Component Interaction Matrix				
Component	Input Data	Processing	Output Data	Downstream
Layer 2: Validation	Raw fifty-six features, 15,120 obs (elevation, slope, aspect, etc.)	Validation: range, types, completeness;	Valid fifty-six features, 15,120 obs (nulls flagged/imputed, metadata timestamps, validation report)	Feature Engineering (Layer 3)
Layer 3A: Elevation Processing	Elevation (continuous) 1859–3858 m raw elevation per obs	- Bin into elevation zones (<=2400, 2400–2800, 2800–3200, >3200) - Detect proximity to thresholds (±50 m windows) - Produce distance-to-threshold metric	Elevation zone categorical Threshold flags (near_2400, near_2800...) distance_to_nearest_threshold	Model Training Feature Combin.
Layer 3B: Aspect Transform	Aspect (0–360°), Slope (cont.) (slope 0–??°)	- Convert aspect → sin(aspect), cos(aspect) - Normalize slope (robust scaler / quantile)	sin(aspect), cos(aspect) normalized_slope	Feature Combin. Model Training

		- Outlier smoothing for slope		
Layer 3C: Soil Consolidation	Forty soil types (sparse categorical) one-hot / indexes per obs	- Consolidate soil types by frequency + ecological similarity - Create fallback bucket "rare_soil." - Version mapping rules (config JSON)	Fifteen consolidated soil feature columns (one-hot) soil_group_id, soil_group_desc	Feature Combin. Model Training
Layer 3D: Distance Interact	Distances (4 types), Hillshade (3) (to hydrology/road/fire points), hillshade_9am/noon/3pm	- Normalize distances (log / min-max) - Combine distances → interaction features (e.g., ratio, product) - Generate hillshade deltas and temporal features	normalized_distances (4) interaction_features (≈6) hillshade_features	Layer 4 Training
Layer 4: Model Training	35–40 Engineered Features (train/val) labels (7 cover classes) hardware spec (CPU/GPU)	- Train base models separately: RF, XGBoost, LightGBM - Hyperparam search (Optuna, one hundred trials)	3 Base models (artifacts) per-model CV metrics, feature importances training logs, artifacts in MLflow	Ensemble Integrat. (Layer 4.2)

		- CV strategy: stratified (BUT MUST use spatial blocked CV) - Calibration step (Platt/Isotonic) opcional	inference time per model	
Layer 4.2: Ensemble Integr.	3 Base model artifacts + CV preds model metrics, latency constraints	- Ensemble: weighted voting (w= [0.3,0.4,0.3]) - Option: stacking (meta-learner) for experimental runs - Export distilled model (optional)	- Ensemble artifact (v1.0), ensemble weights - Ensemble CV performance (accuracy 95.2%) - export formats: ONNX / native boosters	Layer 5 Prediction & Uncertainty
Layer 5: Prediction & Unc.	Ensemble outputs (per-model probs) per obs; feature metadata; thresholds	- Combine probabilities: weighted average → final prob vector - Compute aleatoric uncertainty (entropy H(p) normalized by log2) - Compute epistemic uncertainty (variance across models/ensem)	Prediction JSON per obs: - cover_type (argmax) - probability distribution (7) - aleatoric, epistemic, total_uncertainty	Layer 6 Monitoring & Alerts / Decisioning

		- total_uncertainty = sqrt (aleatoric^2 + epistemic^2) - Apply threshold proximity rules: amplify uncertainty ×2 (configurable) - Generate warnings: e.g., "threshold_proximity"	- confidence_score = 1 - total_uncertainty (capped) - warnings list (text, recs)	
Layer 6: Monitoring & Drift & Sensitivity	Stream of predictions + input stats (elevation hist, feature distributions) baseline metrics (training distrib.)	- Drift detection (KL-divergence per feature, PSI, KS tests) - Threshold window counters (±50 m counts & % near thresholds) - Alerting rules (minor/major drift, %accuracy drop) - Trend analysis (accuracy week1→week4 degradation) - Risk scoring for manual review prioritization	Drift reports (KL per feature, PSI) Threshold windows: counts, % obs near windows. Alerts (Grafana / PagerDuty) Alerts (Grafana / PagerDuty) Drift_action_suggestion (retrain/canary/manual)	Ops / Retraining (canary, retrain) Field verification

Layer 7A: REST API Serving (real-time)	Single-observation request (JSON) (56 Features)	- API ingress (NGINX) → FastAPI/Uvicorn - Auth, schema validation, routing - Retrieve model artifact from model cache (Redis) or local mem - Run inference, post-process (warnings, uncertainty) - Log to Postgres (predictions_log) - Return response	API Response JSON: - prediction (cover_type, name) - probabilities (7), confidence_score - uncertainty breakdown (aleatoric/epistemic) - metadata: model_version, latency_ms - warnings & recs p50/p95/p99 latency metrics	Client apps / GIS Field apps, dashboards
Layer 7B: Batch Processing	Batch CSV / GeoTIFF (N samples) (10k → 1M samples)	- Batch orchestration (K8s jobs / Spark / GPU-batching) - Parallel GPU-accelerated inference; chunking & checkpointing - Store results to S3 (batch results + uncertainty maps)	Batch outputs: CSV/GeoTIFF/JSON - prediction per row, uncertainty maps (GeoTIFF) - processing logs & performance report	Policy teams / GIS Reforestation planning

		- Emit metrics to Grafana & update DB for audits	- summary stats (coverage by class, %near thresholds)	
Model Registry (MLflow)	Model artifacts (trained), metadata. (hyperparams, training logs)	- Register model versions (MLflow): metrics, artifacts, tags - Promote (staging → production) after tests - Store FE_version and data lineage links	MLflow entries: model_id, version, stage Link to artifacts (S3/registry)	Serving / Auditability CI/CD pipelines
Model Cache (Redis)	Serialized serving artifacts (hot) (ONNX, native boosters)	- Keep warmed model for low-latency loads (LRU eviction) - Heartbeat checks; reload on version change	Fast model load (mem snapshot) cache_hit/miss metrics	API Serving (Layer7A)
Storage (S3)	Trained models, batch results, maps GeoTIFFs, uncertainty maps	- Versioned object storage; lifecycle rules (archive) - Serve as MLflow artifact store	Artifacts accessible for batch & registry Long-term archive, access logs	Audits / Reproducibility
Database (Postgres)	Predictions logs, metrics, alerts	- Store transaction logs, user queries, monitoring snapshots	Queryable logs: predictions_table, metrics	Reporting / Audits

		- Index for fast retrieval; retention & partitioning policies	alerts_table (drift, threshold hits)	
Monitoring (Grafana) & Alerting	Metrics (latency, accuracy, drift) traces (Tempo), logs	- Dashboards & alerting rules; integrate with Prometheus - Alert escalation (PagerDuty), runbooks link	Visual dashboards: latency p50/p95/p99, drift Alert tickets, on-call pages	SRE / ML Ops