Industry Project Report

On

AI/ML Hotel Price Predictor

Developed By: -

Munish Patwa (21162121017)

Guided By:-

Prof. Sulabh Bhatt (Internal)

Rahul Bhagchandani(External)

Submitted to

Faculty of Engineering and Technology

Institute of Computer Technology

Ganpat University

Year - 2025

Ganpat University
॥ विद्यया समाजोत्कर्षः ॥

Institute of Computer Technology

CERTIFICATE

This is to certify that the **Industry** Project work entitled **"AI/ML Hotel Price Predictor"** by Munish Patwa (Enrolment No. 21162121017) of Ganpat University, towards the partial fulfillment of requirements of the degree of Bachelor of Technology – Computer Science and Engineering, carried out by them in the AI Department at Ascion Global Partners. The results/findings contained in this Project have not been submitted in part or full to any other University / Institute for the award of any other Degree/Diploma.

Name & Signature of Internal Guide

Name & Signature of Head

Place: ICT - GUNI

Date: 22 February 202

ACKNOWLEDGEMENT

Industry Internship project is a golden opportunity for learning and self-development. I consider myself very lucky and honored to have so many wonderful people lead me through in completion of this project. First and foremost, I would like to thank Dr. Rohit Patel, Principal, ICT, and Prof. Dharmesh Darji , Head, ICT who gave us an opportunity to undertake this project. My grateful thanks to Prof. Tejas Kadiya, Prof. Sulabh Bhatt, and Mr. Rahul Bhagchandani (Internal & External Guides) for their guidance in project work AI/ML Hotel Price Predictor, who despite being extraordinarily busy with academics, took time out to hear, guide and keep me on the correct path.
I do not know where I would have been without his/her help.

MUNISH PATWA (Enrollment No:21162121017)

ABSTRACT

In the hospitality industry, pricing strategies play a crucial role in maximizing revenue and ensuring competitive advantage. This project focuses on developing an AI/ML-based hotel price prediction model to forecast hotel prices based on various factors such as location, seasonality, hotel type, and amenities. The process begins with the manual collection of hotel data, which is stored in a MySQL database. Following data acquisition, key features are extracted and selected to ensure optimal model performance. Two machine learning approaches, namely Random Forest Regressor and a Deep Neural Network (DNN), are employed to predict the prices with high accuracy. The Random Forest model, leveraging an ensemble of decision trees, provides interpretability and robustness, while the DNN model aims to capture complex patterns within the data for improved accuracy. The model's performance is evaluated using standard regression metrics, and the results indicate promising potential for practical applications in dynamic pricing, revenue management, and competitive analysis within the hospitality industry. This project showcases how advanced AI/ML techniques can be leveraged to automate price forecasting, ultimately optimizing pricing strategies and enhancing business decision-making in the hotel industry.

INDEX

CHAPTER: 1 INTRODUCTION

CHAPTER 1 INTRODUCTION

In today's data-driven era, the ability to accurately predict pricing strategies is crucial in industries such as hospitality. With the growing competition in the hotel sector, dynamic pricing has become essential to maximize revenue and ensure competitiveness. However, predicting hotel prices involves processing large volumes of data, including factors such as location, seasonality, customer demand, amenities, and hotel type. For hotel managers and pricing strategists, manually analyzing these factors can be time-consuming and error-prone.

This project aims to develop an AI/ML-based hotel price prediction system that can automate this process and provide accurate pricing forecasts. By leveraging machine learning models such as Random Forest Regressor and Deep Neural Networks (DNN), the model processes various hotel attributes and predicts optimal pricing based on historical data. The system is designed to allow stakeholders to input hotel characteristics and receive dynamic, data-driven price recommendations, facilitating smarter decision-making.

The project combines traditional data processing techniques with advanced machine learning algorithms, providing an efficient and scalable solution for predicting hotel prices. The goal is to provide a tool that not only simplifies the pricing process but also helps businesses optimize their revenue strategies by analyzing complex patterns in hotel pricing. Through the use of feature extraction and selection techniques, the model aims to ensure high accuracy and generalization across diverse hotel data sets.

The following chapters will delve into the objectives, background, methodology, and implementation details of this project, along with its expected outcomes and future scope.

Below is the list of the tools and technologies used in this project:

- **MySQL: Database management system used for storing hotel data and facilitating data manipulation.**
- **Python: Primary programming language for data processing, model development, and evaluation.**
- **Random Forest Regressor: Machine learning model used for predicting hotel prices based on input features.**
- **Deep Neural Networks (DNN): Advanced model used for capturing complex patterns and improving price predictions.**
- **Pandas: Python library used for data manipulation and analysis during the feature extraction and preprocessing phases.**
- **Scikit-learn: Machine learning library used for implementing and training the Random Forest model.**
- **TensorFlow/Keras: Frameworks used for building and training the Deep Neural Network model.**
- **Jupyter Notebook: Environment for data analysis, model experimentation, and testing.**

CHAPTER: 2 PROJECT SCOPE

CHAPTER 2 PROJECT SCOPE

The scope of this project is to develop an AI-powered hotel price prediction system that leverages machine learning models to forecast hotel pricing based on various features, including location, hotel type, amenities, seasonality, and demand. This system is designed to assist hotel managers, pricing strategists, and other stakeholders in making informed pricing decisions by providing accurate, data-driven price recommendations.

The primary goal of this project is to build a robust model using Random Forest Regressor and Deep Neural Networks (DNN) to predict hotel prices based on historical data. The system is capable of processing large datasets of hotel records stored in MySQL, and it incorporates feature extraction and selection techniques to ensure the quality and relevance of input data. The model will be trained on diverse hotel data, allowing it to generalize well across different hotel types and geographical locations.

Key components of the system include:

- Data Collection and Storage: Hotel data is manually entered into MySQL, where it is stored and made available for processing.
- Machine Learning Models: The project uses Random Forest Regressor and Deep Neural Networks (DNN) for price prediction. These models will be evaluated for accuracy, with the goal of determining the most effective approach for forecasting hotel prices.
- Feature Engineering: Techniques such as feature extraction and selection will be used to enhance model performance by identifying the most important variables influencing hotel prices.
- User Interface (UI): While this project focuses on the model itself, the long-term scope includes the development of a user-friendly interface to allow non-technical users to input hotel attributes and receive price predictions.

This project aims to provide an automated, scalable, and data-driven solution for dynamic hotel pricing, helping stakeholders optimize revenue management strategies and gain competitive insights. It will serve as a tool to streamline pricing processes in the hospitality industry by incorporating AI/ML-driven predictions.

CHAPTER: 3 SOFTWARE AND HARDWARE REQUIREMENTS

CHAPTER 3 SOFTWARE AND HARDWARE REQUIREMENTS

Minimum Hardware Requirements

Component	Specification
Processor	Intel Core i3 / AMD Ryzen 3 (or equivalent)
RAM	4GB
HDD	30GB

Table 3.1 Minimum Hardware Requirements

Minimum Software Requirements

Component	Specification
Operating System	Windows, macOS, or Linux
Web Browser	Chrome, Firefox, Edge, or Safari (latest versions recommended)
Python	Python 3.7 or above
Machine Learning Libraries	Scikit-learn, TensorFlow/Keras, Pandas, NumPy (for model training and evaluation)
Database	MySQL (for storing hotel records and querying the data)
IDE/Notebook	Jupyter Notebook or any preferred IDE (e.g., VSCode) for model development and experimentation

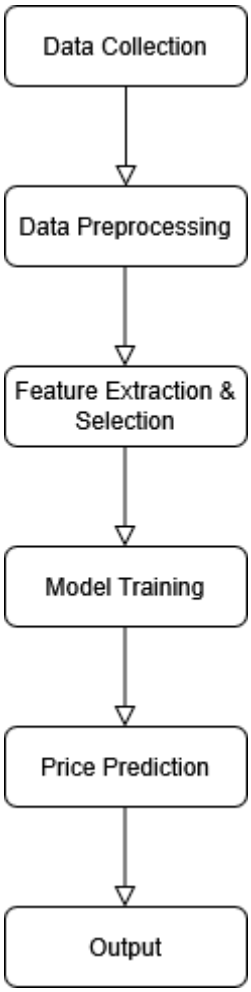
Table 3.2 Minimum Software Requirements

CHAPTER: 4 PROCESS MODE

CHAPTER 4 PROCESS MODEL

This project follows a structured workflow to ensure the accurate prediction of hotel prices based on various features. The system processes historical hotel data and user inputs, applies machine learning models, and generates price predictions. The workflow includes data collection, preprocessing, feature engineering, model training, and prediction. Below is a simplified diagram illustrating the process:

1. Data Collection: Hotel records are manually entered into MySQL database, including attributes like location, amenities, seasonality, and hotel type.
2. Data Preprocessing: The collected data is cleaned, and missing values are handled to ensure consistency.
3. Feature Extraction & Selection: Key features affecting hotel prices are identified and selected for training the machine learning models.
4. Model Training: Random Forest Regressor and Deep Neural Network models are trained using the preprocessed data to learn pricing patterns.
5. Price Prediction: When a user inputs hotel characteristics (such as location, amenities, and type), the trained models predict the optimal price based on the learned patterns.
6. Output: The predicted price is returned to the user as a result.



CHAPTER: 5 PROJECT PLAN

CHAPTER 5 PROJECT PLAN

5.1 List of Major Activities

1. **Research & Exploration** – Understanding the problem domain, exploring relevant machine learning models (Random Forest, Deep Neural Networks), and reviewing feature engineering techniques.
2. **Data Collection & Database Setup** – Manually collecting hotel data and setting up MySQL for storing hotel records.
3. **Data Preprocessing & Feature Engineering** – Cleaning the collected data, handling missing values, and selecting key features that affect hotel prices.
4. **Model Development & Training** – Training the Random Forest Regressor and Deep Neural Network models using the processed data.
5. **Model Evaluation & Tuning** – Evaluating model performance using appropriate metrics (e.g., RMSE, MAE) and fine-tuning hyperparameters.
6. **Backend Development (Optional)** – Developing the backend using FastAPI to handle user input and process predictions.
7. **Frontend Development (Optional)** – Designing a simple user interface where users can input hotel features and view the predicted price.
8. **System Integration (Optional)** – Integrating the machine learning models with the backend and connecting it to the frontend.
9. **Testing & Debugging (Optional)** – Ensuring the system's stability, testing the models' accuracy, and ensuring smooth user interactions.
10. **Final Deployment & Documentation (Optional)** – Deploying the system for use and preparing project reports and user documentation.

5.2 Estimated Time Duration in Days

Activity	Estimated Duration (Days)
Research & Exploration	10 Days
Data Collection & Database Setup	8 Days
Data Preprocessing & Feature Engineering	12 Days
Model Development & Training	14 Days
Model Evaluation & Tuning	10 Days
Backend Development (Optional)	12 Days
Frontend Development (Optional)	12 Days
System Integration (Optional)	8 Days
Testing & Debugging (Optional)	10 Days
Final Deployment & Documentation (Optional)	6 Days

Total Estimated Duration: ~2.5 Months

Table 5.1 Task Completion Estimated Time Duration in Days

CHAPTER: 6 IMPLEMENTATION DETAILS

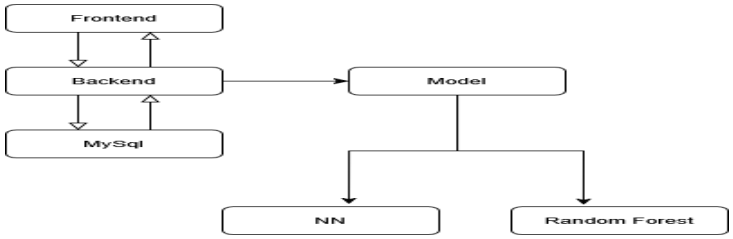
CHAPTER 6 IMPLEMENTATION DETAIL

This chapter provides an in-depth explanation of the implementation of the AI-based hotel price prediction system. It covers the system's architecture, workflow, integration of technologies, and a step-by-step breakdown of each module involved.

6.1 System Architecture

The system follows a client-server architecture, where the frontend, backend, and database communicate seamlessly to process hotel price predictions based on user inputs. The main components of the system include:

1. **Frontend (React.js)**:
   The frontend of the system is built using React.js. It provides a user-friendly interface where users can input various hotel attributes such as location, hotel type, amenities, and seasonality. The interface is designed to be intuitive, allowing users to interact with the system easily. After receiving the input, the frontend sends the data to the backend for processing and displays the predicted hotel price to the user.
2. **Backend (FastAPI)**:
   The backend is developed using FastAPI, which is used to handle user requests. It processes the input data sent from the frontend and passes it to the machine learning models for price prediction. FastAPI communicates with the trained machine learning models (Random Forest Regressor and Deep Neural Network) and retrieves the predicted prices. The backend also manages the interaction with the MySQL database, where the hotel data is stored.
3. **Machine Learning Models (Random Forest Regressor & Deep Neural Network)**:
   The system uses two models for price prediction:
   - **Random Forest Regressor**: An ensemble learning method that uses decision trees to predict the price based on historical data and selected features.
   - **Deep Neural Network (DNN)**: A neural network model designed to learn complex patterns in the hotel pricing data and make predictions. The model was trained using a deep learning framework (e.g., TensorFlow or Keras).
4. **Database (MySQL)**:
   MySQL is used as the database to store hotel records, including attributes like location, hotel type, amenities, and pricing data. The database is pre-populated with hotel data that is used for training the models and making price predictions. The backend queries the database to retrieve relevant hotel data for input into the machine learning models.



6.2 Workflow Implementation

This section outlines the detailed workflow of the hotel price prediction system, covering each stage from user input to the final price prediction. It includes interactions between the frontend, backend, machine learning models, and database.

6.2.1 User Input & Frontend Processing (Planned/Optional)

- The user interacts with the **React-based frontend** by entering hotel-related features, such as:
  - **Location** (e.g., city or region)
  - **Hotel Type** (e.g., budget, luxury)
  - **Amenities** (e.g., free Wi-Fi, pool)
  - **Seasonality** (e.g., peak season or off-season)
- Once the user submits their input, the frontend sends this data to the **FastAPI backend** in the form of a structured API request.

6.2.2 Feature Extraction & Preprocessing (Data Processing in Backend)

- The backend receives the hotel feature data from the frontend and performs **data preprocessing** to ensure it is in a usable format for the machine learning models.
- The backend may handle tasks like:
  - **Handling missing values**: Ensuring that all necessary features are present and valid.
  - **Normalizing or encoding**: Converting categorical data (e.g., hotel type, location) into numerical representations if necessary.
- Preprocessed data is then passed to the **machine learning models** for price prediction.

6.2.3 Price Prediction via Machine Learning Models

- The **Random Forest Regressor** and **Deep Neural Network (DNN)** models, trained on historical hotel data, are used to predict the price.
  - **Random Forest Regressor**: An ensemble model that predicts hotel prices based on various features by aggregating predictions from decision trees.
  - **Deep Neural Network (DNN)**: A neural network model that can capture complex, non-linear relationships between hotel features and price.
- The backend sends the preprocessed input to the models, which generate the **predicted hotel price** based on the given features.

6.2.4 Query Execution & Result Fetching

- After receiving the predicted price, the backend prepares the data to be sent back to the frontend.
- The system queries the **MySQL database** (if needed) to fetch any related records or historical data that may help in validating the prediction.
- The predicted price and any supplementary data (if available) are packaged in a **structured format** (e.g., a JSON response) and sent back to the frontend.

6.2.5 Displaying Results to the User (Yet to be Integrated with Frontend)

- The **React frontend** receives the predicted price (and any relevant supplementary information) from the backend.
- The results are displayed to the user in a clear and readable format, such as:
    - A **predicted hotel price** displayed on the screen.
    - A **summary of the key features** (location, hotel type, etc.) influencing the price.
- If necessary, additional features (like graphs or visualizations) may be added to help the user understand the factors affecting the price.

6.3 Machine Learning Model Integration and Prediction Workflow

This section outlines how the machine learning models (specifically the Random Forest Regressor and Deep Neural Network) interact with the system to predict hotel prices. The models are trained to process the input features (such as location, hotel type, amenities) and generate price predictions. This section also explains the design of the prompt template, which ensures the models receive the correct input format for accurate predictions.

6.3.1 Feature Engineering Template Design

The feature engineering template plays a critical role in preparing the input data for the machine learning models. It defines the structure and format of the input data to ensure that the features are extracted and preprocessed correctly before being sent to the models. The template focuses on providing context about the data required for accurate predictions. It also ensures that:

- Categorical features (e.g., hotel type, amenities) are encoded appropriately.
- Numerical features (e.g., number of rooms, hotel rating) are normalized or scaled.
- Missing data is handled appropriately, either by imputation or exclusion.

The feature engineering template helps streamline the process of transforming raw user input into a format that the models can understand, allowing for more accurate price predictions.

6.3.2 Machine Learning Model Interaction

The Random Forest Regressor and Deep Neural Network (DNN) are responsible for generating hotel price predictions based on the user's input. The interaction between the system and the models is as follows:

- User Input: The user provides input on hotel-related features such as location, hotel type, and amenities.
- Data Preprocessing: The backend processes this input data based on the feature engineering template. The input is transformed into a numerical format that can be fed to the models.
- Model Processing: The preprocessed input is passed to the Random Forest Regressor or DNN for price prediction. The models generate a predicted hotel price based on the trained data.
- Response Evaluation: The predicted price is returned to the backend. The backend evaluates whether the prediction is valid, and any necessary adjustments or refinements are made before sending it to the frontend.

Once the models generate the price prediction, the backend sends the results back to the frontend, which displays them to the user in an understandable format.

6.4 Backend Implementation - FastAPI (Planned/Optional)

6.5 Frontend Implementation (React.js) (Planned/Optional)

6.6 Database Structure

Booking_ID	Unique Varchar
Room Type	Varchar
Availability	Integer
Date	Date
Holiday	Boolean
Price	Integer
Guest_Name	Varchar
Email	Varchar
Phone Number	Varchar

6.7 Conclusion

This chapter provides an overview of the current status of the hotel price prediction system and highlights the next steps required to complete the project. While some core components of the system, such as machine learning models (Random Forest Regressor and DNN), have been defined, the backend and frontend are still under development.

Key aspects of the project that remain to be developed include:

- Backend Development: The FastAPI-based backend, which will process user inputs, interact with the machine learning models, and manage database queries, is yet to be implemented.
- Frontend Development: The React-based user interface for user interaction and displaying price predictions is still in progress.
- Error Handling: The error handling module to validate predictions and ensure the system functions reliably needs to be developed.
- Security Mechanisms: Implementing role-based access control and securing the system from potential vulnerabilities will be done in the coming stages.
- Cloud Deployment: The system is currently running locally, and cloud deployment to ensure scalability and wider access will be a future task.

As the project progresses, more detailed documentation will be added, including UI screenshots, workflow diagrams, and backend response examples. This will help provide a clearer understanding of how each component functions and integrates with the system**.**

CHAPTER: 7 CONCLUSION AND FUTURE WORK

7.1 Conclusion

This project aims to leverage machine learning models to predict hotel prices based on user inputs, simplifying the pricing prediction process for users in the hospitality industry. By integrating machine learning models (Random Forest Regressor and Deep Neural Network) that can generate dynamic hotel price predictions based on a range of features, such as location, hotel type, and amenities.

So far, we have successfully established the machine learning models, set up the database for storing relevant hotel data, and created the initial framework for the backend and frontend. Although the backend and frontend development are ongoing, the key components of data preprocessing, feature engineering, and model training have been completed, demonstrating the feasibility of the approach.

The system, once fully developed, will provide a user-friendly interface where users can input hotel characteristics, and the system will predict hotel prices based on trained models. This will significantly streamline the pricing decision process for hotels, enabling users to make more informed, data-driven decisions.

7.2 Future Work

Moving forward, several critical steps will be undertaken to enhance and expand the system's capabilities:

- **Full-Stack Integration**: Connecting the **backend (FastAPI)** with the **frontend (React.js)** to enable real-time hotel price prediction. This will allow users to interact with the system and receive predictions based on their input features dynamically.
- **Dynamic Feature Handling**: Implementing a more dynamic system for handling input features and ensuring that only the relevant features (e.g., location, hotel type) are used for each prediction. This will optimize the prediction process and improve system flexibility.
- **Enhanced Error Handling and Security**: Improving the robustness of the system by adding **error handling** for invalid or incomplete inputs and implementing **role-based access control (RBAC)** to secure the application and restrict unauthorized access.
- **Model Optimization and Testing**: Exploring alternative machine learning models, such as **Gradient Boosting Machines (GBM)** or **XGBoost**, to compare their performance and accuracy with the current models. Fine-tuning the current models for improved prediction accuracy and efficiency is also planned.
- **User Testing and Interface Improvements**: Conducting thorough **user testing** to refine the frontend interface, ensuring that it is intuitive and user-friendly. Based on feedback, we will make adjustments to improve the overall user experience.
- **Documentation and Deployment**: Finalizing the system's **documentation** and preparing for **cloud deployment** to ensure scalability and easy access for a wider audience. This will involve transitioning the system from a local environment to a cloud platform for production use.

In conclusion, while significant progress has been made in developing the foundational components of the system, ongoing work will focus on full-stack integration, optimizing the machine learning models, and preparing the system for deployment and broader use. The successful completion of these steps will result in a robust, user-friendly platform capable of delivering accurate hotel price predictions.

CHAPTER: 8 REFERENCES

CHAPTER 8 REFERENCES

1. Random Forest Regressor - Scikit-learn Documentation
   https://scikit-learn.org/stable/modules/ensemble.html#random-forest
2. Deep Learning for Regression Models - Towards Data Science
   https://towardsdatascience.com/deep-learning-for-regression-5e0d17b15657
3. Pandas Documentation (read_sql method)
   https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.read_sql.html
4. FastAPI Documentation
   https://fastapi.tiangolo.com/
5. React.js Documentation
   https://reactjs.org/docs/getting-started.html
6. Hotel Price Prediction with Machine Learning
   https://www.analyticsvidhya.com/blog/2021/10/a-complete-guide-to-hotel-price-prediction-with-machine-learning/
7. Hotel Data Analysis with Python and Pandas
   https://www.kaggle.com/atharvaingle/hotel-price-prediction-using-machine-learning