INFO20003 Semester 2, 2024

Assignment 1: ER Modelling

Due: Week 5 - Friday 23rd August 2024, 5:59pm Melbourne Time.

OGRE: Open Generative-AI Responsible Environment

With the recent advancements in generative AI (GenAI) -- such as generation of text and code (ChatGPT, Gemini, and Bard); images (Stable Diffusion, Midjourney, and DALLE); music (Suno); and video (Sora) -- many potential issues have been raised, including copyright problems, fake news, and low-quality content. Also, the fact that one must subscribe to so many different platforms is making things difficult for the creative industry, as the costs are not cheap!

You and your classmates decide to launch a startup company that offers a unified platform for people to use such generative technologies responsibly and cost-effectively, while allowing for ethical auditing and legal compliance. Your startup is known as OGRE: The Open Generative-AI Responsible Environment. OGRE has successful partnerships with all the major tech companies, which is good news!

Your classmates/business partners reckon that you have *the best Database skills* amongst all of them, so they selected you to oversee the database infrastructure for OGRE. 😊
(The other business partners will take care of the AI and infrastructure side!)

As part of this, you will be creating a MySQL database to store this information. The following specifications have been provided to you to assist in your design.

Platform and Session

A platform is a specific GenAI offering or model by a vendor, such as "*Stable Diffusion 3 Medium*". For most platforms, users create 'sessions' to allocate resources so they can issue prompts. To cope with state-of-the-art platforms which might be risky (such as OpenAI's Sora video generator), users might be required to lodge applications for sessions in order for OGRE staff to verify that these models are not used for e.g., spam or misinformation: OGRE staff manually approve these requests before sessions can be made (more later).

Your will be creating the technical infrastructure for the platforms, sessions, and users.

For each platform, the system records its details, that are: name (e.g., as discussed, "*Stable Diffusion 3 Medium*"), a single vendor (e.g., *StabilityAI*), and technology partner (e.g., *nVidia Inc*). Then, for each platform, we have different sessions where users can create and work on.

A session consists of a non-unique version string internal to the platform (e.g., '*Model version 1023'*, or '*20240101beta1' - note that other sessions can have similar version strings*); cost accumulated in AUD (in terms of both hours, to one decimal point, such as 4.5 hours; and cost per hour e.g., $1.00); description (e.g., "*Gilbert's session for prototype design work*"); region (e.g., *Australia*); the session's starting and ending date and time; and a technical note of how it is used (e.g., "*Python app*" or "*Appstore at http://play.google.com/*" or "*http://colab.google.com/*"). The identifier, region, and dates/times can obviously differ between sessions. A session also has a special flag indicating if it requires an application to be made before it can be used (more later). Of course, each session is linked to one user; a user can have many sessions as they can afford!

Each session has at least one runtime, where each runtime is a particular configuration of virtual machine hardware to enable GenAI models to be run. Each runtime is associated with the following information: a processor type (e.g., *GoogleTPU*, *Ryzen CPU*, *Nvidia RTX4070*, *Snapdragon X Elite*, etc.), description (as a text description of maximum 500 characters), parameters (text up to 500 characters, such as "*--cpu-limit=50*") and maximum prompt count (e.g., 40 prompts), to comply with capacity restrictions. Each runtime is associated with exactly one session. The information of a runtime, therefore, can be different for different sessions and platforms.

Prompts

Each session has at least one prompt. Each prompt has a date/timestamp; prompt text (e.g., "*Draw me a picture of a happy cat listening to Rick Astley*"), parameters (e.g., "*system:You are an artist AI*"); attributes indicating if the prompt contains potentially risky material (any combination of the following: spam, fake news, restricted content, illegal content, with potential to add other attributes in the future such as 'election interference' as the need arises); result (which might range from simple text to an entire MP4 file).). Note that Prompt parameters and Runtime parameters are different from each other.

Also, to comply with ethical and legal requirements, each prompt has zero or more administrative reviews by Administrators. Each review must be date/timestamped, with an administrative comment (e.g., an administrator, in a single admin review, might comment that the prompt has been "Sent to legal team").

Users

For each user, the system records their details such as first name, last name, a unique email address, a unique phone number, a unique payment token (in UUID format which looks like "cadc6b45-c296-4cb9-9470-b30aaf4a3cb3"), a payment method (one of Google Pay, PayPal, Apple Pay, or WeChat Pay), the names of any affiliations of the user (e.g., University of Melbourne, Atlassian etc), and total number of warnings. A user can be affiliated with any number of affiliations. As for warnings, this is defined as the number of times a user has been

warned for violations such as creating spam; for most users, this will be zero; but this goes up to a maximum of three before they are terminated.

A unique feature of the system is that it keeps track of the connections between users. This allows a community of responsible GenAI users. Each user can have up to 50 connections with other users. For each fan connection, the system stores the start date of the connection and a description (e.g., "fellow students of INFO20003 art club"). If a connection is ended by one of the users, it should no longer be recorded in the system.

Applications

For most sessions, users can simply write prompts to generate results as per other GenAI systems, as long as they have the money needed. As mentioned earlier, for restricted platforms, users need to apply for Applications instead. In this case, a platform can have any number of applications: each application contains a free-form text field (e.g., a short formal letter requesting access and justifying their business case).

A user can have at most one application request, as an absolute maximum, across all platforms. (e.g., if OpenAI has both Sora and GPT10, a user can apply once in total, so they must choose between the two). All applications are tracked by the system for auditing purposes, no matter their status. Of course, the user can withdraw their application, or have the application manually approved, to reset the quota back to zero.

Each application must be processed by an Administrator. If an application is manually approved, we need to record which Administrator approved it. Once an application is approved, the User then requests for a session (one per application), with the approval attached to the session. Obviously, not all applications are accepted; and not all users might follow through with their approved application e.g., due to costs.

Administrators

OGRE employs a team of Administrators, who are also Users in their own right. As an auditing and compliance requirement, each Administrator has a unique, persistent staff ID (numeric), first name, last name, and address (e.g., 'Melbourne HQ, Swanston St'). To ensure Administrators are neutral and don't process/act on their own accounts and that of their friends, all Administrators are required to declare their user account.

Your team at OGRE and their subcontractors hire Administrators and assign them to platforms. This is because some Administrators specialise in specific types (e.g., some legally trained Administrators can better respond to issues such as copyrighted images). We record the start and end date of employment for each platform, and a flag indicating if the Administrator has done all the training required. A staff member can work across multiple platforms and are able to renew/cancel their assignments as OGRE supports flexible working and a work-life balance.

Finally, recall that Administrators need to be able to attach reviews on prompts as required under Prompts. Recall that "each prompt has zero or more administrative reviews by Administrators".

Business Requirements

Your database design needs to be able to meet the business's needs to answer the following questions. You do **NOT** need to write SQL queries in your assignment to answer these questions.

1. How many Administrators have been assigned to 2 or more different *platforms* (e.g., Stable Diffusion XL and Google Gemini)?

2. How many times has the user named Pfake Niews been cautioned for violating policies?

3. Out of all the platforms that use an application system, which one has the highest rate of applications to sessions?

4. Which users with connections to Farzaneh did not use any prompt in 2024?

5. (For auditing purposes) Which unethical Administrators have reviewed prompts by users that their own user accounts are connected to?

6. How many Administrators who are also users have "Github" as an affiliation?

7. How many sessions with a total cost of $0.00 (free) did the users Lucy, Colton, Firman, and Angela use in total?

8. For sessions powered by any Snapdragon processor runtimes, which ones have more than 200 prompts in use?

9. (For auditing purposes) What is the result of the prompt(s) created on *Facebook LLAMA* on (22 May 2024, 10:15 to 10:20 am) for sessions with version numbers 12.1beta and 12.2ReleaseCandidate?

10. (For legal purposes) Which users created prompts that have at least 5 reviews by Administrators where the prompt included the words 'breaking news', and whose user accounts already have 2 violation strikes?

Instructions

Assignment 1 is worth 10% of your final mark. The assignment will be graded out of 100 marks as described in the table below:

Component	Grading
ER Physical Model with assumptions	80% (8.0 raw final marks)
Conceptual Model in Chen's Notation	20% (2.0 raw final marks)
.nwb Physical Model File submitted	ASSIGNMENT HURDLE

You are to analyse this business case and design a **Conceptual ER Model** in **Chen's notation (MUST be hand drawn, either pen/paper or stylus/tablet/iPad[1])** as taught in class and a **Physical ER Model** for a MySQL Relational Database in **Crow's foot notation (modelled with MySQL Workbench).**

You may list any assumptions you have made about the model. There is a 200-word limit for assumptions. Assumptions must not be used to simplify the assignment, but only to justify your decision about any ambiguity in the study.

Assignment Submission

Please pay special attention to the penalties listed [⚠].

You are to submit the assignment under the Assignments tab on Canvas LMS. The submission will require you to **submit two (2) files:**

1) **A SINGLE PDF document** containing:
   a) a legible (hand drawn) picture of a conceptual model in Chen's notation, scanned/photographed/digitised.
   b) a screenshot/export of your Physical ER Model done in MySQL Workbench. Ensure that tables are fully expanded so all attributes are readable. Please ensure the image is not blurry.
   c) any assumptions you made (limit of 200 words).

   Note: All these components must be within a **SINGLE** PDF document. You can use an online tool like https://smallpdf.com/merge-pdf to merge multiple PDFs together into a single PDF.

2) **A copy of your final .mwb MySQL Workbench file of your Physical ER model.**
   Note: This model will only be used by staff in circumstances where the screenshot/export of your model in the PDF is unreadable (**i.e., if we have to open this file, a penalty of 10% will occur** ⚠), so remember to include a clear screenshot/export of your physical model in the PDF! If the screenshot/export of your physical model is unreadable and the .mwb file was not submitted, you will receive zero marks for your physical model ⚠.

Remember, if you fail to submit clear and legible models your assignment will be penalised ⚠.
If you submit non-hand-drawn models for Submission (1a) your assignment will be penalised ⚠.

Please make sure that you **submit** your files on Canvas ⚠. After uploading the files, you need to press **'Submit Assignment'** to submit the files. If you submit late because you failed to press the submit button and only noticed this after the deadline, your submission will be considered late just like any other late submission to maintain fairness for all students.

Unless you have an approved extension (see below), **you will be penalised -10% of the maximum number of marks in the assignment per calendar day that your submission is late** ⚠. For instance, if you received a 78% raw score, but submitted 2 days late, you'd receive a 58% for the assignment.

[1] ***EXAM TIP: Note that your exam might require you to hand-draw diagrams****; as such this is to encourage you to get practicing and not over-depend on the computer!*