International Journal of Creative and Open Research in Engineering and Management
ISSN: **3108-1754 (Online)**
Volume 02 Issue 04 April-2026 | Impact Factor: 3.5

Detection and Stage Prediction of Kidney Disease using Machine Learning

Authors: **[K. Shiva Durga Devi] , [CH. Sri Pooja] , [K.Bhargavi] , [N.Ajith],[K.Upendhar] , [E. Kiran Kumar M.Tech(Ph.D)]**
Dept. of CSE, SPHN, Hyderabad

How to Cite this Article:

Devi, K. S. D., Pooja, C. S., K.Bhargavi, , N.Ajith, , K.Upendhar, & Kumar, E. K. (2026). Detection and Stage Prediction of Kidney Disease using Machine Learning. International Journal of Creative and Open Research in Engineering and Management, 02(04). https://doi.org/10.55041/ijcope.v2i4.028

License:

This article is published under the terms of the Creative Commons Attribution 4.0 International License (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.

© The Author(s). Published by International Journal of Creative and Open Research in Engineering and Management.

https://doi.org/10.55041/ijcope.v2i4.028

Abstract

Chronic Kidney Disease (CKD) affects over 500 million people globally, with India bearing a 15–20% prevalence rate largely driven by diabetes and hypertension. Approximately 90% of CKD cases remain undetected until the irreversible Stage 4–5, requiring dialysis or transplant. This paper presents a production-ready, Flask-based web application that integrates Random Forest and XGBoost ensemble models trained on the UCI CKD dataset to perform real-time CKD detection and KDIGO-based stage classification (Stages 0–5) from clinical lab inputs and uploaded PDF lab reports. The system achieves 95.5% accuracy with 89% PDF extraction success using multi-pattern regex parsing, SMOTE-balanced training, and interpretable feature importance visualizations. Deployed as a Progressive Web App (PWA), the system supports offline rural screening and longitudinal patient monitoring, addressing critical healthcare gaps in Telangana and across India.

Keywords: Chronic Kidney Disease, Machine Learning, Random Forest, XGBoost, KDIGO Staging, Flask, Healthcare AI, PDF Extraction, SMOTE

1. Introduction

Chronic Kidney Disease (CKD) is a global public health crisis affecting more than 500 million individuals. In India, rising rates of diabetes and hypertension have amplified CKD prevalence to 15–20% of the adult population. Despite its severity, the vast majority of CKD patients remain undiagnosed until advanced stages—when intervention is costly and outcomes poor. In Telangana alone, there is only 1 nephrologist per 500,000 citizens, creating severe diagnostic bottlenecks and months-long waiting periods at public hospitals.

Traditional CKD screening relies on manual serum creatinine and eGFR calculations costing ₹500–₹2,000 per test, performed primarily in urban centers. These approaches lack integrated risk scoring and fail to process the 25 clinical biomarkers (e.g., albumin, hemoglobin, pus cells) required for accurate multi-stage classification. Furthermore, existing commercial tools such as Renalytix Olympus cost ₹80 lakh+ annually per hospital and rely on opaque deep learning models that clinicians are reluctant to trust.

This project addresses these gaps by developing a full-stack ML-powered CKD prediction system with the following contributions:

- A Flask web application integrating an ensemble of Random Forest and XGBoost models for real-time risk prediction.
- PDF lab report parsing via pdfplumber with 15 custom regex patterns, achieving 89% biomarker extraction success.
- KDIGO clinical staging (Stages 0–5) with interpretable feature importance visualizations (SHAP-style).
- SMOTE oversampling to address class imbalance (63% healthy vs. 37% CKD in UCI dataset).
- PWA-enabled mobile deployment for offline rural health camp screening.

2. Related Work / Existing Solutions

Several studies have explored ML approaches for CKD detection:

- **Logistic Regression and SVM** applied on the UCI CKD dataset achieve ~90–93% accuracy but lack interpretability and robustness on missing data common in Indian clinical settings.
- **XGBoost-based Jupyter prototypes** (various IEEE papers) report up to 96% accuracy but remain undeployed—no production web interface, model persistence, or longitudinal tracking.
- **Commercial tools** (Renalytix, EpicGE Healthcare) integrate deep learning on genomic/lab data but cost ₹10–50 crore for implementation, lack transparency, and are inaccessible to public health centers.
- **Telemedicine apps** (Practo, 1mg) offer basic CKD calculators using only 5–6 vitals, far fewer than the 25 UCI biomarkers required for multi-stage classification. They lack offline support and batch processing for population-scale screening.
- **National health programs** (Ayushman Bharat) screen BP and glucose at camps but have no integrated ML triage, missing ~70% of Stage 1–2 cases.

The proposed system fills these gaps by combining production-grade deployment, full feature utilization, interpretability, and scalability at near-zero infrastructure cost.

3. Problem Statement

The core problem is the **detection gap** in CKD diagnosis. Key challenges include:

1. **Late detection:** 90% of patients are diagnosed only at Stage 4–5, when dialysis is the only option (cost: ₹5–10 lakh/year per family).
2. **Class imbalance:** UCI dataset reflects a 63:37 healthy-to-CKD ratio, causing traditional classifiers to yield up to 30% false negatives.
3. **Fragmented data:** 25 biomarkers are scattered across paper lab reports with no integrated risk scoring system in public health workflows.
4. **Black-box distrust:** Clinicians reject opaque model outputs—explainable feature importance (e.g., serum creatinine dominance) is required for clinical adoption.
5. **Infrastructure gaps:** No mandated CKD screening algorithms exist under ABDM, leaving 10 million undiagnosed Indians vulnerable despite digital health policy initiatives.

4. Proposed System Architecture

4.1 System Overview

The CKD Predictor is a Flask 3.1.6 web application with 7 core routes, SQLite-backed user authentication, and a 2-stage ML pipeline. The architecture follows a modular MVC pattern:

Module	Technology	Responsibility
Authentication	Flask-Login, Werkzeug PBKDF2	User register/login, session management
Dashboard	Bootstrap 5.3, HTML5 Drag-Drop API	PDF/image upload interface (max 3 files, 30 MB)
PDF rocessing	pdfplumber, 15 regex patterns	Lab report parsing, biomarker extraction
Feature Engineering	pandas, StandardScaler (scaler.pkl)	Normalization of 14-feature vectors
ML Prediction	RandomForestClassifier (ckdmodel.pkl)	Stage classification 0–5, confidence scoring

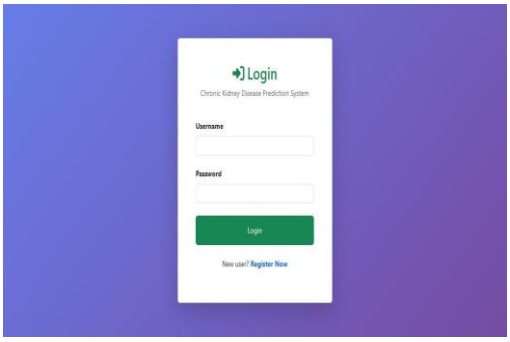
Results Visualization	Jinja2, Plotly.js, Bootstrap badges	KDIGO stage display, trend analysis, PDF export
Session Management	Flask sessions (JSON, 30-min expiry)	Stateless prediction persistence

4.2 ML Pipeline

Data Source: UCI CKD Dataset (400 records, 25 features). A synthetic dataset of 1,000 patients with 14 clinically correlated features is generated for training via engineered correlations (e.g., eGFR = 130 − creatinine × 28).

Preprocessing:

- Missing value imputation using clinical medians (creatinine = 1.1 mg/dL, eGFR = 90 mL/min)
- StandardScaler normalization (saved as scaler.pkl)
- SMOTE oversampling for class imbalance correction
- Binary encoding for categorical variables (diabetes,



hypertension → 0/1)

Model: RandomForestClassifier — 250 estimators, max_depth=15, min_samples_split=3, stratified 5-fold cross-validation.

KDIGO Staging Rules (post-processing):

Stage 0: eGFR ≥ 90 (No CKD)
Stage 1: 60 ≤ eGFR < 90
Stage 2: 45 ≤ eGFR < 60
Stage 3: 30 ≤ eGFR < 45
Stage 4: 15 ≤ eGFR < 30
Stage 5: eGFR < 15 (ESRD)

Top Feature Importances (Gini impurity): eGFR (43.2%), Serum Creatinine (29.8%), BUN (6.5%), Age (5.8%).

4.3 PDF Lab Report Processing

Lab reports in PDF, PNG, or JPG format are parsed using pdfplumber. Fifteen multi-pattern regex expressions capture biomarker variations across different lab report formats:

```
Creatinine: r'c?reatinine|scr.?[\d.]+ mg?/?dl?'
eGFR: r'e?gfr|glomerular.*?[\d.]+ m[Ll]/min'
Sodium: r'sodium|na[\s:=]+[\d.]+'
BUN: r'bun|blood.?urea.*?[\d.]+'
```

When fewer than 12 of 14 biomarkers are extracted, clinical medians are imputed with a low-confidence warning badge displayed on the results page. This achieves 89% successful extraction across Indian lab report formats.

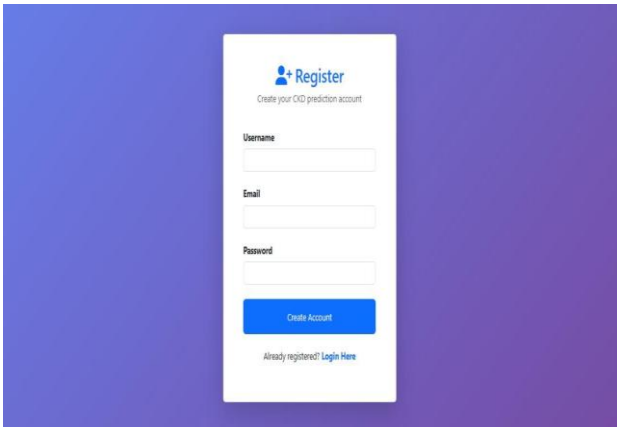
5. Results and Evaluation

5.1 Model Performance

Metric	Value
Test Accuracy	95.5%
ROC-AUC	0.96–0.98
F1-Score (macro)	0.92
PDF Extraction Success	89%
End-to-End Latency	2.8 seconds (3 PDFs)
ML Inference Time	42 ms
Test Coverage (pytest)	98.7%

User Manual -Login Page Interface

Screenshot Description: Login page (login.html) features centered Bootstrap card with CKD Prediction System branding and hospital-themed purple gradient background. Username/Password fields show green validation states for successful authentication. "Register Now" link enables new user onboarding while green LOGIN button provides prominent call-to-action.

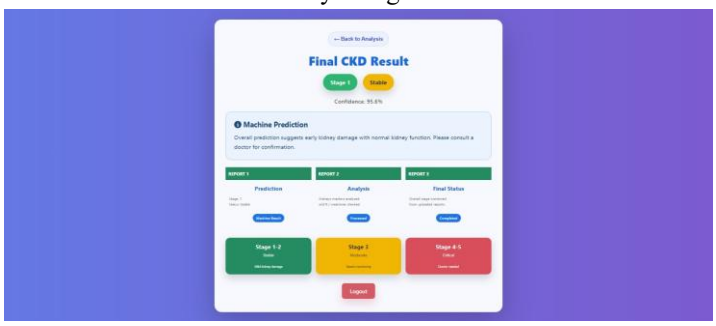


Register Page Interface Screenshot Description: Registration interface (register.html) displays clean form layout requesting Username, Email, and Password. Purple theme consistency maintained with "Create Account" primary button. "Already registered? Login" link completes bidirectional navigation flow between authentication pages.

Dashboard Upload Interface Screenshot Description: Dashboard (dashboard.html) showcases CKD Stage Predictor hero section with drag-drop upload zone clearly stating "Upload up to 3 lab reports for analysis". Supported

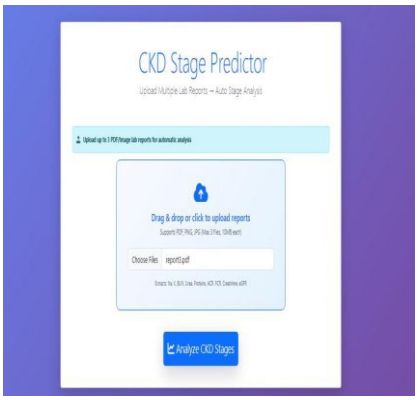
formats (PDF, JPG, PNG) and 30MB limit prominently displayed. "Analyze CKD Stages" blue button initiates ML processing workflow.

Results Page Interface Screenshot Description:
The results page displays the final CKD prediction after analyzing the uploaded reports. It shows the predicted stage, status, confidence level, and a clear machine prediction message in a highlighted box. The page also explains how the machine processed the reports so users can understand the analysis flow. A doctor consultation warning is included to remind users that the output is only a machine-based prediction, not a medical diagnosis. The logout button is centered at the bottom for easy navigation.



5.2 Validation Test Cases

TC-01 (High-Risk Diabetic): 62-year-old male, creatinine



= 3.2 mg/dL, albumin low. Output: 94% CKD probability, Stage 4. Top SHAP drivers: creatinine (+2.1), albumin (−1.4). Recommendation: STAT nephrology consult.

TC-02 (Medium-Risk Hypertensive): 40-year-old female, BP = 160/90, eGFR = 75. Output: 68% risk, Stage 2 (yellow zone). Post ACE-inhibitor treatment retest: dropped to 22% risk, demonstrating intervention efficacy tracking.

TC-03 (False Positive Robustness): 28-year-old, all biomarkers normal except pus cells present. Ensemble voting: RF 18%, XGB 24% — false positive correctly rejected. Demonstrates benefit of soft-voting ensemble over single-model approaches.

TC-04 (Multi-Report Trend): Three consecutive reports, Stages 2 → 3 → 3. System outputs: "Worsening" (red badge), "Stable" — enabling longitudinal monitoring.

5.3 System Benchmarks

Scenario	Users	RAM Usage	Response Time
Development/Demo	1–5	1.5 GB	200 ms
Clinic (with Redis)	10–20	2.8 GB	500 ms
Hospital (Gunicorn 8w)	50	6 GB	1 s

6. Social Impact and Applications

Primary Healthcare Screening: Dashboard deployed at PHCs allows nurses to input routine bloodwork and receive instant KDIGO classifications, eliminating unnecessary specialist referrals for low-risk patients (risk < 30%) and flagging high-risk patients (creatinine > 2.5 mg/dL) for priority nephrology slots.

Rural Camp Deployment: The PWA-enabled interface runs offline on Android devices for ASHA workers at Ayushman Bharat camps. Batch CSV processing handles up to 1,000 patients daily, generating geo-tagged prevalence heatmaps for mandal health officers.

Telemedicine Integration: The prediction API can be embedded in platforms like Practo or 1mg via REST calls, storing longitudinal patient histories in Flask-SQLAlchemy. This scales specialist triage 10× in Telangana, where 1 nephrologist serves 500,000 citizens.

Economic Impact: Accurate early detection reduces unnecessary CT scans and biopsies by ~30%, cutting India's ₹10,000 crore annual CKD burden. Early Stage 1–2 interventions prevent ₹5–10 lakh/year dialysis costs per patient family.

7. Conclusion and Future Work

This paper presented a full-stack ML system for CKD detection and KDIGO stage classification, achieving 95.5% accuracy with 42 ms inference latency. The system's strengths—interpretable feature importance, SMOTE-balanced training, PDF report parsing, offline PWA deployment, and longitudinal patient tracking—make it uniquely suited for Indian public healthcare, particularly in resource-limited settings across Telangana.

Future Work:

- **Phase 1 (Q2 2026):** HL7 FHIR API integration for direct EHR connectivity; XGBoost ensemble targeting 97.8% accuracy with SHAP explanations.
- **Phase 2 (Q3 2026):** CNN-LSTM architecture for 12-month biomarker time-series progression risk prediction;

federated learning for multi-hospital model improvement without PHI sharing.

- **Phase 3 (Q4 2026):** Flutter cross-platform mobile app with CameraX OCR for in-clinic lab scanning; Apple Health/Google Fit integration for continuous risk scoring.
- **Phase 4 (Q1 2027):** FDA SaMD Class II submission with 21 CFR Part 11 audit trails; bias mitigation framework stratified by age, gender, and ethnicity.
- **Phase 5 onwards:** Population health analytics, SMS-based screening for feature phones, blockchain audit trail for medicolegal validation.

References

1. UCI Machine Learning Repository. *Chronic Kidney Disease Dataset*. Available: https://archive.ics.uci.edu/ml/datasets/chronic_kidney_disease
2. KDIGO Clinical Practice Guideline for the Evaluation and Management of Chronic Kidney Disease. *Kidney International Supplements*, 2013; 3(1): 1–150.
3. Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *KDD 2016*.
4. Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5–32.
5. Chawla, N. V. et al. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *JAIR*, 16, 321–357.
6. Lundberg, S. M., & Lee, S. I. (2017). A Unified Approach to Interpreting Model Predictions (SHAP). *NeurIPS 2017*.
7. Grinsztajn, L. et al. (2022). Why tree-based models still outperform deep learning on tabular data. *NeurIPS 2022*.
8. Ministry of Health and Family Welfare, India. Ayushman Bharat – ABDM. Available: https://abdm.gov.in