COURSE: CPE 221 -> COMPUTING AND SOFTWARE ENGINEERING - DATA ANALYSIS

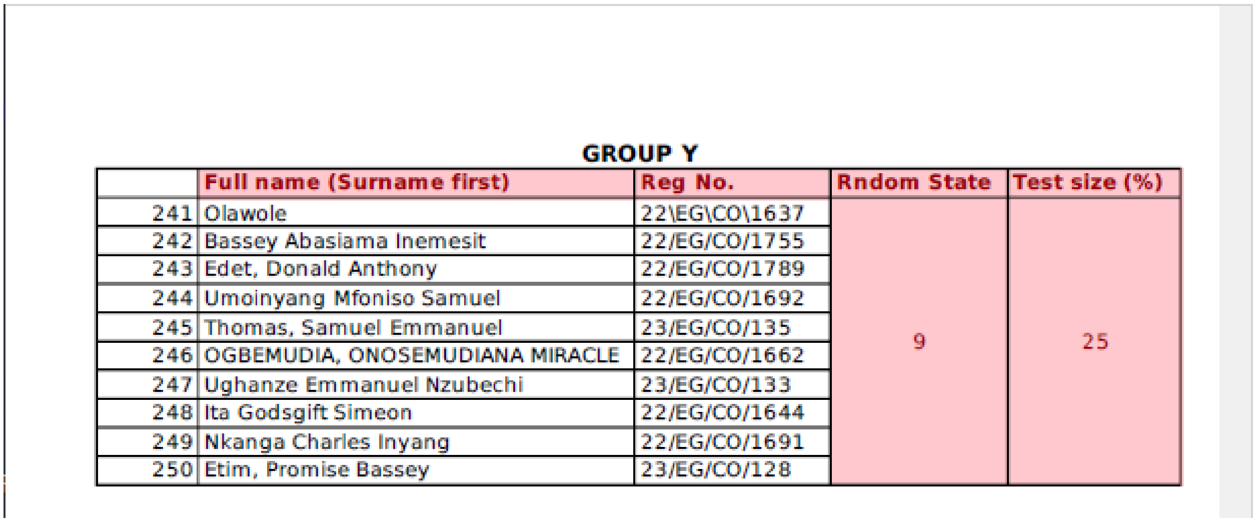
MINI-PROJECT: LAPTOP PRICE PREDICTION

GROUP: GROUP Y

DATE: 09 SEPT. 2025

Note: Olawole with REG NO: 22/EG/CO/1637 was registered in another group

Total group Members: 9



GROUP Y

	Full name (Surname first)	Reg No.	Rndom State	Test size (%)
241	Olawole	22\EG\CO\1637	9	25
242	Bassey Abasiama Inemesit	22/EG/CO/1755		
243	Edet, Donald Anthony	22/EG/CO/1789		
244	Umoinyang Mfoniso Samuel	22/EG/CO/1692		
245	Thomas, Samuel Emmanuel	23/EG/CO/135		
246	OGBEMUDIA, ONOSEMUDIANA MIRACLE	22/EG/CO/1662		
247	Ughanze Emmanuel Nzubechi	23/EG/CO/133		
248	Ita Godsgift Simeon	22/EG/CO/1644		
249	Nkanga Charles Inyang	22/EG/CO/1691		
250	Etim, Promise Bassey	23/EG/CO/128		

Introduction

The goal of this project was to develop a solution for predicting the prices of laptops using a real-world dataset. This involved steps from data cleaning and exploratory analysis to feature engineering, model building, evaluation, and deployment using a Streamlit web application. The dataset used contains laptop specifications such as manufacturer, type, CPU, GPU, RAM, storage, operating system, weight, screen size, and price in euros. By understanding how these features interact, the project aims to provide accurate price predictions and insights into the factors that most influence laptop pricing.

A. Data Cleaning and Preprocessing

Upon inspecting the dataset, it was confirmed that there were no missing values, which meant that no imputation was required. This is a good indicator of the dataset's quality and allows the model to train without introducing biases from missing data handling.

Categorical variables were handled using two encoding strategies:

1. One-Hot Encoding was applied to Company, TypeName, OpSys, CpuBrand, and GpuBrand. This approach was chosen because these variables represent non-ordinal categories where no inherent ordering exists. One-hot encoding allows the model to treat each category independently, avoiding assumptions about relative rankings.

2. Target Encoding was applied to CPU Name and GPU Name. These features have a large number of unique values, making one-hot encoding inefficient and prone to high dimensionality. Target encoding replaces each category with the mean price of laptops belonging to that category, providing meaningful numeric input while keeping the model compact.

No outliers were removed, as preliminary visualization showed the numerical features to be reasonably distributed, and extreme values appeared to reflect real-world laptops.

B. Exploratory Data Analysis (EDA)

The dataset was analyzed using histograms, scatter plots, box plots, and correlation heatmaps to identify patterns and relationships. Key findings include:

1. Laptop price generally increases with higher RAM, SSD storage, higher PPI, and better CPU performance.
2. Gaming laptops and ultrabooks tend to command higher prices than basic notebooks.

3. CPU and GPU brands significantly impact pricing, reinforcing the choice of target encoding for these columns.
4. EDA confirmed that the engineered features were meaningful and likely to improve predictive accuracy.

C. Feature Engineering

To enhance model performance, several derived features were added:

1. Pixels per Inch (PPI): Calculated from screen resolution and screen size to capture display clarity, which can influence price.

2. High RAM Indicator: A binary feature representing whether the laptop has above-average RAM, as higher memory often increases value.

3. SSD Indicator: Indicates whether the storage is SSD, reflecting faster performance.

4. Storage Type: Distinguishes between HDD, SSD, or hybrid storage setups.

5. CPU Performance Class: Categorizes CPUs into performance tiers to capture processing power more effectively than the raw CPU names alone.

These features were designed to capture factors that directly affect laptop value, making the model more sensitive to meaningful variations in hardware and display specifications.

D. Model Building

The dataset was split into training and testing sets, with 25% of the data reserved for testing and a random state of 9. A linear regression model was trained on the processed features, including the original encoded columns and the newly engineered features. This model was chosen due to its interpretability and its ability to capture linear relationships between laptop specifications and price and the data can be easily interpreted by the model.

E. Model Evaluation

The trained model was evaluated on both the training and test sets to assess performance and generalization:

Training Set: RMSE = 258.98, MAE = 192.34, $R^2 = 0.85$

Test Set: RMSE = 295.88, MAE = 214.96, $R^2 = 0.85$

The metrics demonstrate that the model captures the underlying relationships effectively without overfitting. The low errors and high R^2 indicate that features such as CPU, RAM, GPU, storage type, and screen resolution play critical roles in determining price.

F. Streamlit App Deployment and Usage

To make the model accessible, a Streamlit web application was created. The app takes user input for laptop specifications and outputs the predicted price.

Before prediction, the app:

1. Encodes categorical features the same way as during training: one-hot encoding for Company, TypeName, OpSys, CpuBrand, YouBrand and target encoding for CPU Name and GPU Name.

2. Computes the engineered features: PPI, High RAM, SSD, Storage Type, and CPU Performance Class.

After preprocessing, the trained linear regression model predicts the laptop price. This ensures predictions remain consistent with the model training process.

Instructions for Use:

1. Install required packages listed in requirements.txt.

2. Launch the app with:

```
streamlit run app.py
```

3. Enter laptop specifications into the input fields.

4. The app will encode the inputs, calculate derived features, and display the predicted price.

5. Users can experiment with different configurations to explore price variations based on hardware changes.

The Streamlit interface is intuitive and designed for both technical and non-technical users, making laptop price prediction accessible and interactive.

G. Conclusion

This project successfully demonstrates the approach to creating a laptop price prediction model. The combination of careful preprocessing, feature engineering, and linear regression modeling produced a reliable and interpretable prediction tool. Also, Deploying the model through Streamlit allows users to interact with the model in real time, offering insights into how hardware specifications influence price.