

```

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report
import pickle
import streamlit as st

# Load or simulate soil sensor data
data = {
    'Moisture': np.random.uniform(10, 50, 100),
    'pH': np.random.uniform(4.5, 8, 100),
    'Nitrogen': np.random.uniform(10, 100, 100),
    'Phosphorus': np.random.uniform(5, 50, 100),
    'Potassium': np.random.uniform(5, 50, 100),
    'SoilQuality': np.random.choice(['Poor', 'Moderate', 'Good'], 100)
}
df = pd.DataFrame(data)

# Encode labels
df['SoilQuality'] = df['SoilQuality'].astype('category').cat.codes

# Split data
X = df.drop('SoilQuality', axis=1)
y = df['SoilQuality']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train ML model
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

# Save the model
with open('soil_model.pkl', 'wb') as f:
    pickle.dump(model, f)

# Streamlit App
def create_streamlit_app():
    st.title("Soil Quality Prediction")

```

```
moisture = st.slider("Moisture", 10.0, 50.0, 30.0)
ph = st.slider("pH", 4.5, 8.0, 6.5)
nitrogen = st.slider("Nitrogen", 10, 100, 50)
phosphorus = st.slider("Phosphorus", 5, 50, 25)
potassium = st.slider("Potassium", 5, 50, 25)
```

```
if st.button("Predict"):
    features = np.array([moisture, ph, nitrogen, phosphorus, potassium]).reshape(1, -1)
    with open('soil_model.pkl', 'rb') as f:
        model = pickle.load(f)
    prediction = model.predict(features)[0]
    st.write(f"Predicted Soil Quality: {prediction}")
```

```
create_streamlit_app()
```



