Zapier - Data to Google Sheets

As part of this group task, you are expected to deploy a custom service in Hasura that triggers a Zapier Zap. The project should also implement the following use-cases. Please note that Zapier has a 7 day free trial that can be extended on request. If there is no extension provided, then different members of the team should create accounts to replicate the Zap.

Deliverables

The following are the deliverables for this group (*1 per group*):

1. **Plan of action doc:** A 1-3 page document with an outline of the group's plan for delivering on **Submission-1** (*see next point*). This document should have a list of sub-tasks along with their details and a list of resources/documentation that will be referred to during the implementation. Basic wire-frames for the frontend can also be included in this document.

2. **Submission-1:** For this submission, you will have a skeletal integration with the API assigned to you and implement the tasks mentioned as follows:

    a) **Backend features/tasks**
        i) Create a custom service that adds data to a table and triggers a Zapier Zap's webhook that adds the name of the table to a Google Spreadsheet as a new row.
        Zapier docs: https://zapier.com/help/creating-zap/
        Please note that you will need to use Webhooks as a triggers for your Zapier Zap (Webhooks-> Catch Hook). This will create a POST webhook that you can call with JSON data from your custom service.

        ii) Support the frontend use-cases documented below.

    b) **Frontend features/tasks**
        i) A UI for users to select a table and enter values to insert into the database
        ii) User should be notified or given some feedback that data has been saved and that the zap has been triggered.

The goal of **Submission-1** is to get your group's work published at https://hasura.io/hub. To achieve this, you will be expected to first submit your code and a list of deliverables (to be shared before the deadline) in a GitHub repo. This submission will be reviewed by your mentor and, after incorporating the feedback, you will then be expected to publish the final draft.

3. **Submission-2:** This is where your group gets to be creative - you are expected to plan and implement a set of features (or build a small MVP app) that extends the integration that you have built as part of **Submission-1.** The emphasis here is on iterative development and not about showing off a large feature set. Plan conservatively and focus on creative uses of the API instead of a trying to implement a lot of features.

Hasura Feature Checklist

Use-Case	Hasura Feature to be used
Data Storage	Postgres instance in the cluster. You can import data using a CSV file or use the Data APIs
Inserting or fetching data from DB	Hasura Data API
User Authentication (login, signup, role-assignment, OTP)	Hasura Auth
Storing static files like images, documents, etc.	Hasura File APIs
Custom Code, cron jobs	Custom microservice

Other information

a) Each team should work on 1 Hasura cluster as collaborators (to be used for team presentations and submissions).
b) Please use the Hasura documentation for information on using the aforementioned features. Please report any errors or missing information in the documentation.

More details around the exact deliverables, mode of submission and deadlines (based on exam schedules) will be shared shortly over email and Slack.

Good Luck and have fun building apps!

-Team Hasura