

# Task 2: Missed Class / Attendance Follow-Up Automation

## Context

Seamless Assist works with gyms and wellness studios.

One of our product ideas is to automatically detect when a member has missed classes and send them a **personalized outreach (SMS/email)**.

This task focuses on building a **prototype workflow** to simulate this feature using **mock data** and simple automation logic.

👉 No live CRM or messaging integration is required.

---

## Your Task

Implement a **prototype of Missed Class Automation** using mock data and simulate how follow-up messages would be sent.

---

## Requirements (Prototype Scope)

### 1. Mock Data Setup

Create a dataset of **10 members** in JSON or CSV format.

Each member should have the following fields:

- {
- "id": 1,
- "first\_name": "John",
- "last\_attended\_at": "2025-08-25",
- "membership\_status": "active", // active | frozen | canceled
- "is\_recurring": true,
- "phone": "+155555501",
- "email": "john@example.com"
- }

## 2. Inactivity Detection Logic

A member becomes **eligible for follow-up** if:

- `last_attended_at`  $\geq$  7 days but  $<$  35 days ago
- `membership_status` is not **frozen** or **canceled**

Additional rule:

- If `last_attended_at`  $\geq$  35 days AND `is_recurring = true`  $\rightarrow$  skip messaging (manual intervention needed).
- 

## 3. Message Simulation

Instead of sending real SMS/emails:

- **Log the message** to the console or write it into a file.

Example message format:

- To: John (SMS)
  - Body: "Hey John, we missed you last week! Book your next class here 📌 [Booking Link]"
  - Support **placeholders** like `{{first_name}}` and `{{last_attended_at}}`.
- 

## 4. Logging

Maintain a **mock messages log** (JSON/CSV) with:

- Who was messaged.
  - Who was skipped (and why).
-

## 5. Optional (Bonus)

Build a small dashboard (React or simple HTML) that shows:

- Members eligible for messaging.
  - Members skipped due to the 35+ days rule.
  - The messages log.
- 

## 6. AWS Integration

- Set up a **free AWS account** (if you don't have one).
  - Attempt to send the message using an AWS service (e.g., Amazon SNS for SMS or Amazon SES for email) instead of just logging it.
  - If unable to send real messages (due to sandbox or verification limits), document the steps taken and include screenshots or code showing your attempt.
- 

## Deliverables

Your project repo should contain:

1. **Mock dataset** (`members.json` or `members.csv`)
2. **Script/code** for inactivity detection + message simulation (and AWS integration)
3. **Log file** showing results
4. **(Optional)** Dashboard UI
5. **README file** with:
  - How to run the prototype
  - Any assumptions made

- Steps taken for AWS setup and message sending (with screenshots/code if real sending isn't possible)

---

## Evaluation Criteria

- **Correctness:** Logic for eligibility detection and skipping is implemented properly.
- **Code Quality:** Clean, modular, and easy to follow.
- **Logging & Reporting:** Clear logs of who was messaged vs skipped.
- **AWS Integration:** Attempt to use AWS SNS or SES for actual message sending.
- **Creativity (Bonus):** If you add the optional dashboard, design and usability will be noted.

---

## Submission Guidelines

- Share your project repo (GitHub or zip file).
- Include the README.
- Deadline: **48 hours** from receiving this task.