily driven by Resource Allocation and Mental Fatigue Score. Specifically, their Resource Allocation was 6.0 and Mental Fatigue Score was 6.2.	Redistribute tasks across team. Additionally, recommend workload reduction and time off.
emp1247	0.4161	Medium	Resource Allocation	(Risk score: 0.42) This employee shows Medium burnout risk, primarily driven by Resource Allocation and WFH Setup Available. Specifically, their Resource Allocation was 4.0 and WFH Setup Available was 1.0.	Redistribute tasks across team. Additionally, review remote work support.
emp1245	0.2991	Low	Mental Fatigue Score	(Risk score: 0.30) This employee shows Low burnout risk, primarily driven by Mental Fatigue Score and Resource Allocation. Specifically, their Mental Fatigue Score was 4.1 and Resource Allocation was 4.0.	Recommend workload reduction and time off. Additionally, redistribute tasks across team.
emp1243	0.1132	Low	Resource Allocation	(Risk score: 0.11) This employee shows Low burnout risk, primarily driven by Resource Allocation and Mental Fatigue Score. Specifically, their Resource Allocation was 1.0 and Mental Fatigue Score was 5.9.	Redistribute tasks across team. Additionally, recommend workload reduction and time off.

Fig. 5. Employee-level table with risk score, level, top feature, reason and action

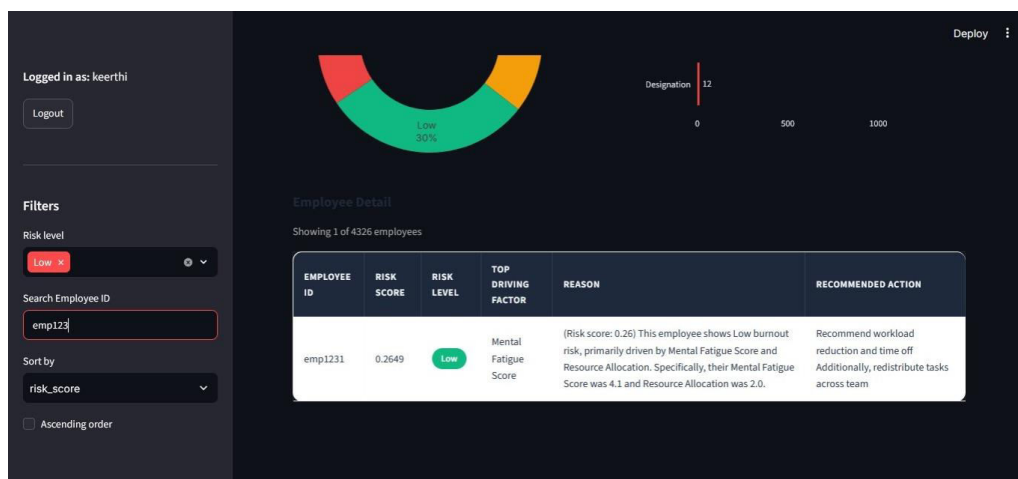
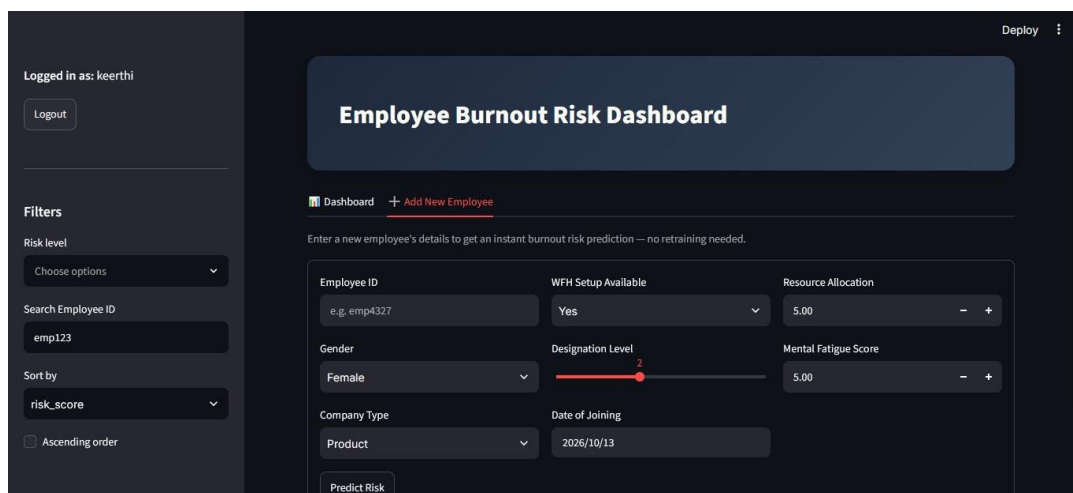


Fig. 6. Employee detail view filtered by risk level and employee ID search



Employee Burnout Risk Dashboard

Dashboard + Add New Employee

Enter a new employee's details to get an instant burnout risk prediction — no retraining needed.

Employee ID e.g. emp4327	WFH Setup Available Yes	Resource Allocation 5.00
Gender Female	Designation Level 2	Mental Fatigue Score 5.00
Company Type Product	Date of Joining 2026/10/13	

Predict Risk

Fig. 7. Add New Employee form for real-time burnout risk prediction

E. System Testing

Testing followed the modular structure of the system (Table VI). Unit tests covered missing-value detection, removal of rows with missing targets, encoding, Days_of_Service generation, inference, known and unknown action mapping, and correct and incorrect credentials. Functional tests covered login and logout, KPI display, risk filtering, employee search, sorting, the new-employee form, live prediction, explanation and action display. Integration tests followed the complete path from cleaning to the dashboard. The system was tested with multiple HR accounts and was reported to authenticate users, display analytics and score new employees in real time.

TABLE VI. TESTING STRATEGY AND COVERAGE

Testing level	Verification objective
Unit testing	Individual functions: cleaning, encoding, action mapping, authentication
Functional testing	Login/logout, KPIs, filter, search, sort, live prediction, explanation, action
Integration testing	Data hand-off from cleaning to training, persistence, inference, SHAP, action mapping, LLM and dashboard
Performance testing	Dashboard load, filtering and scoring a new employee without retraining
Security testing	Unauthenticated access blocked; invalid credentials rejected; no plain-text passwords; LLM-failure fallback
Usability testing	Clarity of risk categories, charts and plain-English explanations for non-technical HR users

F. Discussion and Limitations

The results show that predictive accuracy, interpretability and actionability can be combined in one deployable workflow. Keeping the LLM out of the decision path means that the score, category and action remain reproducible even when the language service is unavailable. Several limitations should be noted. The performance figures depend on the quality and representativeness of a single public dataset, and SHAP contributions describe model behaviour, not causal effects on burnout. The output is an HR analytics signal and not a clinical or psychological assessment, and it must not be used as an autonomous employment decision. Fairness across employee groups has not yet been evaluated. SHA-256 hashing and session-state authentication are adequate for an academic prototype, but a production system would need stronger password hashing, secret management, transport encryption, role-based access, audit logging and privacy governance.

V. CONCLUSION AND FUTURE WORK

This paper presented BurnoutSense AI, a complete applied ML workflow for employee burnout risk prediction. After data cleaning and feature engineering, XGBoost was selected over Linear Regression and Random Forest, achieving approximately 0.90 R^2 , 0.047 MAE and 0.060 RMSE on held-out data. SHAP explains each prediction, a deterministic rule base converts the dominant drivers into HR actions, and an LLM, confined to wording, makes the explanation readable. A secured multi-user Streamlit dashboard delivers workforce analytics and real-time scoring to HR users.

Future work includes an employee search view with individual history, alerting for employees who reach the High-risk category, continuous retraining and evaluation as new data arrive, historical trend analysis, department-level and role-level analytics, stronger authentication with role-based access, model monitoring and drift detection, and fairness and bias evaluation across employee groups before organisational deployment. Human review will remain central to all HR decisions.

REFERENCES

- [1] B. W. G. Teodosio, M. J. O. T. Lira, P. H. M. Araújo and L. R. C. Farias, "Support Vector Machine-Based Burnout Risk Prediction with an Interactive Interface for Organizational Use," arXiv preprint.
- [2] M. Johnson and P. Williams, "Explainable Artificial Intelligence in Human Resource Analytics," Springer Nature, vol. 8, no. 2, pp. 95–108, 2023.

- [3] N. Syafiqah et al., "Development of Occupational Burnout Prediction Models Using Machine Learning Techniques and Maslach Burnout Inventory," in Proc. IEEE 15th Control and System Graduate Research Colloquium (ICSGRC).
- [4] A. Sharma and R. Gupta, "Employee Burnout Prediction Using Machine Learning," International Journal of Intelligent Systems.
- [5] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in Proc. 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, 2016, pp. 785–794.
- [6] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in Advances in Neural Information Processing Systems, vol. 30, 2017, pp. 4765–4774.
- [7] S. M. Lundberg et al., "From local explanations to global understanding with explainable AI for trees," Nature Machine Intelligence, vol. 2, no. 1, pp. 56–67, 2020.
- [8] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.
- [9] "Employee Burnout Analysis Dataset," Kaggle, publicly available dataset for employee burnout prediction.
- [10] W. McKinney, "Data Structures for Statistical Computing in Python," in Proc. 9th Python in Science Conference, 2010, pp. 56–61.



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 9940 572 462  6381 907 438  ijirset@gmail.com



www.ijirset.com

Scan to save the contact details