

## AI Class Assistant - Task 5

---

### Task 5 Title: Flashcard Generation + Progress Tracking Integration

---

#### Person A: Flashcard Generator Integration

##### Objectives:

- Generate flashcards from uploaded or summarized content.
- Display flashcards in the frontend.
- Store flashcards in the backend.

##### Steps:

##### 1. Backend (Flask/FastAPI):

- Create a new endpoint: `/generate_flashcards`
- Input: Summarized or raw lecture text.
- Process: Use an LLM to create a list of Q&A flashcards.
- Output: JSON list like:

```
[
  {"question": "What is ML?", "answer": "Machine Learning is..."},
  {"question": "Define supervised learning", "answer": "Supervised learning is..."}
]
```

- Save the flashcards in MongoDB (or preferred database).

##### 2. Frontend (React):

- Create a new React component `<FlashcardViewer />`
- Use card-flip animation to show Q (front) and A (back).
- Fetch flashcards from backend via API.

##### 3. Postman Testing:

- Send POST request to `/generate_flashcards` with example content.
  - Ensure API responds with well-formed JSON.
  - Check stored data in DB.
-

## Person B: Progress Tracking System

### Objectives:

- Track user activity: summaries done, quizzes taken, flashcards reviewed.
- Display progress in the frontend as charts or bars.

### Steps:

#### 1. Backend:

- Add new endpoint: `/progress`
- Input/Output schema:

```
{
  "user_id": "abc123",
  "summaries_done": 4,
  "quizzes_taken": 2,
  "flashcards_reviewed": 10
}
```

- Update values when user completes tasks.
- Store in database.

#### 2. Frontend (React):

- Create `<ProgressDashboard />` component.
- Use a library like `recharts` or `chart.js` to show:
  - % flashcards reviewed
  - Number of quizzes/summaries
- Automatically update UI on user actions.

#### 3. Postman Testing:

- Test POST and GET on `/progress`
- Verify data updates correctly in DB and UI.

---

Let me know if you need code templates or diagrams for this task.