Tugas Besar — Proyek Machine Learning I
(*Aplikasi Python + Streamlit — Domain: Kesehatan atau Pertanian)*

1. Tujuan umum
   Mahasiswa merancang, mengimplementasikan, dan mendokumentasikan sebuah aplikasi ML berbasis ***Python + Streamlit*** yang memecahkan masalah nyata di bidang ***kesehatan atau pertanian***.
   Proyek mencakup seluruh tahapan ML:
   EDA → preprocessing → modelling → evaluasi → deployment sederhana.

2. Ketentuan wajib proyek
   a) Domain: Pilih salah satu *Kesehatan* (mis. prediksi penyakit / klasifikasi citra) atau *Pertanian* (mis. klasifikasi penyakit tanaman / prediksi hasil panen).
   b) Ukuran dataset: Minimal 500 sampel yang relevan untuk masalah (label balance tidak wajib sempurna, tetapi jelaskan). Sumber data: dataset publik, data institusi dengan izin, atau hasil sampling.
   c) Bahasa & tools: Python (≥3.8), library ML (scikit-learn, XGBoost/LightGBM), SHAP (atau permutation importance), Streamlit untuk UI.
   d) Versi kontrol: Submit repository Git (GitHub/GitLab). Pastikan commit historinya jelas.
   e) Privasi & etika: Bila data medis/personal, sertakan pernyataan anonimasi & izin penggunaan data.
   f) Dokumentasi: README.md, cara setup environment (requirements.txt), dan petunjuk run app.
   g) Deliverables akhir: kode (repo), file dataset (atau link), notebook analisis, aplikasi Streamlit (hosted atau cara run lokal), laporan akhir (PDF), dan presentasi (pptx).

3. Lembar tugas rincian pekerjaan & milestone (mingguan)
   - Minggu 1 (Kickoff)
     - Pilih topik & dataset.
     - Kirim *one-page proposal* (judul, latar, target, sumber data, metode awal).
   - Minggu 2 (Bab 1–2 draft)
     - Upload Bab 1 (latar belakang, masalah) & Bab 2 (tinjauan pustaka singkat).
     - Upload dataset (CSV) dan preview EDA singkat.
   - Minggu 3 (Bab 3 draft & prototyping / Implementasi)
     - Implementasi preprocessing & baseline model (Logistic/Decision Tree/Linear).
     - Notebook awal + uji metric.
   - Minggu 4 (Integrasi aplikasi & evaluasi lanjutan)
     - Finalize modelling: tambahkan XGBoost/LightGBM, cross-validation, confusion matrix, precision, recall, F1, AUC.
     - SHAP / feature importance.

    - Streamlit app (basic UI: upload data, run model, show metrics, explainability).
- Minggu 5 (Finalisasi & presentasi)
    - Submit laporan akhir (Bab 1-3), repo lengkap, link app (atau instruksi run), dan presentasi 10–15 menit.

4. Komponen teknis yang harus ada (detail minimum)

A. Data & EDA
- Deskripsi dataset: fitur, target, ukuran, contoh 5 baris sebelum & setelah preprocessing.
- Visualisasi: distribusi target, korelasi, boxplots, missingness heatmap.
- Penanganan missing & outlier: jelaskan metode dan alasan.

B. Preprocessing pipeline (harus reproducible)
- Encoding kategorikal (one-hot / target encoding jika high-cardinality)
- Scaling (StandardScaler / MinMax) untuk fitur numerik
- Train-validation-test split: jelaskan strategi (temporal? stratified?) dan pastikan no leakage.
- Cross-validation: minimal 5-fold (stratified jika klasifikasi).

C. Modelling (minimal)
- Baseline: Logistic Regression / Decision Tree.
- Advanced: Random Forest + XGBoost atau LightGBM (wajib menambahkan salah satu boosting).
- Hyperparameter tuning: GridSearchCV / RandomizedSearchCV (laporkan best params).
- Evaluasi: Confusion Matrix, Accuracy, Precision, Recall, F1-score (khusus kelas minor), AUC-ROC (jika relevan), dan *per-fold* stats (mean ± std).

D. Interpretability
- SHAP summary plot dan setidaknya 2 SHAP dependence plots untuk fitur penting, atau permutation importance jika SHAP tidak feasible.
- Penjelasan singkat: mengapa fitur penting secara domain.

E. Robustness & Limitations
- Analisis ketidakseimbangan kelas & mitigasi (class weight / SMOTE / undersampling — jelaskan implikasi).
- Diskusikan keterbatasan dataset (bias, high-cardinality, integrasi multi-tabel jika ada).

F. Aplikasi Streamlit (UI minimum)
- Halaman Home: deskripsi proyek & dataset.
- Halaman Upload & Preview: upload CSV & lihat preview.
- Halaman Run Model: pilih model, tombol predict, tampilkan metrics & confusion matrix.
- Halaman Explainability: tampilkan top feature (SHAP) & contoh penjelasan prediksi.
- Halaman Download: nda export predictions CSV.

G. Reproducibility

- requirements.txt / environment.yml
- Script run_app.sh atau instruksi run streamlit run app.py
- Notebook Jupyter/Colab untuk exploratory work.

5. Format & struktur repository (recommended)

6. Contoh topik (pilihan) — cepat pilih salah satu
   Kesehatan
   - "Early detection of diabetic risk using clinical features (≥500 records)"
   - "Breast cancer classification from tabular features (Wisconsin-like) with explainable ML"
   - "Prediction of heart disease risk with LightGBM + SHAP"

   Pertanian
   - "Tomato leaf disease classification (image → features/transfer learning; ≥500 images)"
   - "Crop yield prediction per season using environmental & soil features (≥500 rows)"
   - "Clustering soil profiles for precision fertilization + classification model"

7. Penilaian kualitas minimal untuk lulus tugas
   - Dataset ≥ 500 baris, kode bisa dijalankan (streamlit run app/app.py) tanpa error.
   - Model ter-evaluasi dengan metrik lengkap (Precision/Recall/F1/Confusion matrix).
   - Ada minimal satu model boosting (XGBoost/LightGBM).
   - Ada dokumentasi singkat tentang etika & privasi data (khusus medis).
   - Laporan final mengikuti struktur Bab 1–3.