Model Optimization and Tuning Phase Report

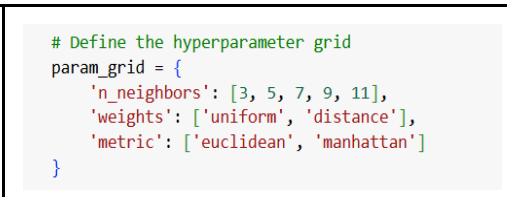
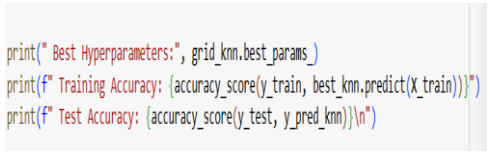
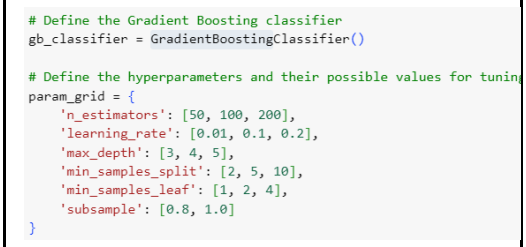
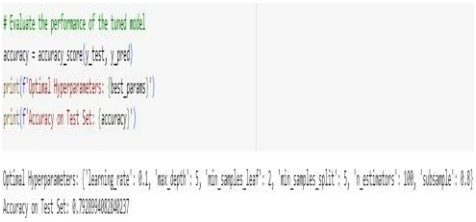
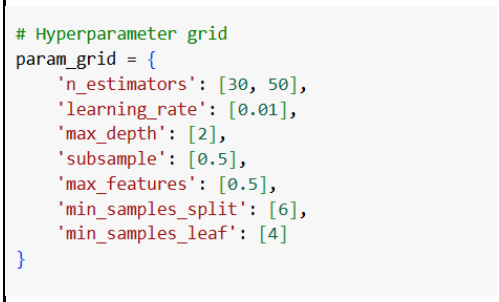
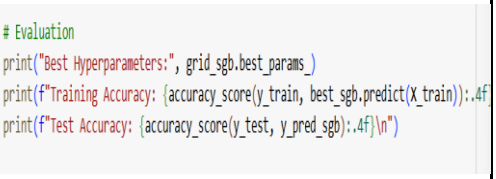
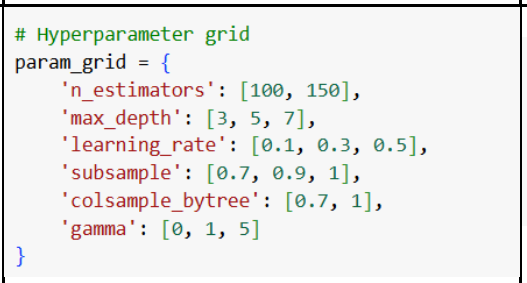
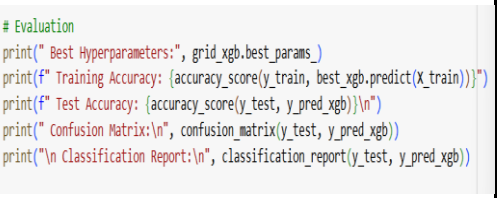
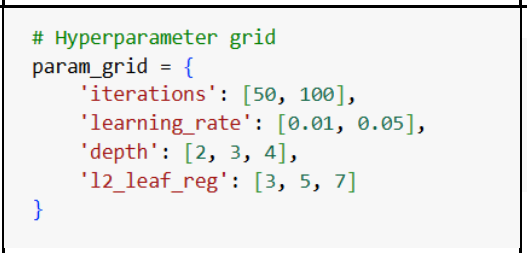
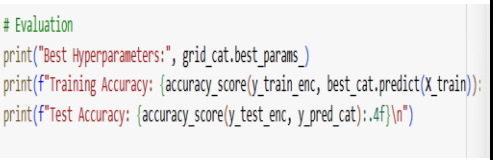
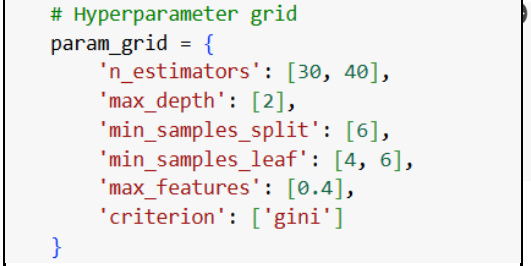
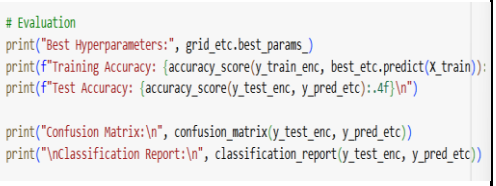
Date	2 July 2025
Team ID	SWTID1749634408
Project Title	Early Prediction for Chronic Kidney Disease Detection: A Progressive Approach to Health Management
Maximum Marks	10 Marks

Model Optimization and Tuning Phase

The Model Optimization and Tuning Phase involves refining machine learning models for peak performance. It includes optimized model code, fine-tuning hyperparameters, comparing performance metrics, and justifying the final model selection for enhanced predictive accuracy and efficiency.

Hyperparameter Tuning Documentation (6 Marks):

Model	Tuned Hyperparameters	Optimal Values
Decision Tree	`grid_param = { 'criterion' : ['gini', 'entropy'], 'max_depth' : [3, 5, 7, 10], 'splitter' : ['best', 'random'], 'min_samples_leaf' : [1, 2, 3, 5, 7], 'min_samples_split' : [2, 3, 5, 7], 'max_features' : ['sqrt', 'log2', None] }`	`print(f"Training Accuracy of Decision Tree Classifier is {accuracy_score(y_train, ...` `print(f"Test Accuracy of Decision Tree Classifier is {dtc_acc} \n")` `print(f"Confusion Matrix :- \n{confusion_matrix(y_test, y_pred_dtc)}\n")` `print(f"Classification Report :- \n {classification_report(y_test, y_pred_dtc)}")`
Random Forest	`# Hyperparameter grid param_grid = { 'n_estimators': [50, 100], 'max_depth': [3, 5, 7], 'max_features': ['sqrt'], 'min_samples_split': [4, 6], 'min_samples_leaf': [2, 4], 'criterion': ['gini'] }`	`# Evaluation print("Best Hyperparameters:", grid_rf.best_params_) print(f"Training Accuracy: {accuracy_score(y_train, best_rf.predict(X_train)):.4f}" print(f"Test Accuracy: {accuracy_score(y_test, y_pred_rf):.4f}\n") print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred_rf)) print("\nClassification Report:\n", classification_report(y_test, y_pred_rf))`

KNN		
Gradient Boosting		
Stochastic Gradient Boosting		
XGboost		
CATbooost		
Extra Trees		

LGBM Classifier	`# Hyperparameter grid`
	`param_grid = {`
	`'n_estimators': [50, 100],`
	`'learning_rate': [0.01, 0.05],`
	`'max_depth': [2, 3],`
	`'num_leaves': [7, 15],`
	`'min_child_samples': [15, 30],`
	`'subsample': [0.6, 0.8],`
	`'colsample_bytree': [0.6, 0.8]`
	`}`
		`# Evaluation`
		`print("Best Hyperparameters:", grid_lgbm.best_params_)`
		`print(f"Training Accuracy: {accuracy_score(y_train_enc, best_lgbm.predict(X_train))`
		`print(f"Test Accuracy: {accuracy_score(y_test_enc, y_pred_lgbm):.4f}\n")`
		
		`print("Confusion Matrix:\n", confusion_matrix(y_test_enc, y_pred_lgbm))`
		`print("\nClassification Report:\n", classification_report(y_test_enc, y_pred_lgbm)`

Performance Metrics Comparison Report (2 Marks):

Model	Optimized Metric
Decision Tree	Classification Report: precision recall f1-score support 0 1.00 0.98 0.99 52 1 0.97 1.00 0.98 28 accuracy 0.99 80 macro avg 0.98 0.99 0.99 80 weighted avg 0.99 0.99 0.99 80 Confusion Matrix: [[51 1] [0 28]]
Random Forest	Classification Report: precision recall f1-score support 0 1.00 0.98 0.99 52 1 0.97 1.00 0.98 28 accuracy 0.99 80 macro avg 0.98 0.99 0.99 80 weighted avg 0.99 0.99 0.99 80 Confusion Matrix: [[51 1] [0 28]]
KNN	Classification Report: precision recall f1-score support 0 1.00 0.98 0.99 52 1 0.97 1.00 0.98 28 accuracy 0.99 80 macro avg 0.98 0.99 0.99 80 weighted avg 0.99 0.99 0.99 80 Confusion Matrix: [[51 1] [0 28]]

XGBoost Classifier	Classification Report: precision recall f1-score support 0 1.00 0.98 0.99 52 1 0.97 1.00 0.98 28 accuracy 0.99 80 macro avg 0.98 0.99 0.99 80 weighted avg 0.99 0.99 0.99 80 Confusion Matrix: [[51 1] [0 28]]
ADA Boost Classifier	Classification Report: precision recall f1-score support 0 1.00 0.98 0.99 52 1 0.97 1.00 0.98 28 accuracy 0.99 80 macro avg 0.98 0.99 0.99 80 weighted avg 0.99 0.99 0.99 80 Confusion Matrix: [[51 1] [0 28]]
Gradient Boost	Classification Report: precision recall f1-score support 0 1.00 0.98 0.99 52 1 0.97 1.00 0.98 28 accuracy 0.99 80 macro avg 0.98 0.99 0.99 80 weighted avg 0.99 0.99 0.99 80 Confusion Matrix: [[51 1] [0 28]]
Stochastic Gradient Boosting	Classification Report: precision recall f1-score support 0 1.00 0.98 0.99 52 1 0.97 1.00 0.98 28 accuracy 0.99 80 macro avg 0.98 0.99 0.99 80 weighted avg 0.99 0.99 0.99 80 Confusion Matrix: [[51 1] [0 28]]

CAT Boost Classifier	Classification Report: precision recall f1-score support 0 1.00 0.98 0.99 52 1 0.97 1.00 0.98 28 accuracy 0.99 80 macro avg 0.98 0.99 0.99 80 weighted avg 0.99 0.99 0.99 80 Confusion Matrix: [[51 1] [0 28]]
Extra Trees Classifier	Classification Report: precision recall f1-score support 0 1.00 1.00 1.00 52 1 1.00 1.00 1.00 28 accuracy 1.00 80 macro avg 1.00 1.00 1.00 80 weighted avg 1.00 1.00 1.00 80 Confusion Matrix: [[52 0] [0 28]]

Final Model Selection Justification (2 Marks):

Final Model	Reasoning
Stochastic Gradient Boosting	The Stochastic Gradient Boosting model was selected for its superior performance, exhibiting high accuracy during hyperparameter tuning. Its ability to handle complex relationships, minimize overfitting, and optimize predictive accuracy aligns with project objectives, justifying its selection as the final model.