Task 2 — AI-Driven IoT Concept: Smart Agriculture Simulation System

1) Sensors & Actuators (hardware)

Environmental sensors

- **Air temperature** (°C)
- **Relative humidity** (%)
- **Solar radiation / PAR (Photosynthetically Active Radiation)** or light sensor
- **Wind speed & direction** (for spray/evaporation modeling)
- **Rain gauge** / precipitation

Soil sensors

- **Soil moisture** (volumetric water content) — multiple depths (e.g., 10 cm, 30 cm)
- **Soil temperature**
- **Soil pH**
- **Soil electrical conductivity (EC)** — proxy for salinity/nutrients

Plant / imaging sensors

- **RGB camera** (for canopy cover, disease spotting)
- **Multispectral / NDVI camera** (if available) — useful for vigor & stress detection

Location & context

- **GPS** (for geotagging sensor nodes / plots)
- **Weather station / external forecast feed** (API)

Actuators

- **Irrigation valve / pump control**
- **Fertilizer dispenser / fertigation control**
- **Greenhouse vents / heaters / shade control** (if greenhouse)

Connectivity

- **Gateway (Raspberry Pi / microcontroller + LoRa / NB-IoT / WiFi)**

---

2) Proposed AI Model for Predicting Crop Yields

High level approach (hybrid, accurate, practical)

Use a **multi-modal model** that fuses:

- **Tabular & time-series data** (sensor streams, weather history, management actions)
- **Image features** (RGB / NDVI for canopy health)

Architecture (recommended):

1. **Feature extractors**
    - *Time-series branch*: 1D-CNN or Temporal Transformer / LSTM to extract temporal features from sensor sequences (soil moisture, temp, humidity, cumulative rain).
    - *Image branch*: lightweight CNN (MobileNetV2 / EfficientNet-lite) to extract visual crop-health features from periodic images.
2. **Feature fusion**
    - Concatenate time-series embeddings + image embeddings + static features (soil type, variety, planting date).
3. **Prediction head**
    - Dense layers → output: **continuous yield estimate** (kg/ha) or **yield class** (low/medium/high).
4. **Loss / metrics**
    - Regression: **MAE**, **RMSE**, **R^2**.
    - If classifying: **accuracy**, **F1-score**, **confusion matrix**.

Why hybrid?

- Tabular time-series captures environmental dynamics and management actions.
- Images capture canopy-level health and disease/stress that sensors may miss.
- Fusion improves robustness and accuracy.

Lightweight / Edge variant

- For edge inference, use quantized TFLite versions:

    ○ Image branch → MobileNetV2 quantized.

    ○ Time-series → small 1D-CNN or tiny Transformer quantized.

- Heavy training and periodic retraining happen in the cloud; inference for alerts runs at the edge.

Training details & labels

- **Inputs:** sliding windows of past N days (e.g., 30–90 days) of sensor readings + latest image(s) + static features.
- **Label:** measured yield at harvest per plot (kg/ha). Use aggregated historical harvest records.
- **Data augmentation:** add noise to sensors, image augmentation (flip, brightness) for robustness.
- **Cross-validation:** time-aware CV (train on earlier seasons, validate on later seasons) to avoid leakage.

---

3) Data Flow Diagram (text + ASCII diagram)

Narrative flow (short)

1. **Sensors** collect raw data at configured intervals.
2. **Edge gateway** (Raspberry Pi / microcontroller) performs local preprocessing (filtering, compression, simple aggregation) and local inference for immediate alerts.
3. Preprocessed data batches are **sent to cloud** (periodic/when connectivity available) for storage, heavy model training, and global analytics.
4. **Model training & versioning** in cloud; periodic model updates pushed back to edge.
5. **Dashboard** and **actuator control** (irrigation/fertilizer) operate from edge or cloud depending on latency/criticality.

ASCII Data Flow Diagram

```
[ Field Sensors ]                [On-device Camera]

  (soil moisture, temp,            (RGB / NDVI)

   pH, EC, humidity, rain)

       |                                 |

       |                                 |

       v                                 v

    ┌──────────────────────────────────────────────────┐

    |                  Edge Gateway                    |

    | (Raspberry Pi / Microcontroller + LoRa/WiFi)     |

    |  - Timestamp, geotag                               |

    |  - Local preprocessing (filter, smoothing)        |

    |  - Feature extraction (simple stats)              |

    |  - Local inference for immediate alerts           |

    |  - Buffering & secure transmission to cloud       |

    └──────────────────────────────────────────────────┘

                      |

             (LoRaWAN / NB-IoT / WiFi)

                      |

                      v

    ┌──────────────────────────────────────────────────┐

    |                      Cloud                       |

    |  - Ingest & store time-series & images           |

    |  - Advanced preprocessing & feature engineering |

    |  - Train & validate hybrid model (CNN + LSTM)    |

    |  - Model registry & version control              |

    |  - Historical analytics, dashboards, API         |
```

```
└──────────────────────────────────────────────────┘

       |                          |

(Model updates)             (User / Farmer)

       |                          |

       v                          v

┌────────────────────┐      ┌────────────────────┐

|   Edge Devices     |◄─────|   Web / Mobile UI   |

| (deployed model)  |      |  - Yield forecasts |

| - Deploy TFLite    |      |  - Alerts & reports|

| - Local inference |      └────────────────────┘

└────────────────────┘

       |

(Actuation commands)

       |

       v

┌────────────────────┐

|  Irrigation /      |

|  Fertigation Unit |

└────────────────────┘
```

---

4) Sampling rates, communication & storage suggestions

- **Sensor sampling:** soil moisture & temp — 15–60 min; weather sensors — 5–15 min; images — daily or twice weekly (higher frequency during critical growth stages).
- **Communication:** LoRaWAN for long-range low-power farms; WiFi for fields with coverage; NB-IoT where available.
- **Edge buffering:** store last N days locally to survive connectivity gaps.

- **Cloud storage:** time-series DB (InfluxDB / Timescale) + object storage for images.

---

5) Evaluation & KPIs

- **Model metrics:** MAE (kg/ha), RMSE, R^2. Target depends on crop — e.g., aim for MAE < 10% of mean yield.
- **Operational metrics:** latency for critical alerts (< 1 min on edge), data completeness (>95%), energy consumption of nodes, uptime.

---

6) Simulation considerations (since this is a simulation system)

- Use **synthetic data + historical datasets** to simulate seasons. Vary irrigation/fertilizer actions to test model sensitivity.
- Include **noise** and **missing data scenarios** to validate robustness.
- Provide a scenario runner that simulates multiple fields with different soil types and weather traces.