Daily QA Learning Report

Date: 2026-01-07
Tester: Nabaraj Thapa
Project: School Management System (SMS)
Session Type: API Testing & Postman Learning

Objectives of the Session
1. Understand HTTP methods and their practical usage in SMS.
2. Learn how to test SMS APIs using Postman.
3. Identify the differences between UI-based testing and API testing.
4. Understand common API errors and how to debug them.
5. Learn the correct structure for sending requests to the backend (JSON vs form-data).

Key Learnings
1. HTTP Methods in SMS:
   - GET: Retrieve data (e.g., student list, subject details).
   - POST: Add or update data (most commonly used).
   - PUT/PATCH: Update data (mostly not supported in SMS).
   - DELETE: Remove data (restricted in SMS).

2. Backend Limitations:
   - Backend often expects POST for updates instead of PUT.
   - Optional fields may cause backend crashes if not sent correctly.

3. Postman Usage:
   - Able to test APIs outside of the UI.
   - Learned how to configure headers, authentication, and request body.

4. JSON and Form-Data:
   - Raw JSON works for most text data but cannot be used for file uploads.
   - File fields (e.g., profile_picture, birth_cit_certificate) must be sent as arrays in multipart/form-data, even if empty.

5. Error Handling Insights:
   - 422 Validation Error: Missing required fields, invalid data types, or incorrect IDs.
   - 403 Forbidden/Error: Backend crashes when expected array keys are missing or null.
   - 405 Method Not Allowed: Occurs when using unsupported HTTP methods.

6. UI vs Postman Behavior:
   - UI automatically sends empty file arrays; Postman must replicate this exactly.
   - Understanding backend expectations prevents repeated 403 errors.

7. Backend Bug Awareness:
   - Learned to identify backend design flaws such as undefined array key or array offset on null.
   - Documented proper bug reporting steps (expected vs actual behavior, severity).

8. Debugging & Validation:
   - Learned to inspect network requests in the browser to replicate correct request structure in Postman.
   - Developed skills to map UI actions to API requests accurately.

Session Outcome

- Successfully understood API request structures for adding/updating students in SMS.
- Identified correct use of Postman form-data for optional file uploads.
- Gained clear understanding of common HTTP errors and backend behavior.
- Prepared to create proper Postman test cases for SMS modules.

Next Steps

1. Create Postman collection for all SMS API endpoints.
2. Practice updating student and subject records using correct form-data.
3. Document common errors and workarounds for future QA reference.
4. Explore automation testing for repetitive API validations.