Explain how you improvised the project further?

I tried several feature engineering techniques to improve the model's performance:

1. **Discretized Continuous Features**: I binned continuous features like age, balance, and duration into categories to simplify relationships and capture non-linear patterns.
2. **Log Transformation or Normalization**: I applied logarithmic transformations to features such as balance, previous, and duration to reduce skewness.
3. **Outlier Handling**: I capped outliers for features like campaign, previous, and balance to reduce their impact on the model.
4. **Feature Encoding**: I encoded pdays to indicate whether the customer was previously contacted.
5. **New Temporal Features**: I generated features related to the recency of contact and differentiated between first-time and repeat contacts.
6. **Campaign Success Rate**: I created a success\_rate feature to understand the effectiveness of the campaign efforts.
7. **Ratios and Combinations**: I derived ratios such as balance per contact and duration per campaign to measure efficiency.
8. **Binary Features Based on Conditions**: I added binary features to indicate high engagement or repeated targeting.

Despite trying these feature engineering methods, the model's performance score actually fell from around 74 to 68.

I performed hyperparameter tuning on both logistic regression and random forest classifier models using grid search and random search cross-validation. The results were mixed:

1. For logistic regression, tuning improved the recall to 82% but reduced precision to 41%. While the F1 score increased to 70%, this still fell short of the random forest classifier's original score of 74%.
2. The hyperparameter tuning of the random forest classifier was counterproductive, actually decreasing the score from 74% to 73% or lower, despite extensive time spent on grid search and random search cross-validation. Various adjustments were attempted, including:
   1. Undersampling the data
   2. Reducing cross-validation folds from 5 to 3, then to 2
   3. Modifying max\_iter to 10

Ultimately, the most effective improvement came from a simple adjustment: increasing the number of estimators in the random forest classifier from the default of 100 to 800. This straightforward change boosted the score from approximately 74% to 75%, finally exceeding our target threshold.