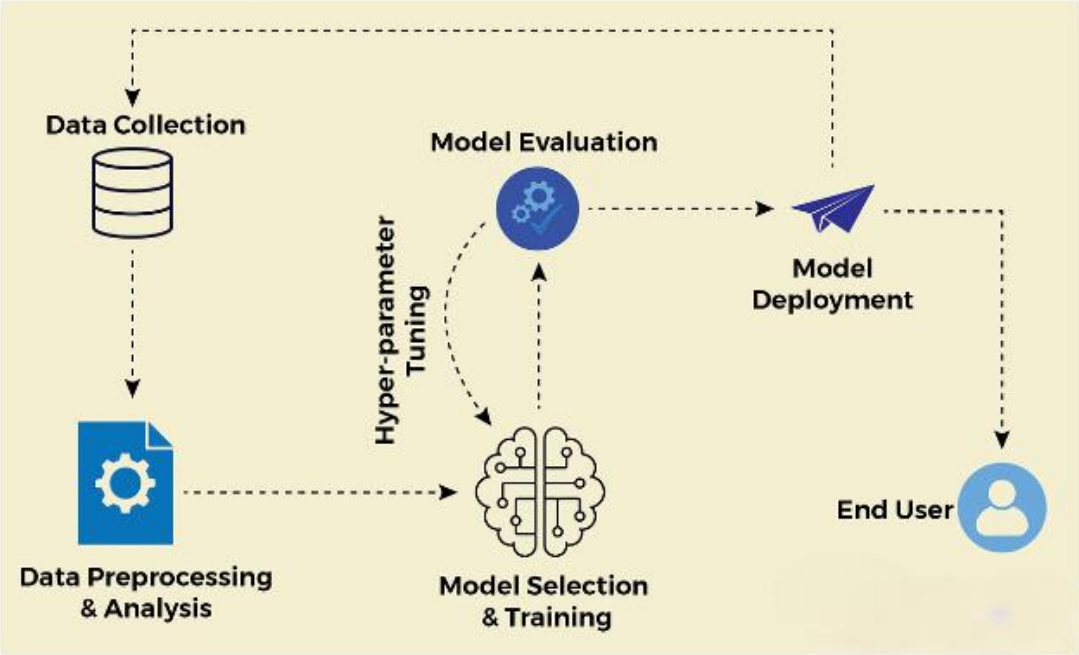
Part 2 - Write documentation on the plan & design you will be following to build this automated machine learning system.

1. **Introduction.**

In designing an automated machine learning (ML) system for production, the goal is to create a scalable, maintainable, and efficient ML pipeline. My system will enable automated data preprocessing, model training, evaluation and deployment ensuring the model remains effective and adapts to new data over time.

2. **System Overview.**

The system will be structured as an end-to-end ML pipeline with the following stages.



A. Data Preprocessing.

1. **Combine Datasets** - Use the join method to merge datasets into a single dataset, creating a comprehensive deployment dataset.
2. **Select Target and Input Features** - Choose the target column and relevant input features from the combined dataset.
3. **Encode Categorical Columns** - Apply label encoding to convert categorical variables into numerical form.
4. **Split Data** - Divide the dataset into training (80%) and testing (20%) sets.
5. **Impute Missing Values** - Fill missing values using mean values.

B. Feature Engineering.

1. **Calculate Mutual Information** - Measure the mutual information between the target variable and each feature.
2. **Set Feature Selection Threshold** - Choose a threshold value (0.05) to select the top features based on their mutual information scores.
3. **Save Preprocessing Artifacts** - Store encoder values imputed mean values, and the selected top features for deployment.

C. Model Selection and Training.

1. **Choose LightGBM Model** - Use LGBMClassifier and configure model parameters.
2. **Hyperparameter Tuning** - Perform hyperparameter tuning using RandomizedSearchCV.
3. **Cross-Validation** - Validate model performance with cross-validation.
4. **Get Best Model** - Retrieve the model with optimal parameters.

D. Model Evaluation.

1. **Evaluate with AUC** - Assess model performance on the test set using the Area Under the Curve (AUC) metric.
2. **Plot Training and Validation Performance** - Visualize the model's performance over training and validation phases.
3. **Plot Confusion Matrix** - Create a confusion matrix to analyze classification outcomes.
4. **Generate Classification Report** - Obtain a detailed classification report.

E. Save the Model for Deployment.

1. **Save the trained model** - For future deployment and use.

F. Model Deployment.

1. **Load Saved Model** - Load the trained LightGBM model.
2. **Load Deployment Dataset** - Load the dataset prepared for deployment testing.
3. **Load Encoders and Imputer Values** - Retrieve saved encoder mappings and imputed mean values.
4. **Load Selected Features** - Load the top features identified during feature selection.
5. **Apply Preprocessing** - Use saved preprocessing values to prepare the new dataset.
6. **Predict Model Accuracy** - Evaluate the model's accuracy on the new dataset.