

ISYS2120 Assignment 1 (Sem2, 2025)

Due: Sunday 31 August, 11:59pm Sydney time

Value: 5%

Late work policy: (from unit outline) Late submissions for assignments will incur a penalty of 5% of the maximum awardable marks for each day, or part-day, past the due date, up to a maximum of 7 days (as after this time, feedback on on-time submissions will be available, resulting in an unfair advantage if submissions after this time were accepted). After 7 days late submissions will not be accepted. Where special consideration is granted for these assessments, extensions of a maximum of 7 days will be permitted. After 7 days, reweighting of other relevant tasks will be applied.

Summary (but details further down are authoritative)

Done individually. You are allowed to use GenerativeAI, but if so, you must submit an appendix showing each prompt you used, and the corresponding output from the computer; you will be marked only for what you have contributed, not on what was done by the computer.

Provided for you: an Entity-Relationship diagram for a domain, along with some text about the domain.

You produce: (A) a relational schema that is good for storing the information about a state of the domain; it should allow any reasonable information that fits the ER diagram, and also it should enforce constraints that are shown in the ER diagram, or implied by general knowledge. (B) executable SQL CREATE TABLE statements implementing the schema in PostgreSQL, enforcing relevant constraints from the ER model and sensible domain knowledge.

Submit: a report with structure as described in detail below [submit on Canvas], and a text file with SQL statements to create tables [submit on Canvas]

The task

The conceptual model below is for a system used by a music festival operations team to manage information about venues, the stages inside those venues, and the festival goers (some of whom hold tickets, some of whom represent sponsors). The system also records the staff who operate the event and the performance slots that take place on stages.

A festival goer may be a ticket holder (with a credit balance for purchases and entry) **or** a sponsor representative (linked to an organisation providing financial support), or both. This is shown with an IsA relationship that is *overlapping and partial*: a festival goer may be neither, one, or both of these subtypes.

Each venue consists of **multiple** stages, and the relationship locatedIn records which stage is in which venue, as well as the zone of the venue where that stage is situated (e.g., "North Field", "Hall A Balcony").

A stage hosts one or more performance slots. Each slot has a unique identifier and records its start time, end time, and agreed performance fee. Festival goers may apply for a performance slot; this application records the proposed fee and requested duration. If accepted, and the slot is assigned, the ticket holder is linked to the slot through the relationship performsAt, which records the final ticket price for the performance.

The system also manages sponsorships: each sponsor representative is connected to the stages they fund through the Sponsors relationship, which records the start date, end date, and the sponsorship amount.

Festival staff are responsible for running the event. Each staff member has contact details and working days recorded. Staff members may manage a stage and may also assist another staff member (for example, acting as a support or deputy). The system also records when a staff member has been in contact with a festival goer, including the most recent date of contact.

The task is to produce a relational schema that corresponds to the ER conceptual data model we provide. We expect you to first identify tables and their attributes, along with primary key and foreign key constraints; this will be expressed using a relational schema diagram. Then you must give SQL CREATE TABLE statements that would be executed to define the schema in a DBMS. Some aspects of the conceptual model are not explicitly specified (for example, it does not indicate the data type for attributes); you need to make sensible choices for these, so that the SQL statements can be executed by the PostgreSQL system used by Edstem for its SQL code windows.

It is important that your proposed schema should try to capture the integrity constraints from the conceptual model, but even more crucial that your schema does not unduly restrict the data -- for example, if the ER diagram allows some situation, your schema must not prevent data showing that situation from being stored (unless the restriction is anyway implied by general knowledge, for example, it will be necessary to restrict what is stored for an attribute so that it fits into the data type on the column). As an example, if the conceptual knowledge allows different entities to have the same value in some attribute, the SQL tables should also allow this; or if the conceptual model allows X to be related to multiple instances of some other entity, then the relational model should not limit X to only one relationship.

In some cases, the conceptual model may allow a situation but you feel that this should be prevented (based on your knowledge of the domain or the way the world works); if so, you can enforce the restriction in your SQL CREATE TABLE statements, and include a comment on the statement that explains why you have added this restriction. Similarly, if there is a restriction in the conceptual model but you can't see a good way to enforce that in the SQL schema, you should say so in a comment on the corresponding CREATE TABLE statement.

Time management

Before the week 3 lab, you should produce a draft relational schema (either in text or as a relational schema diagram) that deals with at least the entity types Stage, TicketHolder, and the relationship types sponsors and appliesFor. Knowledge of how to understand the diagram is covered in the week 2 lecture slides, and week 2 lab, and the knowledge about how to produce a relational design from a diagram, is covered in week 3 lecture and week3 lab.

By the week 4 lab, you are expected to have fixed any issues with your draft schema that were found in week 3, and produced CREATE TABLE statements that can be executed to produce the tables in your schema so far. The commands

should be valid in the precise syntax that is supported on the Edstem SQL system (use the Edstem playground to check that it works).

Submission

You must produce *a text file* (whose name ends in .sql) with the CREATE TABLE statements that can be executed to produce the tables needed to capture the domain information (including all integrity constraints). The commands should be valid in the precise syntax that is supported on the Edstem SQL system (use the Edstem playground to check that it works).

You must also produce *a report* with the following two-part structure (plus a possible appendix, if you used GenerativeAI). The report can be produced with whatever text formatting approach you choose, but what is uploaded must be in **PDF**. In order to make the marks management easier, the cover page should match the template we provide (as a .docx file, and as pdf); you can use the .docx file directly (eg if you choose to work in Word) or you can have another formatting and include or copy our pdf cover.

Part A

A relational schema, presented as a relational schema diagram.

Part B

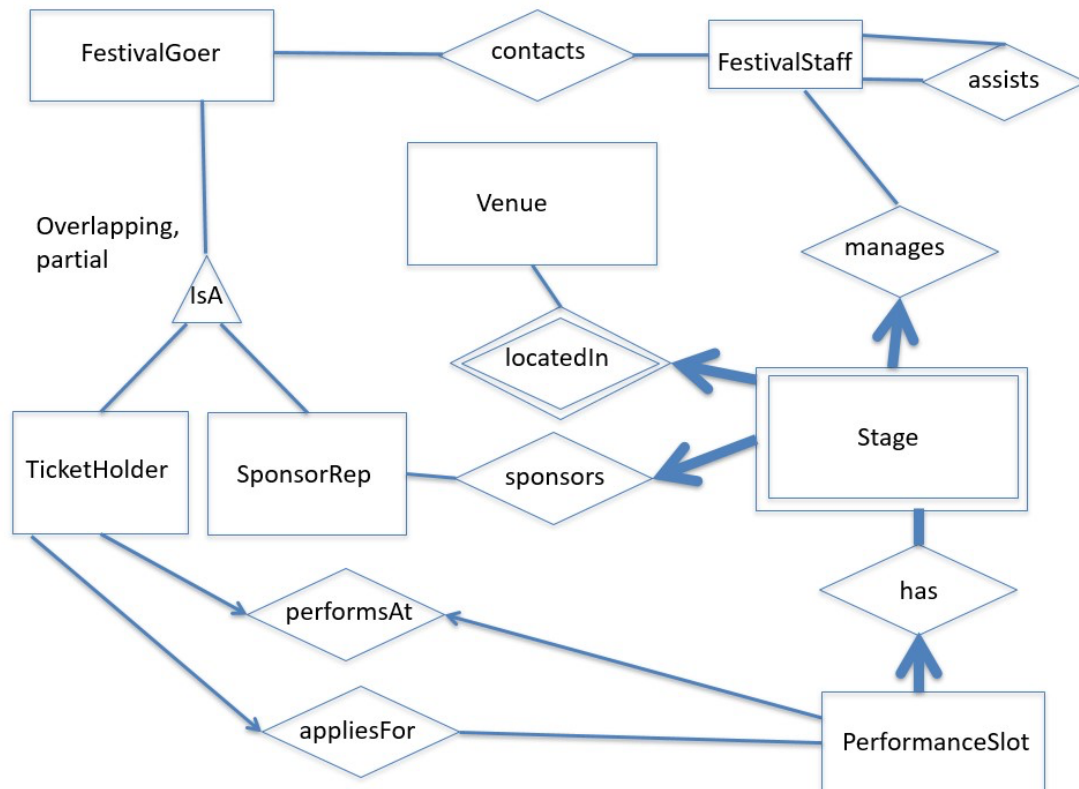
An exact copy of the SQL CREATE TABLE statements from the file. [It is essential that you include the SQL here in the report, as well as in the text file]

[Possible]Appendix

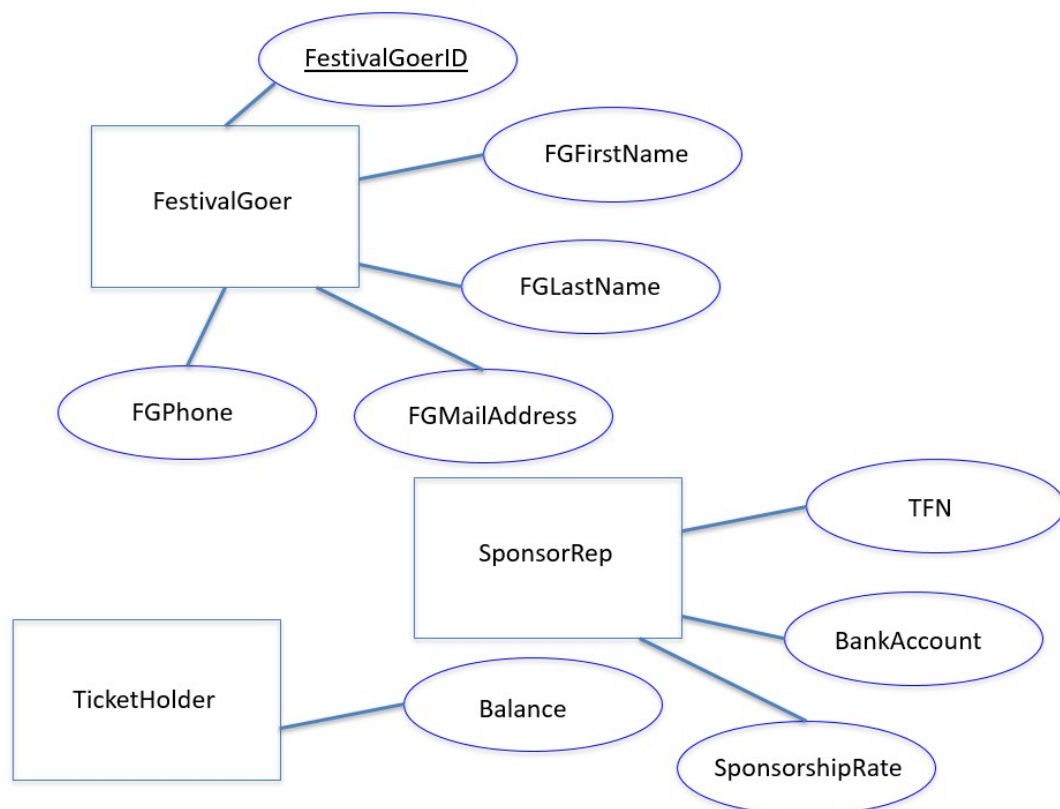
If you used GenerativeAI in your work, you need to submit here a transcript showing the prompts you provided, and the output from the system.

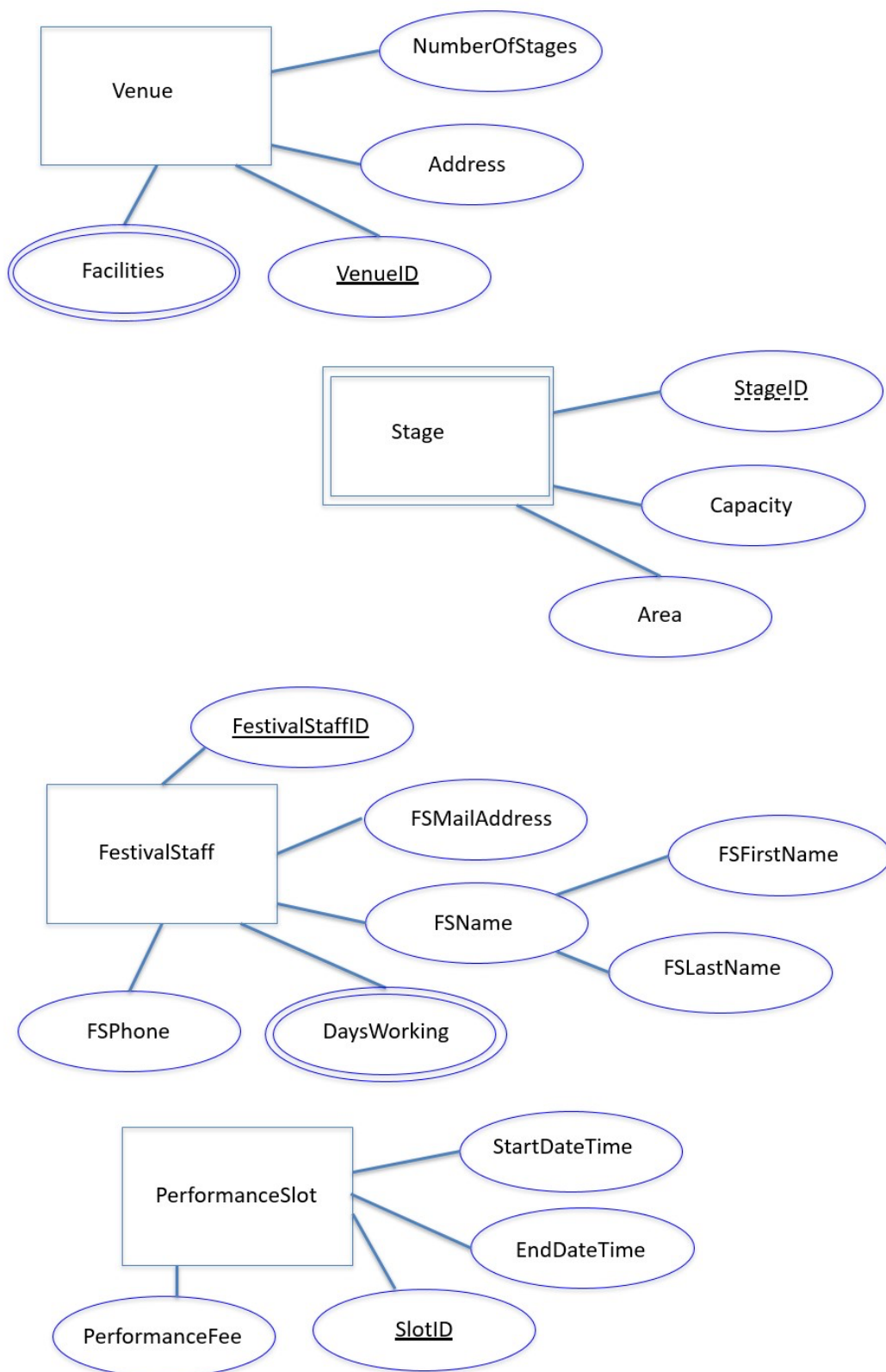
The ER diagram and textual description, for the domain

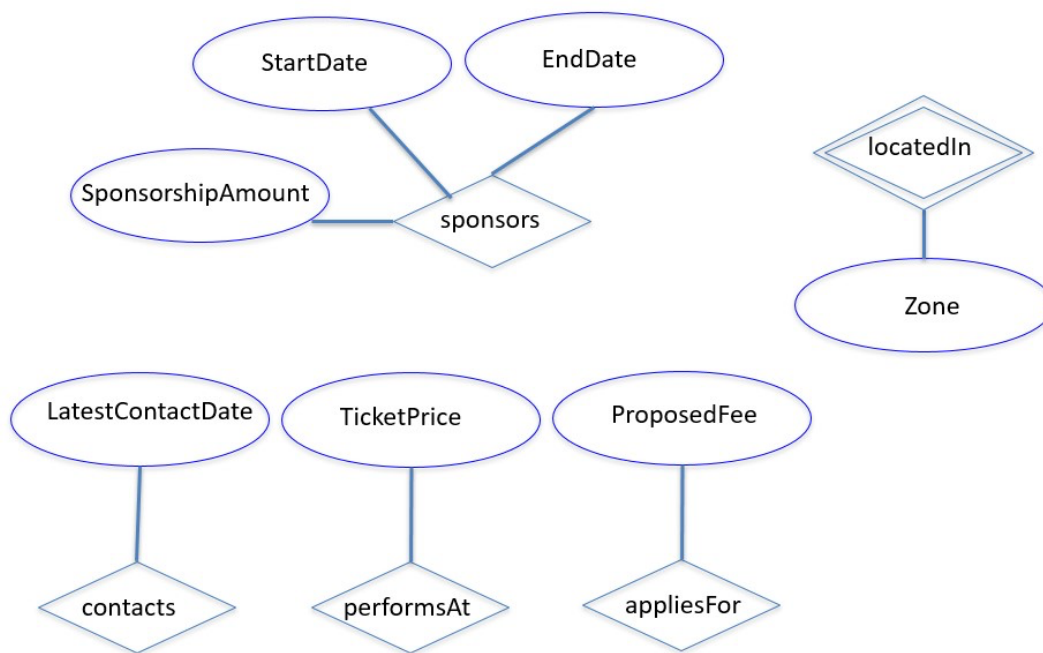
The conceptual model is expressed in an ER diagram, but due to its complexity, we show it in pieces. First is a high-level diagram, showing the entities and relationships, but not the attributes.



Next, we give diagrams showing individual entity or relationship shapes, each with its attributes. If we don't show the relationship here, that means there are no attributes.







How to submit your work

There are two submission links in Canvas. One link is for the SQL text file, and one is for the PDF report. In each case, the file should be uploaded by the due date (except under simple extension, special consideration, or academic plan). You can resubmit at any time up to the due date; we will mark the last submission before the due date. Once the work is marked, *the mark will be shown for the report* (the sql code submission does not get a mark shown in Canvas; it is there in case we need to check syntax in PostgreSQL etc).

Marking

Marks will be posted in Canvas once marking is finished for the class.

There are 4 points in total for the quality of your relational schema as shown in the report. This aspect is made up of 2 points for the way you represent the information about the Entity Sets, and 2 points for how well you represent the information about the Relationship Sets. The marking here will consider whether your schema can capture any reasonable valid state of the domain, and whether it will enforce constraints shown in the ER diagram; sensible variations between the diagram and schema will be permitted only if these are reasonably explained in the report. Note that if you used GenerativeAI in designing your schema, the marking will be based only on what you have contributed to the schema (you will not get marks for the AI's skill in capturing information sensibly!)

There is 1 point for showing you know the correct notations and syntax, for the relational schema diagram and SQL DDL statements. The marking here will consider the use of each approach, as well as whether they match with one another. Note that if you used GenerativeAI in producing diagram or SQL, the marking will be based only on what you have contributed to these (you will not get marks for the AI's skill in using notation correctly!)

Special cases: if the report does not contain a copy of the SQL DDL statements, the most you can get is 2.5 out of 5. If the schema in the schema diagram contradicts the schema declared in SQL, the most you can get is 2 out of 5.

You are allowed to use GenerativeAI, but if so, the report must have an appendix showing each prompt you used, and the corresponding output from the computer; you will be marked only for what you have contributed, not on what was done by the computer.