AI Class Assistant - Day 2 Task Breakdown

Team Members

- Member A: Backend API Developer

- Member B: Tester & Validator using Postman

Member A - Task: Build Flask Backend Upload API

1. Install Flask: pip install Flask

2. Create a new file `app.py` and add file upload logic.

3. Accept lecture files: PDF, DOCX, PPTX, MP3, WAV.

4. Save files to 'uploads/' folder.

5. Add validation: file type and max size 10 MB.

6. Save metadata (filename, timestamp) to `upload_log.json`.

7. Run Flask: python app.py (server at http://localhost:5000/upload)

Member B - Task: Test Upload API Using Postman

1. Install and open Postman.

2. Create POST request to http://localhost:5000/upload.

3. Under Body > form-data, add key: file (type: File), choose any valid file.

4. Send request and confirm upload success.

5. Check uploads/ folder and `upload_log.json` for metadata.

6. Test edge cases:

   - Invalid file types (e.g., .exe) -> should return error.

   - Files > 10MB -> should return error.

   - Missing file -> should return error.

7. Coordinate with Member A if any issue arises.

Deliverables

- Working Flask upload API

- File validation (type + size)

- Files saved in 'uploads/' folder

- Metadata stored in 'upload_log.json'

- Tested thoroughly using Postman