**Development Plan: Automated Model Selection & Hyper parameter Optimization Using Bayesian Optimization**

**🔹 1. Project Initialization**

* Set up a clean Python/Notebook environment
* Install required libraries:

Bash

CopyEdit

pip install scikit-learn optuna xgboost

**🔹 2. Dataset Preparation**

* Load the dataset (load\_breast\_cancer or custom dataset)
* Optionally preprocess using:
  + StandardScaler
  + Handling missing values or categorical encoding (if applicable)

**🔹 3. Model & Hyperparameter Space Definition**

* Define candidate models:

pandom Forest": RandomForestClassifier (),

"SVM": SVC(),

"XGBoost": XGBClassifier()

}

* Define search space in Optuna for each model:
  + For RandomForest: n\_estimators, max\_depth
  + For SVM: C, kernel, gamma
  + For XGBoost: learning\_rate, max\_depth, n\_estimators

**🔹 4. Bayesian Optimization with Optuna**

* Use Optuna’s study.optimize() to:
  + Suggest model + hyper parameters
  + Evaluate using cross\_val\_score
  + Return score (maximize accuracy)

**🔹 5. Model Evaluation**

* Track best model + parameters from Optuna
* Retrain best model on full training data
* Optionally evaluate on test set

**🔹 6. Result Visualization**

* Plot optimization history
* Print best model and hyper parameters

**🔹 7. Optional Enhancements**

* Use OptunaDashboard for GUI-based monitoring
* Add support for custom datasets
* Export best model using joblib or pickle

**📦 Project Output**

* Final best model (saved)
* Report of model selection + hyper parameter tuning
* Accuracy or metric chart
* Optional: GUI for interaction or model upload