BRD (Business Required Document)

Application Title: Survey Report Analyzer For Course Instructors

User will logged into the website, we will give a unique id,

Phase-1

User Uploads the CSV,

send to s3 (raw data).. the file needs to save with the user unique id.append(the count of this file).. I think we can use hash map(key,value) to append count to it

phase-2

Then pre processing using glue, that csv file uses some sort of ETL technique (yet to be decided),

What columns it have,

That needs to send another ai model, that can give what insights can we get from those columns,

If it have column like feedback or comment section we can perform sentiment analysis, to segregate positive, negative neutral

spark pre processing, will upload again in s3 (refined\_data folder) will be saved with user\_Id appended with count can hash map(key, value) be used

what else can we do::

display dashboard may be we can use Quick sight (not yet planned what dashboards may require)

phase 3

That csv files in refined folder, may trigger lambda function that can trigger bedrock and quick sight to display some useful dash boards, and actionable insights

Pdf can be downloaded

Actionable output be like (my assumption ): 70% of students had negative sentiment on the pace of React module — consider slowing down or splitting it into two sessions.

**Keyword and Theme Extraction**

**What to Analyze:**

* Frequently occurring phrases in feedback
* Common suggestions (e.g., "more projects", "too fast", "recordings")

**Why It Helps:**

* Highlights repeating pain points or requests

**Actionable Output:**

"‘More hands-on projects’ appeared in 400+ comments — include 2 more capstone projects."

**What to Analyze:**

* Ask students to rate “difficulty” and “understanding” per topic (if in your CSV)
* Compare: which topics are marked “difficult” and “poorly understood”

**Why It Helps:**

* Pinpoints where teaching style/resources can improve

**Actionable Output:**

"Git and GitHub setup rated as ‘difficult’ and ‘unclear’ — add live demo and troubleshooting guide."

**What to Analyze:**

* Comments with keywords: "too fast", "slow", "overwhelming", "rushed"
* Average feedback sentiment linked to course speed

**Why It Helps:**

* Helps you balance course speed for better engagement

**Actionable Output:**

"React and Spring modules had most 'too fast' comments — revise timing to allocate more practice time."

* All responses in suggestion fields or open-ended comments

**Why It Helps:**

* Directly tells you what students want more or less of

**Actionable Output:**

"80 students asked for a project-based mini-hackathon — consider adding one at the end of course."