INNOMATICS
RESEARCH LABS

Theme 1 - Automated OMR Evaluation & Scoring System

Problem Statement

At **Innomatics Research Labs**, we regularly conduct **placement readiness assessments** across roles like **Data Analytics, and AI/ML** for Data Science with Generative AI course enrolled students. Each exam uses standardized OMR sheets with **100 questions**, distributed as **20 per subject across 5 subjects**.

Currently, the evaluation is **manual**: evaluators visually check each sheet, count correct answers per subject, and prepare student-level reports. With thousands of sheets (≈3000 on a single exam day), this process is:

- **Time-consuming** (delays in releasing results)
- **Error-prone** (human miscounts)
- **Resource-intensive** (requires multiple evaluators)

This manual dependency also **delays feedback loops**, which are critical for student learning and placement preparation.

Objective

The goal is to design and implement a **scalable, automated OMR evaluation system** that:

1. Accurately evaluates OMR sheets captured **via mobile phone camera**.
2. Provides **per-subject scores** (0–20 each) and a **total score** (0–100).
3. Works with **multiple sheet versions** (2–4 sets per exam).
4. Functions **online,** via a **web application interface** for evaluators to manage results.
5. Ensures **<0.5% error tolerance**, aligned with Innomatics' quality standards.
6. Reduces evaluation turnaround from **days to minutes**, freeing evaluators to focus on insights and student engagement.

Sample Data

Click here to download sample data.

Proposed Solution

We propose building an **Automated OMR Evaluation & Scoring System** with the following capabilities:

- **Capture:** OMR sheets can be captured via mobile phone camera.
- **Preprocessing**: Correct for rotation, skew, illumination, and perspective distortion.

- **Bubble Detection & Evaluation:** Identify marked bubbles using classical Computer Vision techniques (OpenCV), supported by ML-based classifiers for ambiguous cases.
- **Answer Key Matching:** Compare extracted answers with pre-defined answer keys (per exam version).
- **Result Generation:** Compute subject-wise scores, total marks, and store them in a structured format.
- **Web Application:** Provide an evaluator-friendly web interface for uploading sheets, monitoring progress, exporting results, and reviewing flagged sheets.
- **Storage & Audit Trail:** Store rectified sheet images, overlays (for transparency), and JSON results for audits.

Workflow

1. **Students fill OMR sheets** during exams.
2. **Sheets are digitized**: Captured individually via mobile.
3. **Evaluator uploads files** via the web application.
4. **System pipeline** executes:
    - Detects sheet orientation.
    - Rectifies perspective distortion.
    - Identifies bubble grid & extracts responses.
    - Classifies marked/unmarked bubbles.
    - Matches with answer key (based on sheet version).
    - Calculates section-wise and total scores.
5. **Results are stored** in a secure database and exportable as **CSV/Excel**.
6. **Evaluator dashboard** displays summaries (per student, per subject, aggregate stats).

Tech Stack (Core OMR Evaluation)

- **Python Programming** – Primary language for building OMR evaluation logic.
- **OpenCV** – Image preprocessing, fiducial detection, perspective correction, bubble extraction.
- **NumPy / SciPy** – Image array manipulation and thresholding calculations.
- **Scikit-learn / TensorFlow Lite (optional)** – Lightweight ML models for ambiguous mark classification.
- **PyMuPDF / PDFPlumber** – Handling scanned PDFs.
- **Pillow** – Image manipulation and format conversion.

Tech Stack (Web Application) - You can choose other tech stack for FE

- **Flask / FastAPI** – Backend APIs to process uploads, run OMR evaluation, and serve results.
- **Streamlit (MVP)** – Frontend for evaluators (upload, dashboard, review).
- **SQLite / PostgreSQL** – Database for storing results, metadata, and audit logs.

How to win a Pizza Party and Cash Prize?

Pizza Parties and a cash pool of 25,000 INR will be announced soon.

Check the last page on the main document 👉 Click here