**Detailed Project Description — *AutoClassPulse: Smart Daily Submission Automation System***

**🏫 Overview**

**AutoClassPulse** is a complete end-to-end automation system designed to ensure daily attendance or class data submissions by teachers are collected, verified, and followed up automatically — without manual intervention.  
It integrates **Google Forms**, **Google Sheets (with Apps Script)**, and **n8n** automation to manage daily submissions, reminders, and disciplinary follow-ups via **email and WhatsApp**.

**📝 Step 1 — Data Collection via Google Form**

Each teacher fills out a **Google Form** once a day to submit their class data (e.g., enrollment, attendance, or progress).  
The form includes:

* **Teacher Email**
* **Date**
* **Grade**
* **Teacher Name**
* **Phone Number**

When the teacher submits the form:

* The response is automatically recorded in the **“Form Responses 1”** sheet in your connected Google Sheet.

This ensures that all daily data entries are timestamped and associated with the correct teacher.

**📊 Step 2 — Data Management in Google Sheets**

The connected Google Sheet acts as the **central database** for all submissions.  
It contains **five key tabs**, each serving a specific purpose:

**1. Form Responses 1**

* Automatically populated by the Google Form.
* Columns:  
  Timestamp | Email | Date | Grade | Teacher Name | Phone Number
* This sheet is never edited manually. It acts as raw source data for all further processing.

**2. Teachers**

* Maintains the **master list** of all teachers who are expected to submit the form daily.
* Columns:  
  Grade | Teacher Name | Email | Phone Number
* This sheet provides the base reference to track who has or hasn’t submitted their data.

**3. Submission Tracker**

* The heart of the system — it tracks submission status **day-wise** and **teacher-wise**.
* Columns:  
  Date | Grade | Teacher Name | Submitted?
* **Automation:**  
  Every morning at **8:00 AM**, the script addDailyEnrollmentBlock() automatically adds a new block of rows for that day, including every teacher from the “Teachers” list.
  + Example: On Oct 19, 2025, all 10 teachers’ names are added automatically with a blank “Submitted?” status.
* A formula in the “Submitted?” column automatically updates based on the “Form Responses 1” data to show:
  + ✅ if the teacher has submitted today
  + ❌ if they haven’t yet
* This happens automatically — no manual dragging or refreshing is required.

**4. Pending Submissions**

* This sheet shows **only the teachers who have not submitted their data yet** for the current day.
* Columns:  
  Date | Grade | Teacher Name | Email | Phone Number | Last Updated
* Updated automatically **every 5 minutes** via the script updatePendingSubmissions().
* If a teacher submits later in the day, they are automatically **removed** from this list on the next update.
* At **11:00 PM**, the sheet is cleared daily by the clearPendingSubmissions() script, preparing it for the next day’s records.

**5. Excused List**

* (Optional) Contains names or emails of teachers who are excused from submission for a specific date (e.g., on leave or training).
* The updatePendingSubmissions() script automatically skips these teachers from the pending list.

**⚙️ Step 3 — Google Apps Script Automation**

Four main scripts control the automation within the Google Sheet:

**1️⃣ addDailyEnrollmentBlock()**

* Runs every morning at **8:00 AM**.
* Creates a new “Date block” in the **Submission Tracker** sheet for all teachers listed in the “Teachers” tab.
* Ensures that each teacher has one row per day for tracking.

**2️⃣ updatePendingSubmissions()**

* Runs every **5 minutes** throughout the day.
* Compares today’s date in the **Submission Tracker** with the **Form Responses 1** sheet.
* Identifies all teachers marked ❌ (not submitted yet).
* Populates the **Pending Submissions** sheet with their:
  + Date
  + Grade
  + Name
  + Email
  + Phone number
  + Timestamp of the last update
* Skips any teachers in the “Excused List”.

**3️⃣ clearPendingSubmissions()**

* Runs daily at **11:00 PM**.
* Clears all data in the **Pending Submissions** sheet to prepare for the next day.
* Ensures no outdated data remains that could cause duplicate or false notifications.

**4️⃣ checkDailySubmissionLimit()**

* Restricts teachers to **one form submission per day**.
* If a teacher tries to submit twice for the same date, their second submission is blocked or ignored.
* Prevents duplication in the “Form Responses 1” sheet.

**🤖 Step 4 — n8n Automation Workflows**

Two separate **n8n workflows** manage automated communication with teachers via **email** and **WhatsApp**.

**Workflow 1 — Morning Reminder Workflow**

**Purpose:**  
Send gentle reminders at **9:30 AM** to all teachers who haven’t yet submitted the day’s data.

**How it works:**

1. Triggered by a **Cron node** at 9:30 AM (Pakistan Standard Time) on **working days only (Mon–Fri)**  
   → 0 30 9 \* \* 1-5
2. Reads the **Pending Submissions** sheet via the Google Sheets node.
3. Sends:
   * 📧 **Emails** using the Gmail node.
   * 💬 **WhatsApp messages** using the WhatsApp Cloud API node.
4. Message content is personalized with teacher’s name and grade.

**Example Reminder Message:**

Dear [Teacher Name],  
Our records show that your class data for [Date] is still pending.  
Kindly submit your daily report before 10 PM today.  
— AutoClassPulse System

**Workflow 2 — Night Show-Cause Workflow**

**Purpose:**  
Send a **formal show-cause notice** to teachers who have not submitted even after the end of the day.

**How it works:**

1. Triggered by a **Cron node** at **10:00 PM (Pakistan Standard Time)** on working days only  
   → 0 0 22 \* \* 1-5
2. Reads the same **Pending Submissions** sheet.
3. Sends:
   * 📧 A formal email show-cause notice.
   * 💬 WhatsApp message version of the notice.
4. Example Show-Cause Message:

Dear [Teacher Name],  
Our system indicates you have not submitted the class data for [Date].  
Please provide a valid explanation for this delay.  
— Administration Office, AutoClassPulse

**Both workflows use:**

* **Google Sheets node** → to read the Pending Submissions data
* **Gmail node** → to send emails
* **WhatsApp Cloud API node** → to send WhatsApp reminders
* **Cron trigger node** → to schedule exact daily run times
* **Filter node** → to avoid duplicate sends or empty rows

**🧭 Daily Automation Flow Summary**

| **Time** | **Action** | **Script / Workflow** | **Description** |
| --- | --- | --- | --- |
| **8:00 AM** | Add daily block | addDailyEnrollmentBlock() | Adds rows for each teacher in the tracker |
| **Every 5 mins** | Update pending list | updatePendingSubmissions() | Refreshes unsubmitted teachers |
| **9:30 AM** | Send reminders | n8n Workflow 1 | Sends WhatsApp + email reminders |
| **10:00 PM** | Send show-cause notices | n8n Workflow 2 | Sends formal notices to unsubmitted teachers |
| **11:00 PM** | Clear pending list | clearPendingSubmissions() | Prepares for next day |
| **Anytime** | Block duplicate form submissions | checkDailySubmissionLimit() | Ensures only one form per teacher per day |

**🧩 Example Integration Diagram (Data Flow)**

Teacher → Google Form → Form Responses Sheet

↓

Submission Tracker ← Daily Block Script

↓

Pending Submissions ← Updated every 5 min

↓

n8n Workflows

↓

[Email + WhatsApp Notifications]

**🧠 Smart Design Benefits**

✅ **Fully hands-free operation** after initial setup  
✅ **No manual checking** required for late submissions  
✅ **Real-time updates** for pending data  
✅ **Automatic resets** for new days  
✅ **Two-level follow-up system** (reminder + show cause)  
✅ **Scalable** — works for any number of teachers or grades  
✅ **Secure** — each institution uses its own Google Sheets and API credentials