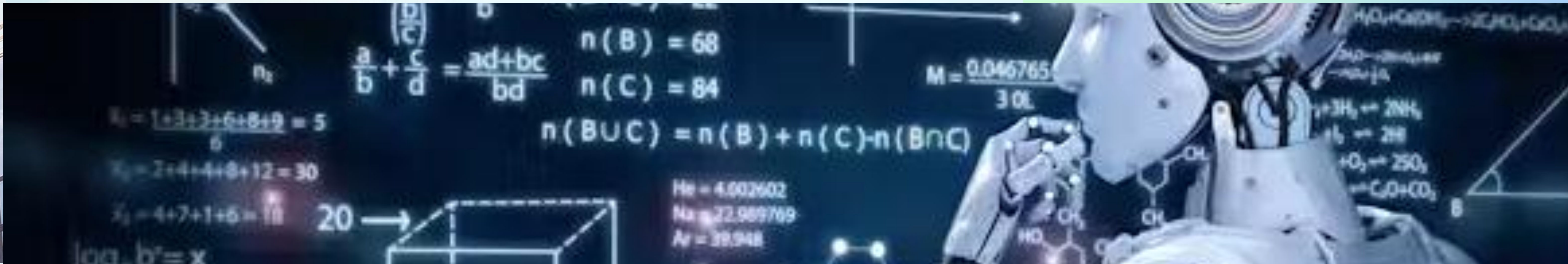
PREDICTION OF HOUSE PRICES WITH EXTREME GRADIENT BOOSTING.

PREDICTION OF HOUSE PRICES WITH EXTREME GRADIENT BOOSTING.

NAME: DINESH A/L MANIVANNAN (211021094)
SUPERVISED BY ASSOC. PROF. DR. ZAHEREEL ISHWAR ABDUL KHALIB

FYP: FYP2


$\frac{a}{b}+\frac{c}{d}=\frac{ad+bc}{bd}$
n(B) = 68
n(C) = 84
n(B∪C) = n(B)+n(C)-n(B∩C)
20 →

Limitations

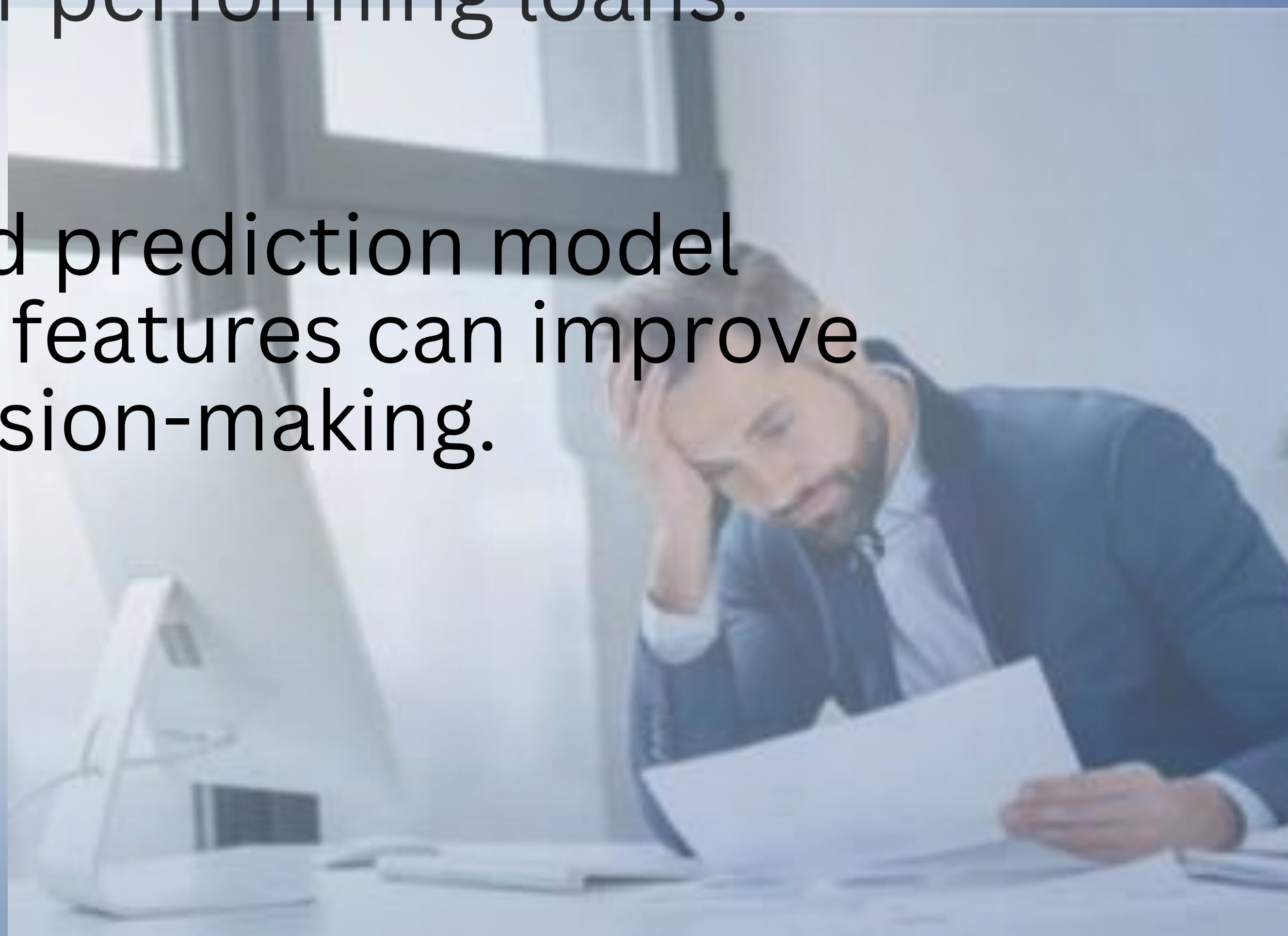
Problem Statement

Objectives

- Overvaluing and undervaluing can cause financial difficulties for buyers and market imbalances for developers.
- Important factors are often overlooked, leading to inaccurate predictions.
- Overpricing forces low- and middle-income families to take unsustainable loans, increasing household debt and non-performing loans.

SOLUTION??

A near-accurate ML-based prediction model incorporating overlooked features can improve pricing accuracy and decision-making.



Problem Statement

Objectives

- To design a tree-based machine learning model for predicting house prices using real world dataset.
- To develop a webpage that allows user to input feature details and get house price prediction.
- To evaluate the performance of model using standard metrics which and MAE, MSE, RMSE and R squared.

Scope

Objectives

Scope

- The project aims for high accuracy and clarity by using algorithm such as Gradient Boosting.
- Assess performance using metrics such as R-squared, Mean Squared Error (MSE), and Mean Absolute Error (MAE).
- Attempts to understand the significance of characteristics such as neighborhood quality and access to conveniences that determine property values.

Limitations

Scope

Limitations

- Model accuracy depends on data quality. Incomplete or outdated data may impact performance.
- Availability of relevant datasets with comprehensive features can be a constraint.
- Tree-based models may overfit and struggle with generalization.

Problem Statemen

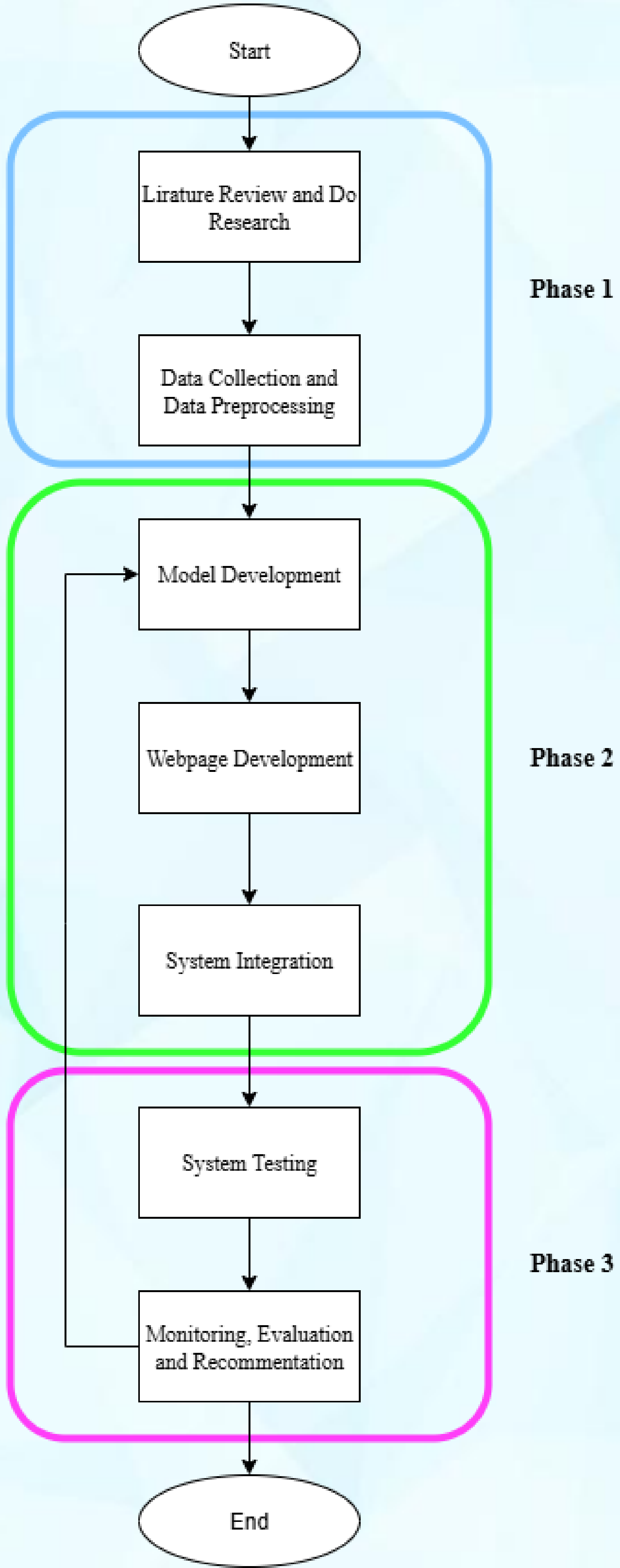
SOTA - PART A

Previous paper	Model	Attributes/Factors	Limitations / Gaps
Kalidass, J., et(2024). HOUSE PRICE PREDICTION USING MACHINE LEARNING.	Random Forest, Gradient Boosting	Property size, neighbourhood quality, proximity to schools and work centres	Dataset preprocessing and feature engineering not emphasized.
Li, Y. (2023). Analysis of Real Estate Predictions Based on Different Models.	Decision Tree, Extreme Gradient Boosting, Random Forest	Square footage, proximity to amenities, market trends, year of construction	The absence of temporal data integration leads to inaccurate forecasting when long-term trends shift.
Mao, M. (2024). A Comparative Study of Random Forest Regression for Predicting House Prices Using.	Random Forest Regression	Property type, neighbourhood characteristics, transport access, historical prices	The model uses static factors and lacks integration of dynamic factors like policy changes or new infrastructure projects.
Quang, T., Minh, N., Hy, D., & Bo, M. (2020). Housing Price Prediction via Improved Machine Learning Techniques.	Random Forest, Extreme Gradient Boosting	Proximity to business hubs, crime rates, public infrastructure, social factors	Improved techniques but lack cross-region adaptability; no detailed explainability analysis.
YAVUZ ÖZALP, A., & AKINCI, H. (2023). Comparison of tree-based machine learning algorithms in price prediction of residential real estate.	Decision Tree, Random Forest, Extra Trees, Gradient Boosting	Proximity to public transport, land area, crime rates, infrastructure	Overlapping features (e.g., public transport and infrastructure) may introduce multicollinearity, affecting prediction reliability.

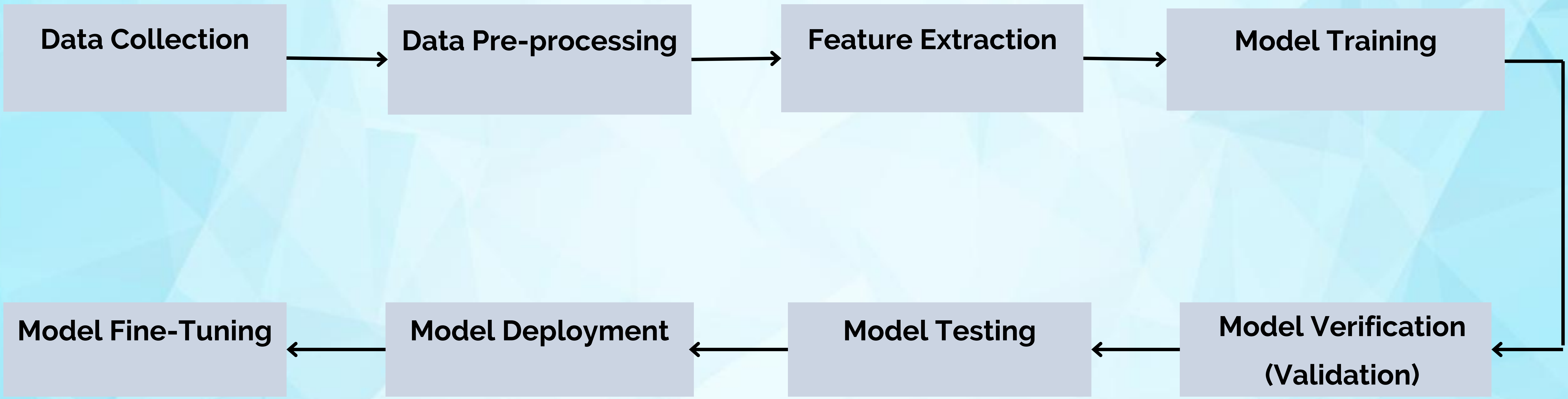
SOTA - PART B

Previous paper	Model	Attributes/Factors	Limitations / Gaps
Kumar, Bv., & Professor, A. (2020). House Price Prediction using Garadient Boost Regression Model	Gradient Boosting	Neighbourhood, lot size, historical pricing data, economic indicators	Performance limited to Gradient Boost regression; no comparative study with other tree based models.
Adetunji, A. B., Akande, O. N., Ajala, F. A., Oyewo, O., Akande, Y. F., & Oluwadara, G. (2022). House Price Prediction using Random Forest Machine Learning Technique	Random Forest Regression	Distance to city centre, property type, land size, social factors	Limited cross-validation techniques; no use of advanced ensemble methods for comparison.
Akash Dagar and Shreya Kapoor. (2020). A Comparative Study on House Price Prediction.	Multivariable Linear Regression, Decision Tree Regression, Random Forest Regression	Area, location, age of property, number of rooms	Lack of scalability for larger datasets; limited focus on hyperparameter optimization for tree-based models.
Chuhan, N. (2024). House price prediction based on different models of machine learning.	Linear Regression, Support Vector Machine (SVM), Random Forest regression, Extreme Gradient Boosting	Size, number of bedrooms and bathrooms, proximity to public transport	No detailed discussion on feature importance or interpretability of results.
Mohd, T., Masrom, S., & Johari, N. (2019). Machine learning housing price prediction in petaling jaya, Selangor, Malaysia.	Linear Regression, Decision Tree, Random Forest, Ridge and Lasso algorithms	Property type, land area, age, location	Study focused only on a specific geographical area (Petaling Jaya, Selangor), limiting wider applicability.

FLOWCHART


Start
Lirature Review and Do
Research
Data Collection and
Data Preprocessing
Phase 1
Model Development
Webpage Development
System Integration
Phase 2
System Testing
Monitoring, Evaluation
and Recommentation
Phase 3
End

MACHINE LEARNING PIPELINE



Data Collection

Data Pre-processing

Feature Extraction

Model Training

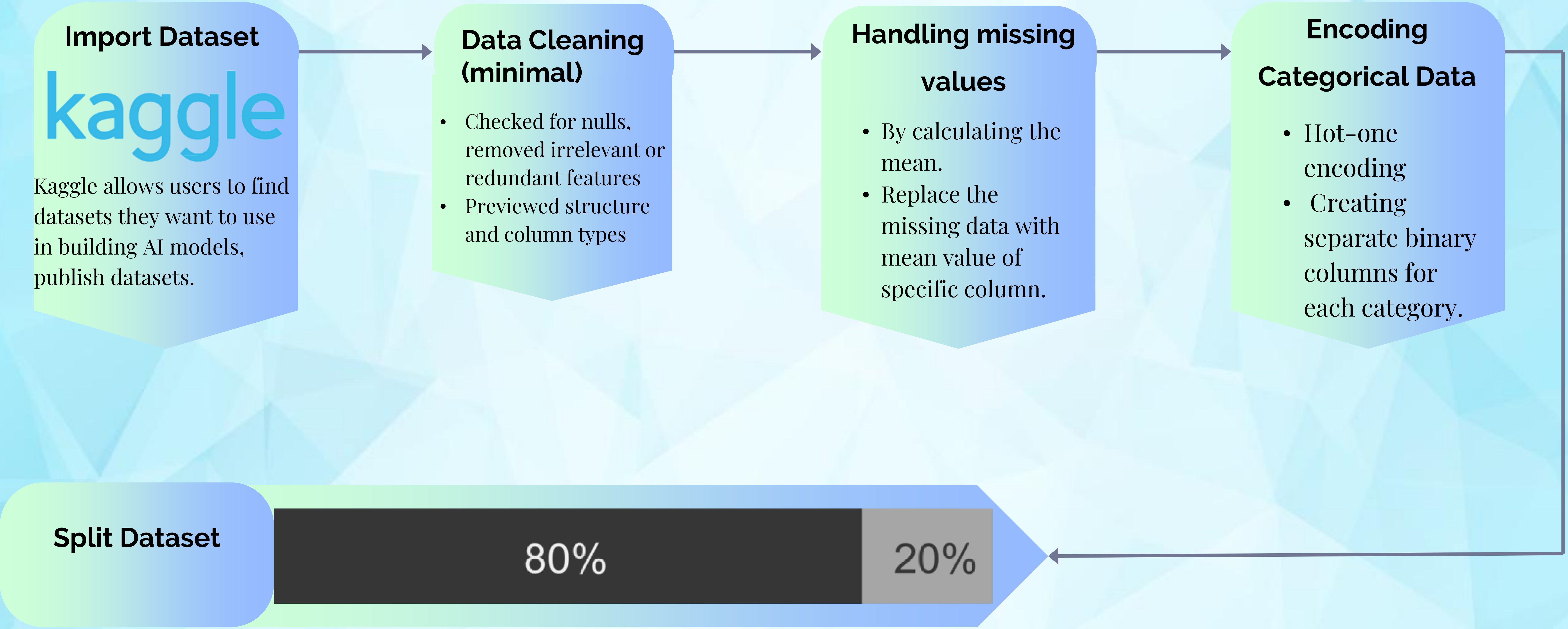
Model Verification (Validation)

Model Testing

Model Deployment

Model Fine-Tuning

DATA PRE-PROCESSING



Import Dataset

kaggle

Kaggle allows users to find datasets they want to use in building AI models, publish datasets.

Data Cleaning (minimal)

- Checked for nulls, removed irrelevant or redundant features
- Previewed structure and column types

Handling missing values

- By calculating the mean.
- Replace the missing data with mean value of specific column.

Encoding Categorical Data

- Hot-one encoding
- Creating separate binary columns for each category.

Split Dataset

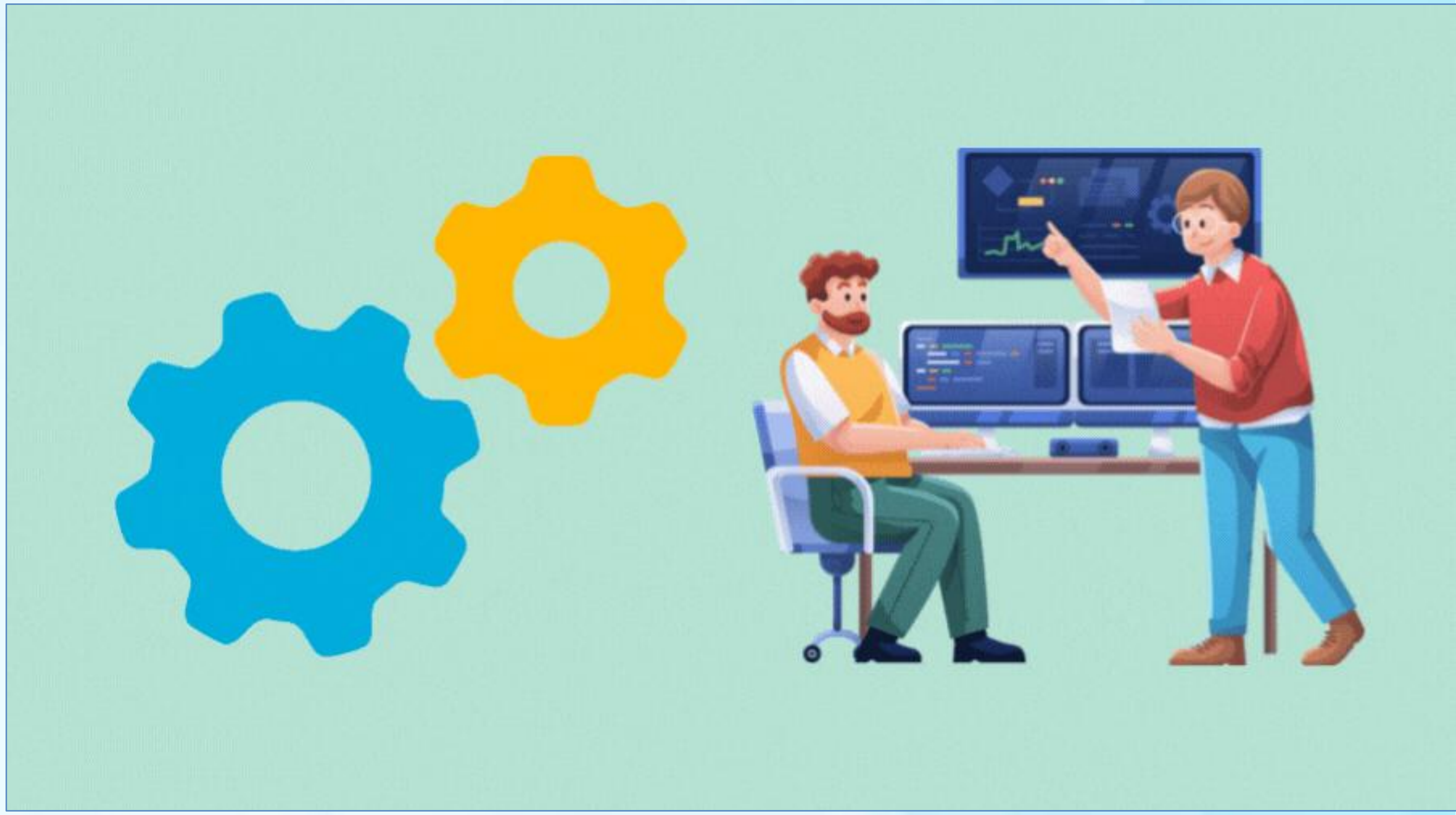
FEATURE EXTRACTION & ENGINEERING

Constructed new features

- TotalSF (Total square footage) = TotalBsmtSF + 1stFlrSF + 2ndFlrSF
- TotalBath = FullBath + 0.5 × HalfBath + BsmtFullBath + 0.5 × BsmtHalfBath

Combined related features

- YearRemodAdd and YearBuilt were used to derive property age
- GarageArea and GarageCars considered jointly to reflect garage utility



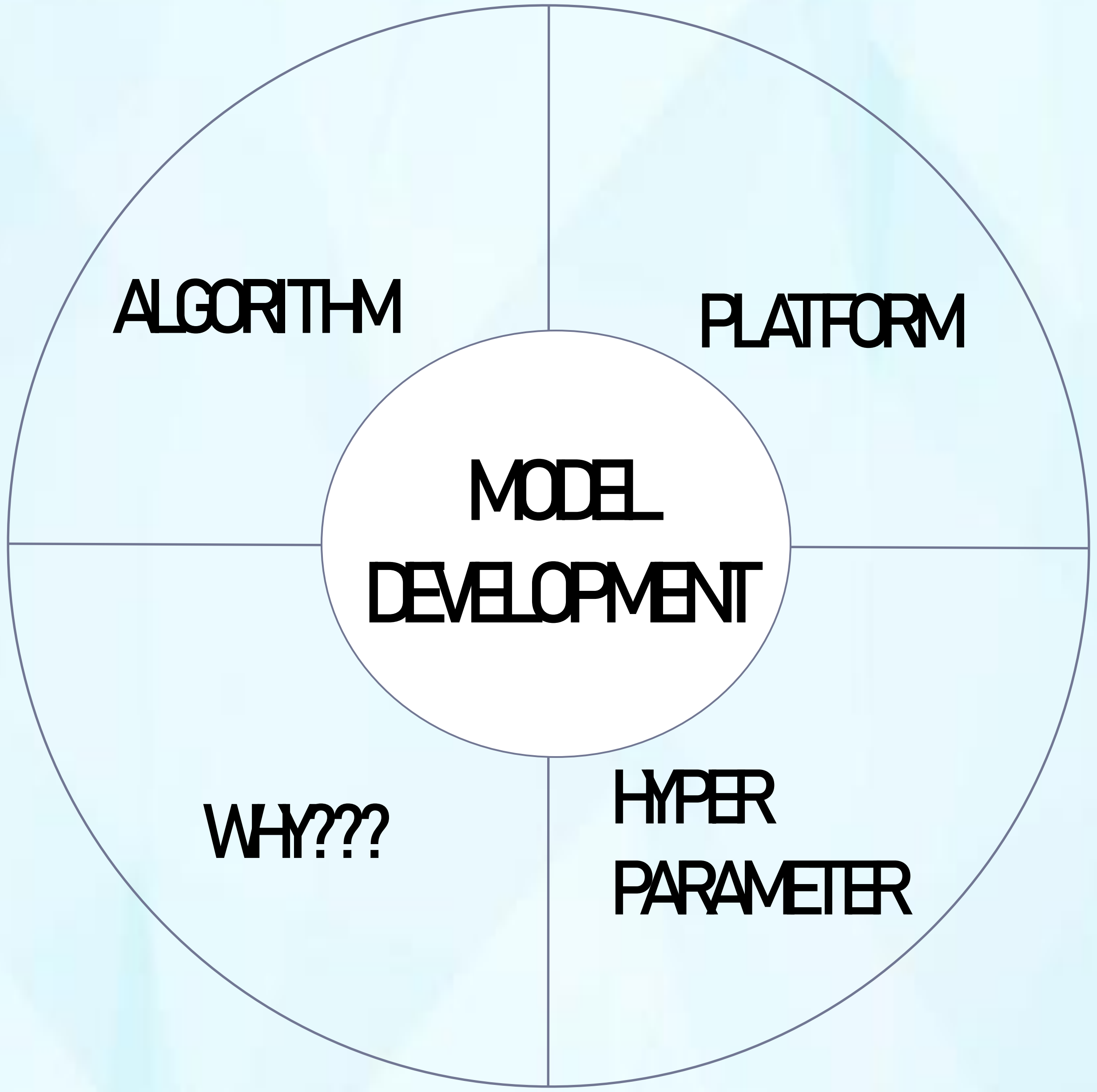
Removed irrelevant or redundant columns

- Dropped ID columns and features with near-zero variance or too many missing values

Converted categorical features

- Transformed into numerical form using one-hot encoding before selection

MODEL DEVELOPMENT



MODEL DEVELOPMENT

ALGORITHM

- Gradient boosting decision tree that build models sequentially to correct predecessor errors.
- Offers high prediction accuracy, resistance to overfitting, and efficient training.

PLATFORM

- Google Colab (Python)
- Libraries used: xgboost, scikit-learn, pandas, matplotlib, seaborn

WHY???

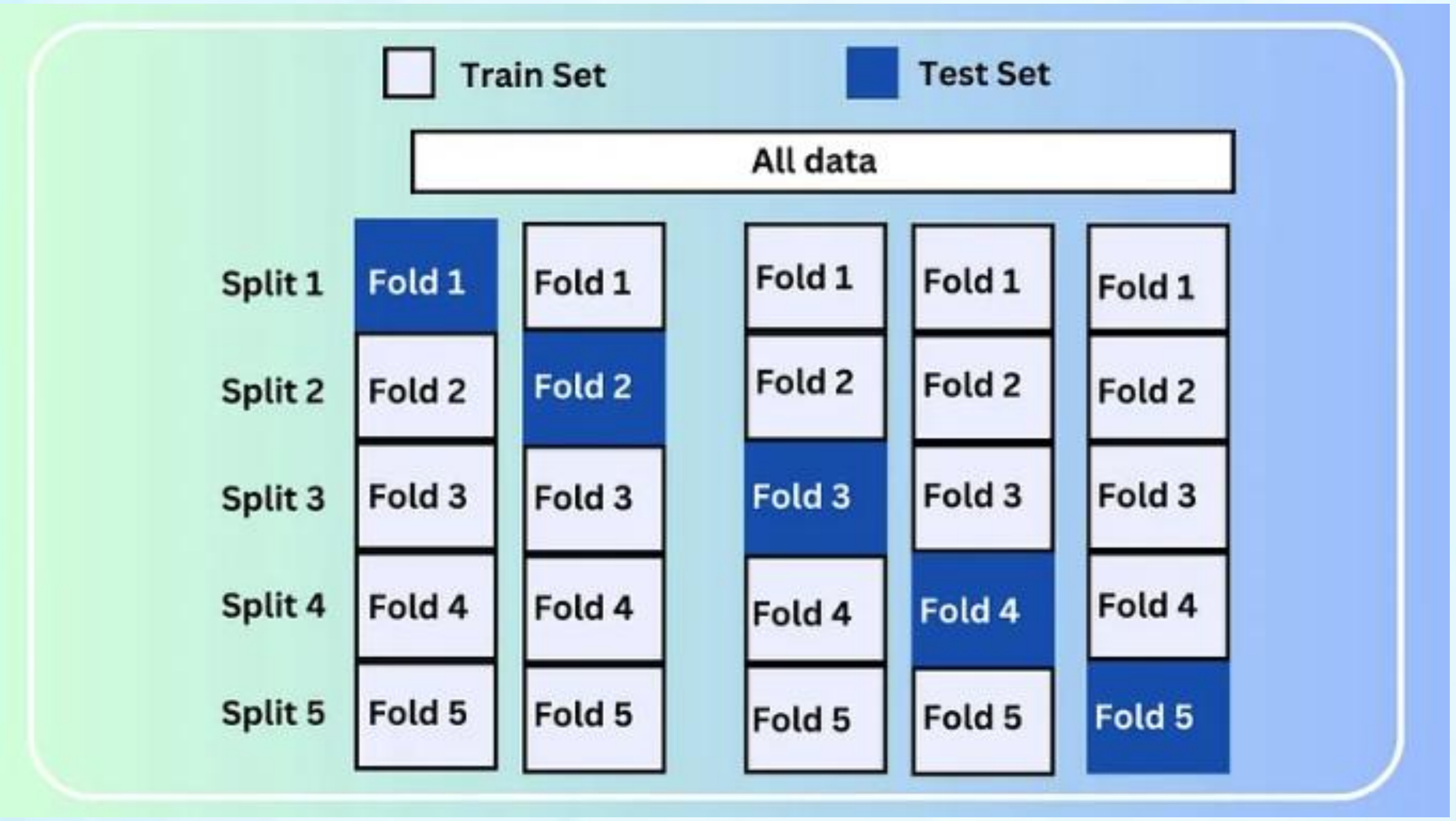
- Handles missing data automatically.
- Supports regularization (L1 & L2) to avoid overfitting.
- Hghly optimized for speed and memory.
- Robust to outliers and irrelevant features.

HYPER PARAMETER

- n_estimators
- learning_rate
- max_depth
- subsample
- random_state

MODEL VERIFICATION

K-Fold Cross Validation



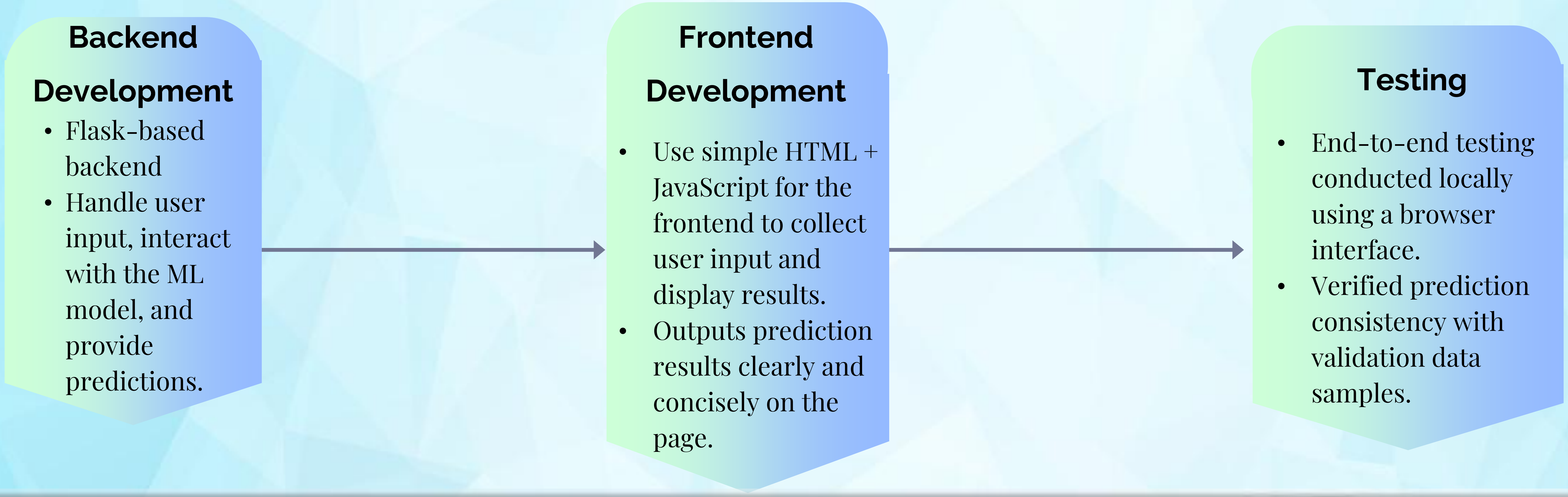
□ Train Set ■ Test Set

All data

Split 1	**Fold 1**	Fold 1	Fold 1	Fold 1	Fold 1
Split 2	Fold 2	**Fold 2**	Fold 2	Fold 2	Fold 2
Split 3	Fold 3	Fold 3	**Fold 3**	Fold 3	Fold 3
Split 4	Fold 4	Fold 4	Fold 4	**Fold 4**	Fold 4
Split 5	Fold 5	Fold 5	Fold 5	Fold 5	**Fold 5**

- Split training data into *k* folds .
- Train model on k−1 folds and validate on the remaining one.
- Ensures the model generalizes across different subsets of data.
- Reduces the risk of overfitting and improves reliability of evaluation metrics

MODEL DEPLOYMENT



Backend Development

- Flask-based backend
- Handle user input, interact with the ML model, and provide predictions.

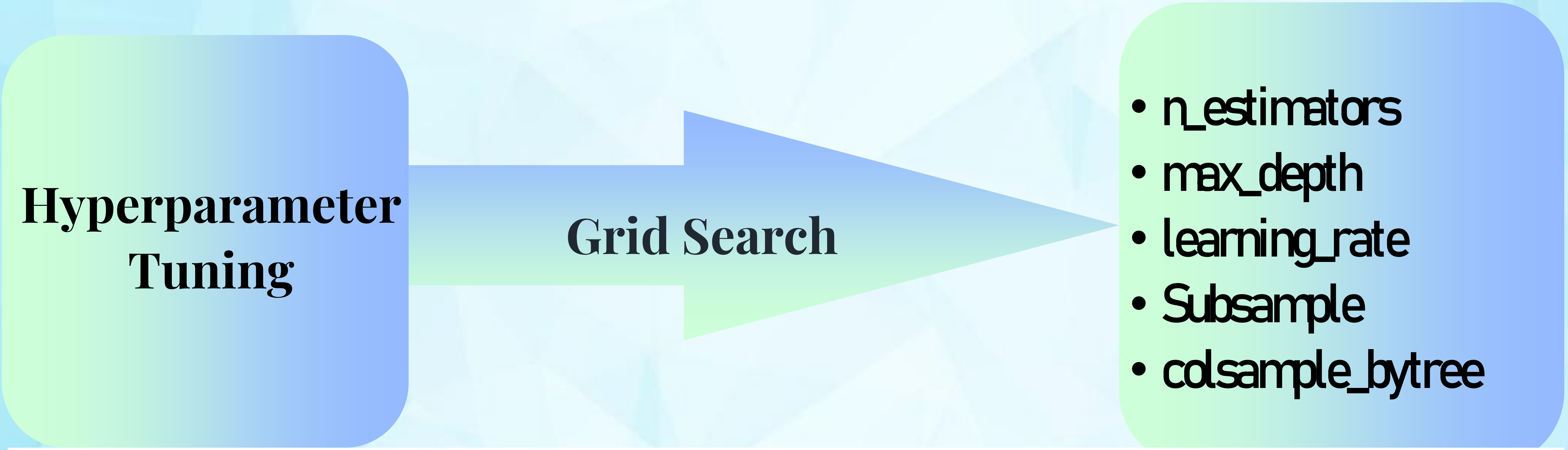
Frontend Development

- Use simple HTML + JavaScript for the frontend to collect user input and display results.
- Outputs prediction results clearly and concisely on the page.

Testing

- End-to-end testing conducted locally using a browser interface.
- Verified prediction consistency with validation data samples.

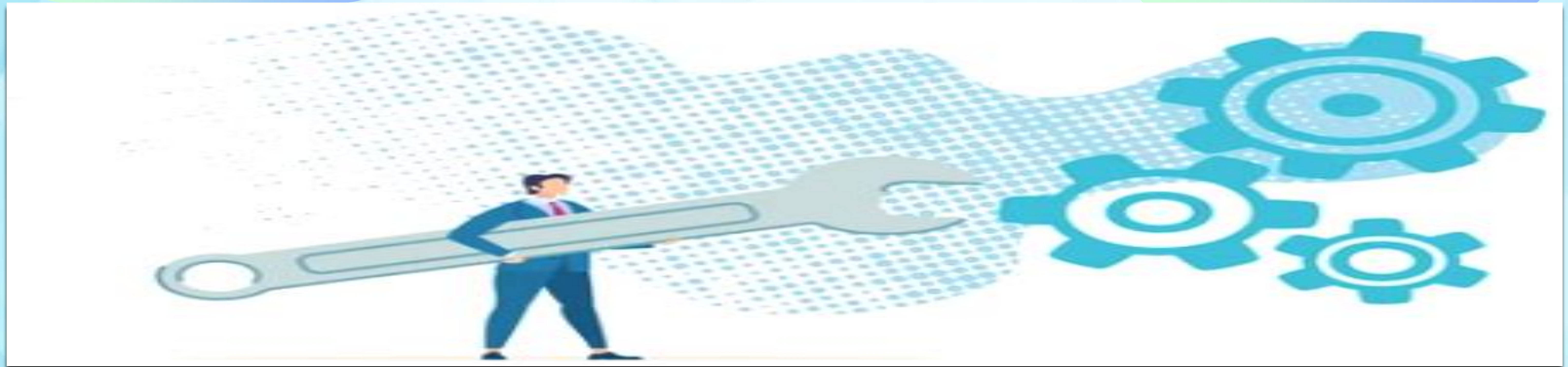
MODEL FUNE-TUNING



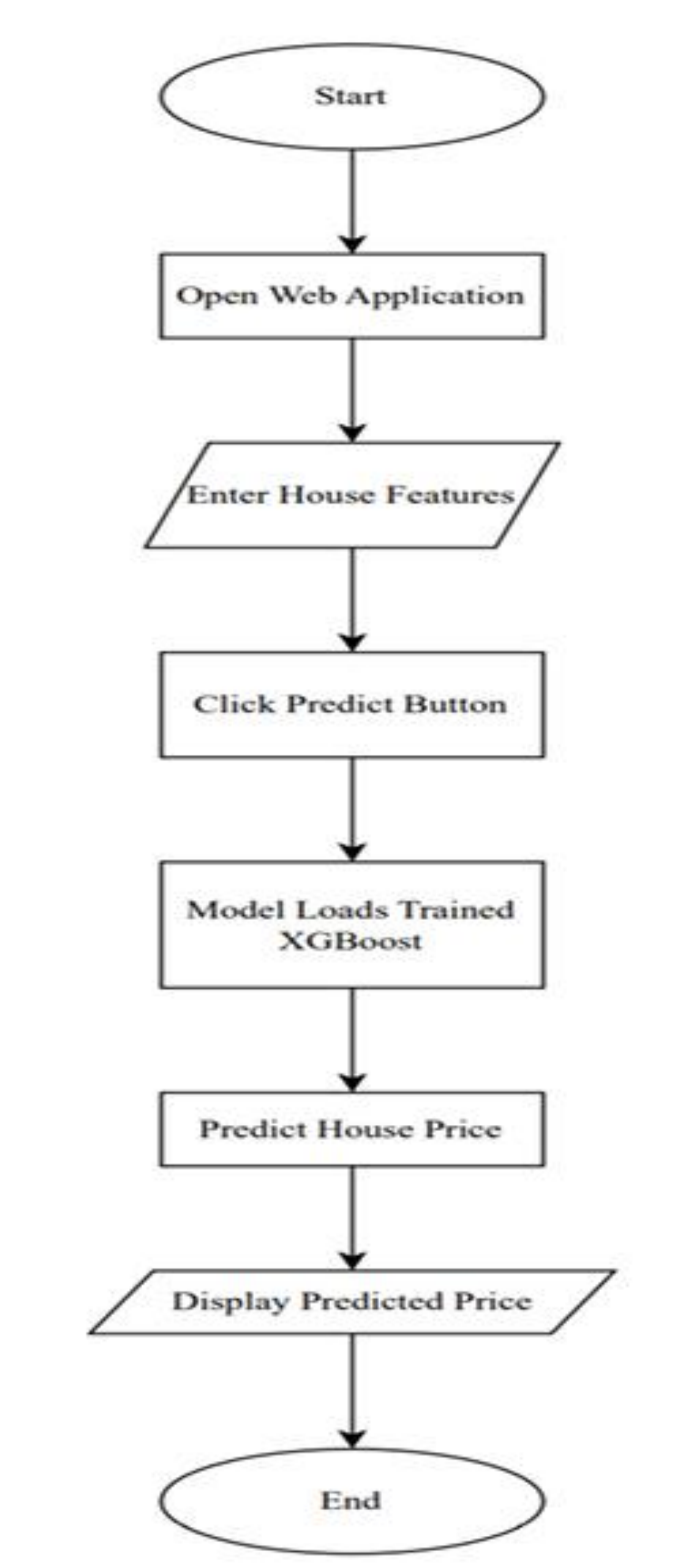
Hyperparameter Tuning

Grid Search

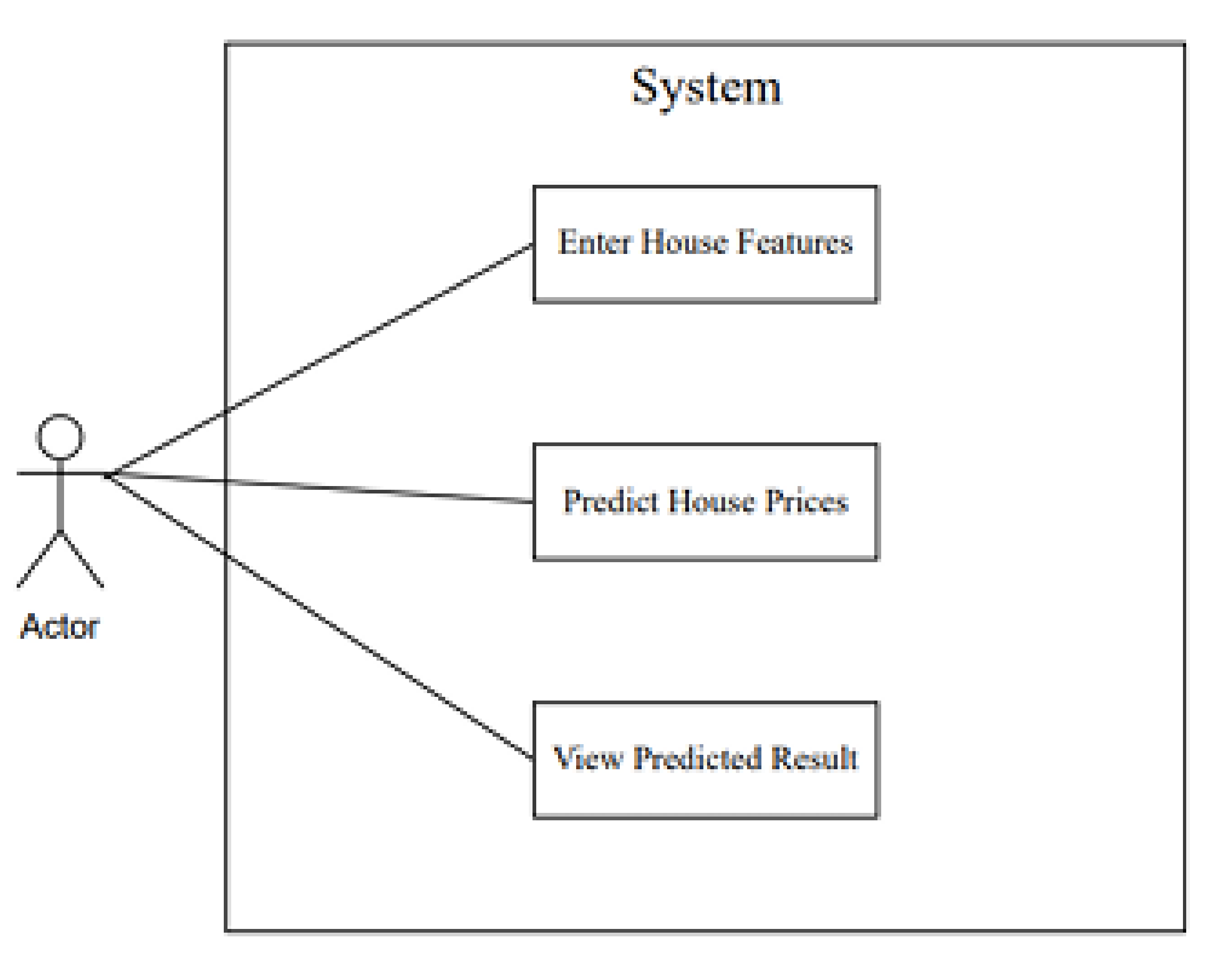
- n_estimators
- max_depth
- learning_rate
- Subsample
- colsample_bytree



FLOWCHART OF WEB-PAGE SYSTEM



USE CASE DIAGRAM



MODEL EVALUATION & PERFORMANCE

Evaluation Metrics Used: MAE, MSE, RMSE, R^2 Score

Dataset Split: 80% Training / 20% Testing

Test Set Results:

- MAE: 16945.41
- MSE: 6.77×10^8
- RMSE: 26030.62
- R^2 Score: 0.9117

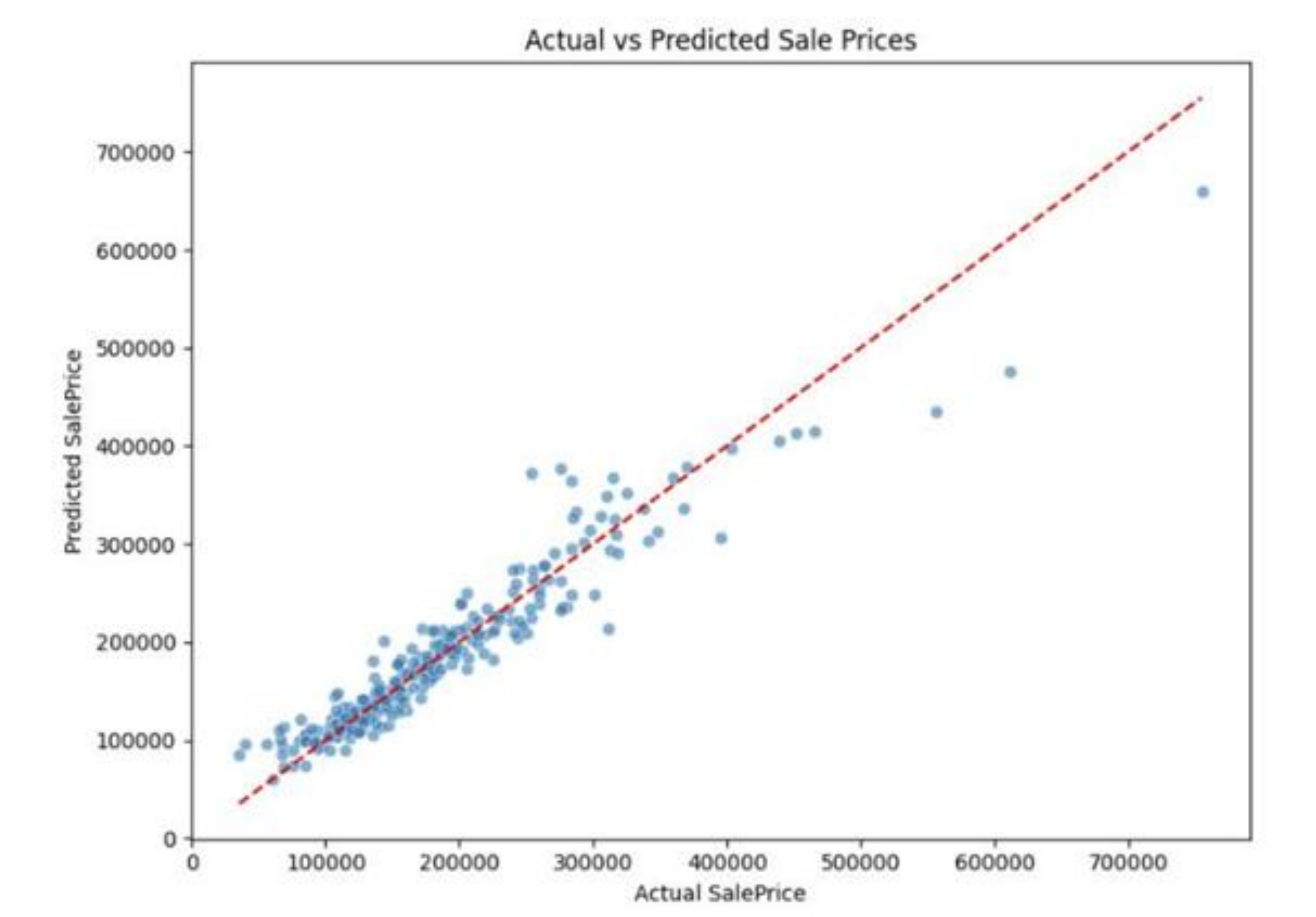
Hyperparameter Values:

- n_estimators = [100,200]
- learning_rate = [3,5,7]
- max_depth = [0.05,0.1]
- Subsample = [0.8,1]
- random_state = [0.8,1]

- **Indicates strong predictive power and good generalization to unseen data.**

RESULT VISUALIZATION

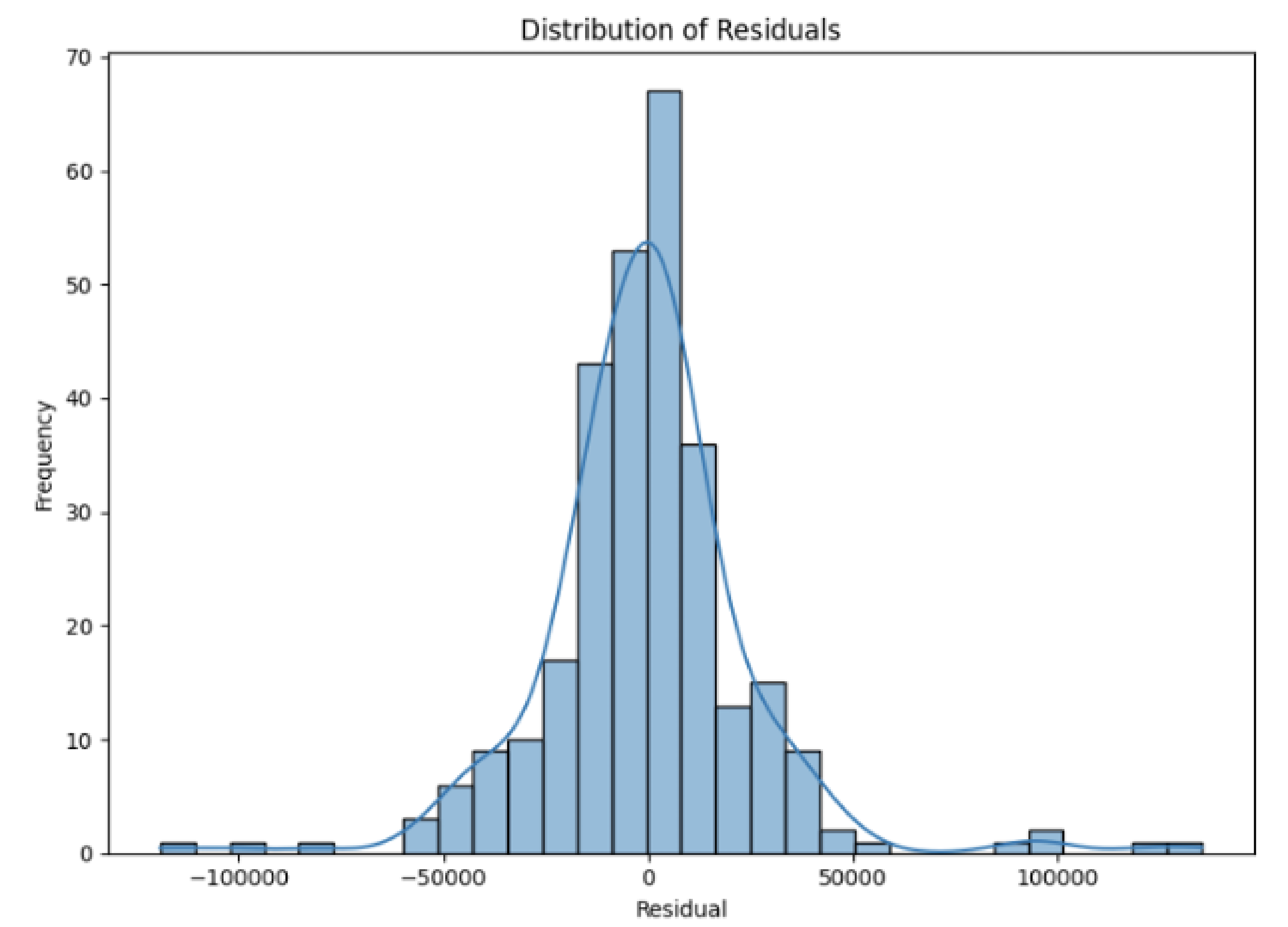
Actual vs. Predicted Prices



- A red diagonal reference line indicates perfect prediction accuracy.
- Majority of data points cluster tightly along the diagonal, reflecting strong predictive performance.
- Despite a few outliers, the model effectively captures the underlying patterns in the housing market.

RESULT VISUALIZATION

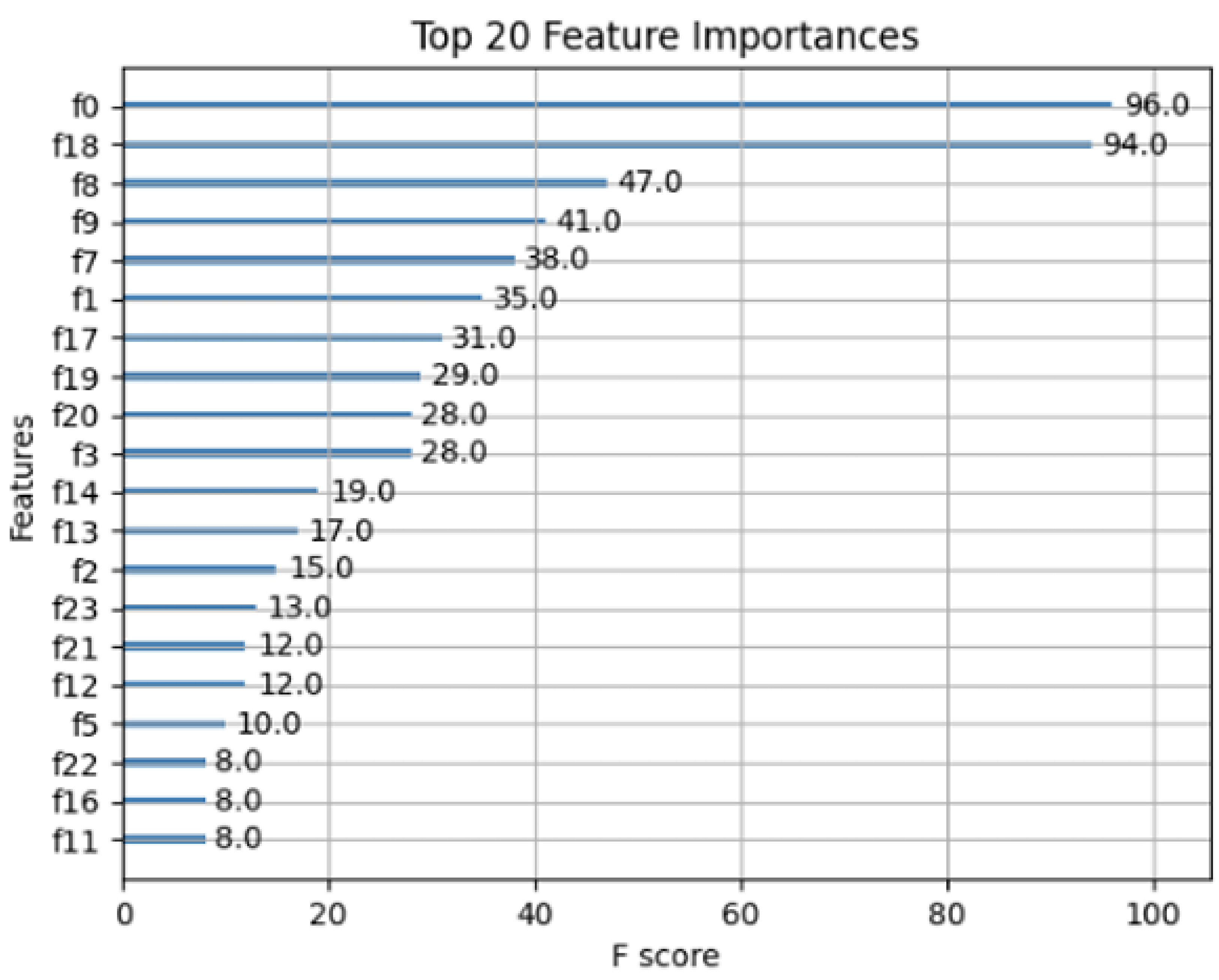
Distribution of Residuals



- This indicates that the model errors are random and unbiased, which is ideal for regression.
- The majority of residuals lie within a narrow range, suggesting high prediction accuracy.
- The distribution's symmetry supports the conclusion that the model neither consistently overestimates nor underestimates prices.

RESULT VISUALIZATION

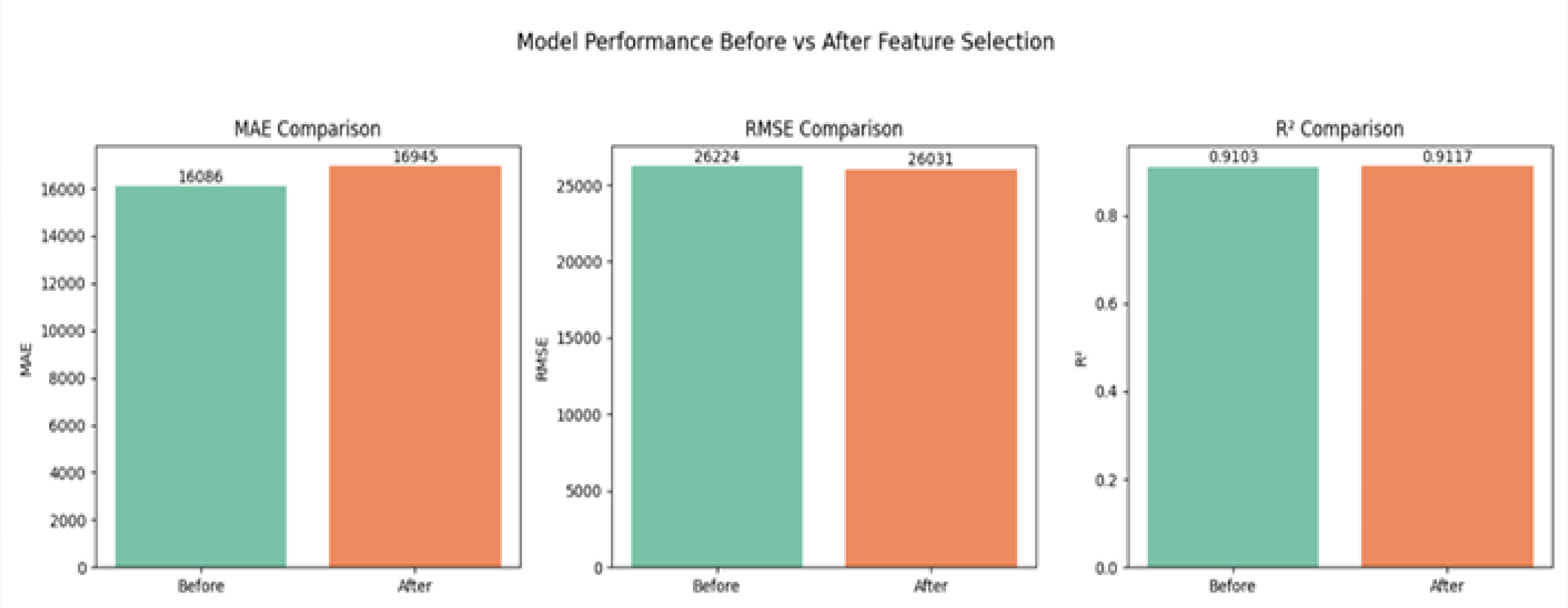
Feature Importance Analysis



- Feature importance confirms that feature engineering was effective.
- The model demonstrates interpretable behaviour without needing domain knowledge.
- Less important features may still contribute to niche cases, and could support future model simplification.

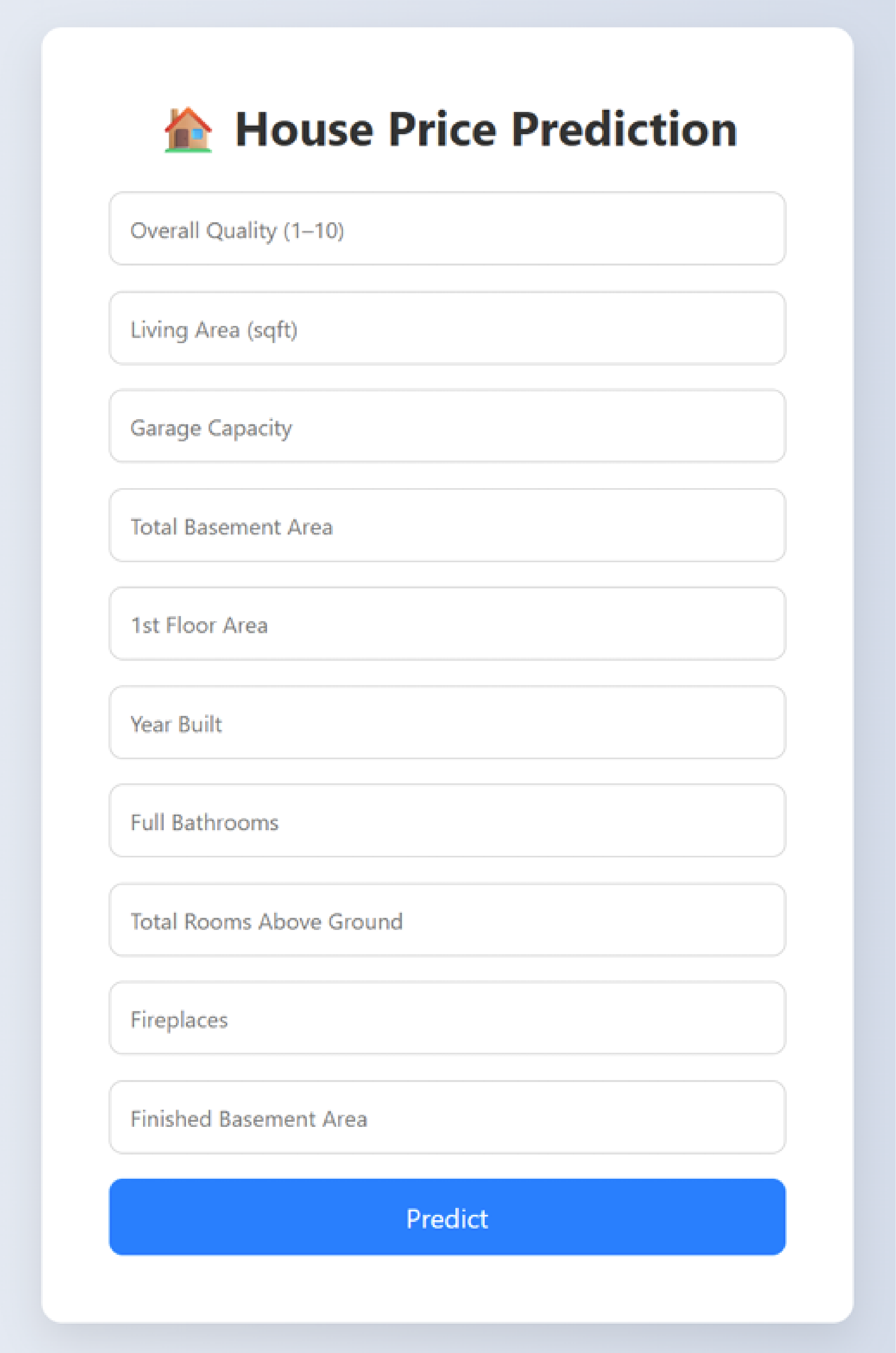
RESULT VISUALIZATION

Performance Comparison Across All Stages



- Each step contributes incremental improvements.
- Final model is more stable, accurate, and generalizable.
- Feature selection helped simplify the model without degrading performance.

WEB-PAGE INTERFACE



🏠 House Price Prediction

Overall Quality (1–10)

Living Area (sqft)

Garage Capacity

Total Basement Area

1st Floor Area

Year Built

Full Bathrooms

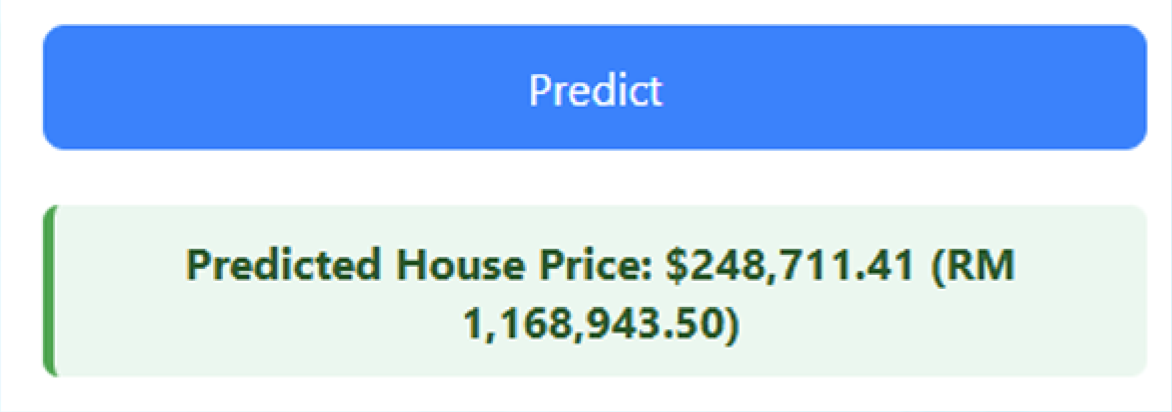
Total Rooms Above Ground

Fireplaces

Finished Basement Area

Predict

SYSTEM OUTPUT



Predict

Predicted House Price: $248,711.41 (RM 1,168,943.50)

Q & A