House Price Predictor

MINI PROJECT REPORT

Of Internship

BACHELOR OF TECHNOLOGY

COMPUTER SCIENCE AND ENGINEERING

Dr. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY LUCKNOW

SUBMITTED BY

1.GOPESH KUMAR SHUKLA:University Roll No. 2200100100148

2.AMBIKESH TRIPATHI:University Roll No. 2200100100054

3.AKSHAT JAIN:University Roll No. 2200100100047

UNDER THE SUPERVISION OF

Mrs.Pallavi Shukla

Submitted To

UNITED COLLEGE OF ENGINEERING AND

RESEARCH, PRAYAGRAJ

UNITED
Group of Institutions
Greater Noida • Allahabad

Proposed Topic:

House Price Predictor

INTRODUCTION:

The real estate market significantly influences a nation's economy, with house prices affected by factors like location, property size, and amenities. Accurate price prediction is crucial for buyers, sellers, and real estate professionals, yet traditional valuation methods often fall short. This project aims to develop an AI-based house price prediction model using machine learning techniques. By analyzing key property features, the model will deliver precise price estimations, enabling data-driven decision-making and enhancing market transparency and efficiency.

Project Overview:

The project aims to develop an AI-driven model that accurately predicts house prices based on various property features, addressing the limitations of traditional valuation methods and enhancing decision-making for buyers, sellers, and real estate professionals. By leveraging machine learning, the model will analyze key attributes, such as location and size, and provide a user-friendly tool for price estimation, ultimately increasing market transparency and efficiency. The methodology includes data collection and preprocessing, feature selection, model development using various algorithms, performance evaluation, and user testing to refine the final product. Expected outcomes include a reliable prediction model that empowers stakeholders with data-driven insights for better investment strategies.

Technology Used:

The project employs machine learning techniques, specifically Lasso Regression, to predict house prices. This approach enhances predictive accuracy and performs automatic feature selection, allowing the model to identify and utilize the most relevant property attributes for pricing predictions.

Conclusion:

The House Price Predictor project successfully integrates machine learning techniques, specifically Lasso Regression, to provide accurate and reliable price estimations based on various property features. By leveraging data-driven insights, the model empowers buyers, sellers, and investors to make informed decisions in the real estate market. This tool not only enhances the efficiency of property valuation but also promotes transparency, paving the way for better investment strategies and market understanding. Future enhancements could include expanding the dataset and incorporating additional machine learning algorithms to further refine predictions.

Feasibility Study:

Feasibility Study for "HOUSE PRICE PREDICTOR" Machine learning Program:

1. Technical Feasibility

- Utilizes widely available technologies (Python, machine learning libraries).
- Development can be executed on standard hardware with adequate specifications.
- Availability of datasets for training and validation.

2. Economic Feasibility

- Cost-effective solution compared to traditional valuation methods.
- Potential for high return on investment through improved decision-making in real estate.
- Low operational costs after initial development.

3. Operational Feasibility

- User-friendly interface designed for non-technical users.
- Suitable for various stakeholders in the real estate market.
- Easy integration into existing real estate platforms.

4. Legal Feasibility

- Compliance with data privacy regulations (e.g., GDPR).
- Ownership rights for used datasets are confirmed

5. Schedule Feasibility

- Project timeline is manageable within typical academic or development cycles.
- Key milestones can be established to ensure timely progress.

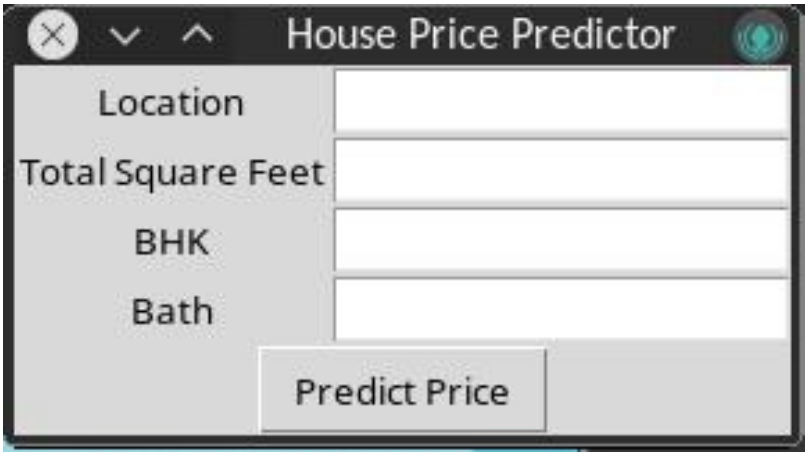
6. Market Feasibility

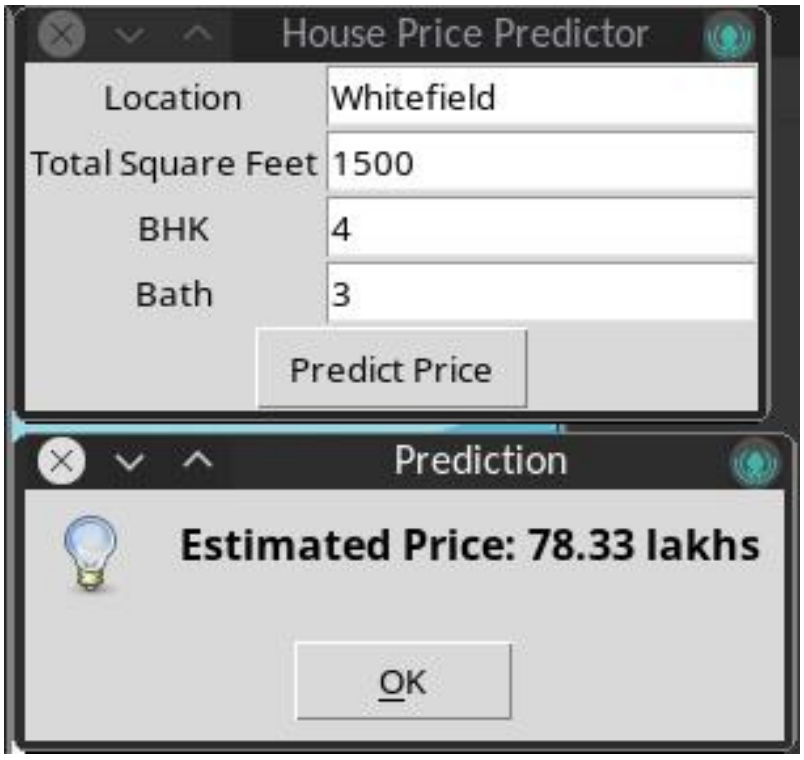
- High demand for accurate house pricing tools in a fluctuating market.
- Positive reception from real estate professionals and potential users.

7. Risk Feasibility

- Identified risks can be mitigated through thorough testing and validation.
- Continuous updates and improvements can enhance model accuracy and reliability.

Snapshot of result





Methodology/ Planning of work :

Phase 1: Data Collection and Preprocessing

1. Collect Bengaluru house prices dataset

2. Clean and preprocess data (handle missing values, outliers)

3.Split data into training and testing sets

Phase 2: Predictive Modeling

1. Select suitable machine learning algorithms (Linear Regression, Lasso, Ridge)

2. Train models using training data

3. Evaluate model performance (R2 score, mean squared error)

Phase 3: GUI Development

1. Design user-friendly GUI using Tkinter

2. Implement data input functionality

3. Integrate predictive model with GUI

4. Test and debug GUI

Phase 4: Testing and Validation

1. Test predictive model accuracy

2. Validate GUI functionality

3. Conduct user acceptance testing (UAT)

4. Iterate on feedback and improvements

Phase 5: Deployment and Maintenance

1. Deploy system (locally or on cloud)

2. Monitor system performance

3. Update models with new data

4. Maintain and update GUI as needed

Facilities required for proposed work:

Hardware Requirements

1. Computer System:

– A personal computer or server with adequate processing power (preferably multi-core CPU) to handle data processing and model training.

2. Graphics Processing Unit (GPU):

– A dedicated GPU (e.g., NVIDIA) for accelerated training of deep learning models, if applicable.

3. Memory (RAM):

– Minimum 16 GB RAM for efficient data handling; 32 GB or more is recommended for larger datasets.

4. Storage:

– Sufficient SSD storage (minimum 256 GB) to accommodate datasets and models, with higher capacity recommended for larger projects.

Software Requirements

1. Programming Language:

– Python: The primary language used for implementing machine learning algorithms due to its extensive libraries and frameworks.

2. Integrated Development Environment (IDE):

– Jupyter Notebook, PyCharm, or Anaconda: IDEs for writing, testing, and visualizing code and data.

3. Machine Learning Libraries:

– Scikit-learn: For traditional machine learning algorithms.

– TensorFlow or PyTorch: For deep learning model development.

4. Data Manipulation Libraries:

– Pandas: For data manipulation and analysis.

– NumPy: For numerical operations and array handling.

5. Visualization Tools:

– Matplotlib or Seaborn:For creating data visualizations to interpret results.

Bibliography :

Under Supervision of our Instructor Mrs,Pallavi Shukla mam

https://www.geeksforgeeks.org/

https://youtu.be/DVxkI1VmpCk?si=ZYg1DgwPwC4pYgO-