

# End-to-End ML System Design (From Implementation to Production)

This section outlines the high-level architecture and lifecycle of deploying the term deposit prediction model into a real production environment for weekly marketing operations.

---

## 1. Data Ingestion

- **Source:** Historical campaign data (e.g., `bank-additional-full.csv`) is pulled from internal databases or updated weekly.
  - **Format:** Structured tabular data with both categorical and numerical features.
  - **Automation:** A scheduled ETL pipeline ingests and cleans data weekly, ready for inference or retraining.
- 

## 2. Data Processing & Feature Engineering

- Missing values are imputed (`unknown` replaced with mode).
  - Categorical features are label-encoded or one-hot encoded as needed.
  - SMOTE is applied to handle class imbalance.
  - Feature scaling applied using `StandardScaler`.
  - Pipeline logic modularized in scripts: `dataset.py`, `features.py`.
- 

## 3. Model Training

- Models are trained using cross-validation (`StratifiedKFold`) and hyperparameter optimization (`BayesSearchCV`).
  - Multiple classifiers (e.g., Logistic Regression, Random Forest, XGBoost) are evaluated.
  - Best model is saved as `best_model.pkl` in a persistent storage directory.
  - Training pipeline can be triggered manually or scheduled for retraining (e.g., monthly).
- 

## 4. Inference & Serving

- A lightweight **Flask app** (`app.py`) or API server serves predictions.
  - New weekly customer cohorts are scored via batch or real-time API.
  - Output: Probability and binary decision (`will convert` / `won't convert`) used to filter high-likelihood contacts.
- 

## 5. Business Integration

- The prediction output is integrated into the call center workflow:
    - Only customers above a certain probability threshold are called.
    - Remaining contacts are deprioritized to optimize cost savings.
- 

## 6. Model Versioning & Deployment

- Models are versioned in a `models/` directory or stored in cloud object storage.

- Deployment is containerized (e.g., with Docker) for portability.
  - CI/CD pipelines can be set up to test and deploy new models safely.
- 

## 7. Monitoring & Maintenance

- **Monitoring Tools:**
    - Log prediction outcomes and track model drift.
    - Monitor key business KPIs: conversion rate, call cost, profit uplift.
  - **Alerts:**
    - Trigger alerts if model confidence or performance drops.
  - **Feedback Loop:**
    - Weekly campaign outcomes are fed back into the system for continuous learning and retraining.
- 

## Optional Enhancements

- **A/B Testing:** Test model recommendations vs. business-as-usual.
- **Explainability:** Use SHAP for model transparency to stakeholders.
- **Bias Checks:** Ensure fair treatment across demographics.