CS2102: Database System

Assignment 02: Schema AY2023/24 Sem 2: Week 05

Objective

The objective of this team project is for you to apply what you have learned in class to design and develop a database application using PostgreSQL. The project is to be done in teams of four students. The project consists of the following four tasks:

(A01) Design an ER data model for the application. Your ER model should capture as many of the application's requirements as possible.

(A02) Translate your ER data model into a relational database schema. Your relational schema should capture as many of the application's constraints as possible.

(P01) Implement an SQL Query for each of the functionalities listed in Application Functionalities.

Current Status

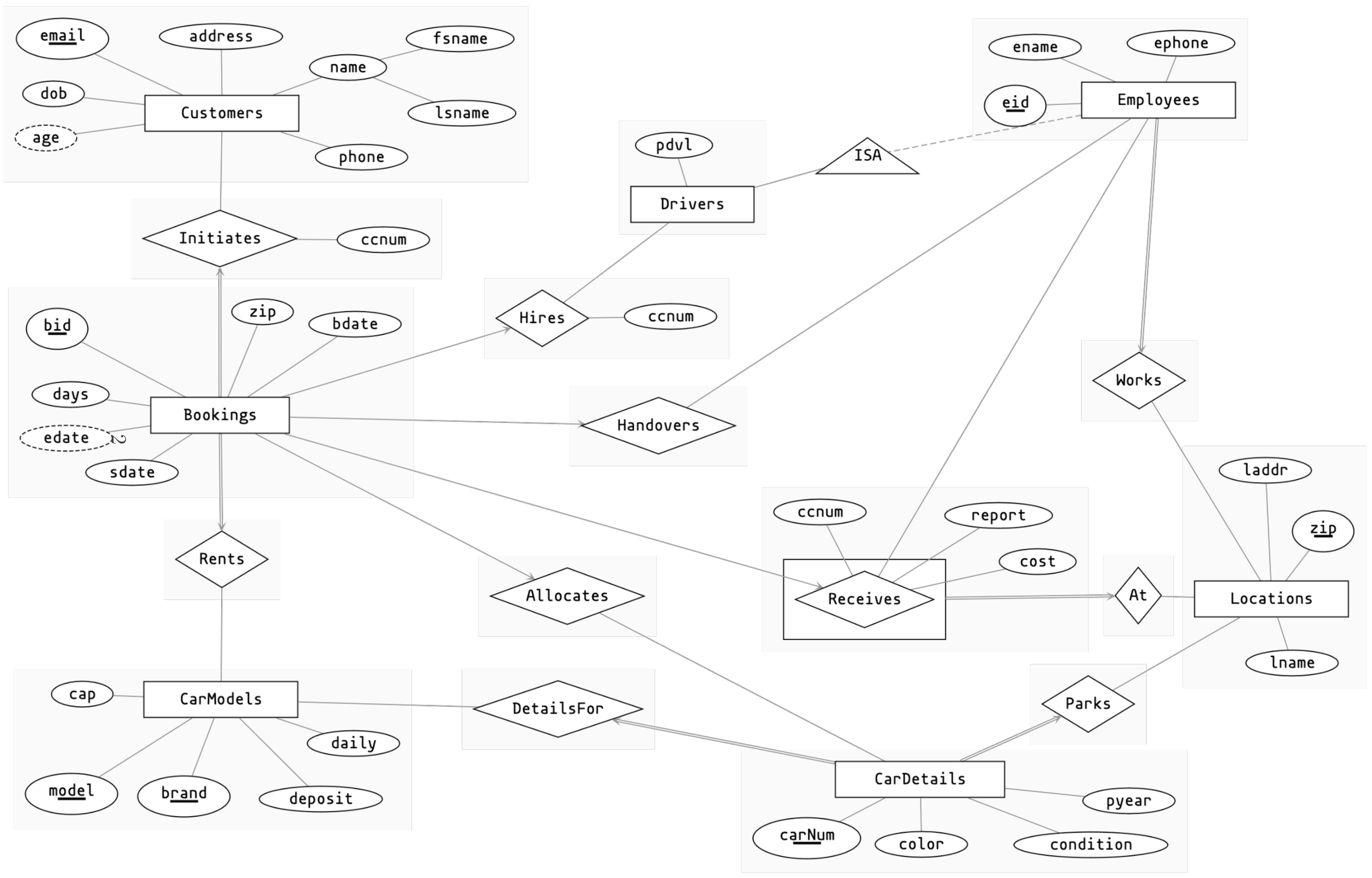
You are given a potential ER diagram on the next page. It is slightly simplified and should capture most of the important constraints necessary for the daily working of the Barg Car.

Additionally, you are also given the list of constraints that the company wish to enforce. Note that potentially, not all of these can be enforced, but you should try as best as you can. This list is available in the template file Assignment02.docx.

Your task is to look at the list of constraints in Assignment02.docx closely and try to enforce as many of those as possible when you translate the ER diagram into schema. For each constraint that is enforced, indicate with a ✓ and for each constraint that is not enforced, indicate with a ✗.

Please also make sure that the key attributes of each entity sets can uniquely identify the other attributes in the entity sets. Additionally, the key attributes of an entity set E_1 should not uniquely identify the key attributes of another entity set E_2. Of course there is an exception to this depending on the constraints. If there is a cardinality constraint on E_1 with respect to a relationship set R and the entity set E_2 also participates in the relationship set R, then the key attributes of E_1 may uniquely identify the attributes in E_2.

That is really just a fancy way of saying what we have already discussed in the lecture (https://thisisadi.yoga/it5100c/slides/L03.html#82).


email
address
fsname
name
lsname
dob
Customers
age
phone
Initiates
ccnum
bid
zip
bdate
days
Bookings
edate
sdate
Rents
cap
CarModels
daily
model
brand
deposit
pdvl
Drivers
ISA
Hires
ccnum
Handovers
Allocates
DetailsFor
ename
ephone
eid
Employees
Works
laddr
zip
Locations
lname
ccnum
report
cost
Receives
At
Parks
CarDetails
pyear
carNum
color
condition

Deliverables

Upload a **zip** file named "`Assignment02.zip`" containing the following files:

- Assignment02.pdf: containing the constraints.
- Assignment02.sql: containing SQL DDL.

Submit your **zip** file on Canvas assignment named "Schema" (`https://canvas.nus.edu.sg/courses/54297/assignments/105023`). Only the **latest** submission will be marked, so make sure your submission is **before the deadline**.

As a general guideline, your schema should:

- not capture constraints not specified in the application's constraints.
  - Any constraints that can be captured but did not may be penalized.
- capture as many of the application's constraints as possible as per the specification.
  - Any constraint specified in the specification but is not captured or incorrectly added may be penalized.
- use the most reasonable data type without worrying about space requirement.
- add reasonable real-world constraints on real-world data.
  - For simplicity, you do not have to check for the validity of the email address.
  - This is typically done by regex but the regex is very very very complex.
  - We have a simpler way to do that, it's called sending verification email.

Deadlines

The deadline for the submission is **12:00** of **Saturday** of Week 06 (*i.e.*, **24 February 2024**). Only submissions through Canvas will be accepted. Submissions through other means (*e.g.*, emails) will not be accepted.

Late Submissions

5 marks (*out of 20*) will be deducted for submissions up to **1 day late**. Submissions late for more than 1 day will receive **0 mark** and will not be graded.

Template

You may refer to the given template that contains the constraints named Assignment02.docx. For each constraints identified, mark if they are enforced in the schema or not.

Grading

The following grading scheme will be used. The details of the grading scheme is hidden.

Validity of Schema (1 Mark)

- Schema should run on PostgreSQL.

Data Requirements (3 Marks)

- The schema should use reasonable data type.

Constraint Requirements (11 Marks)

- The schema should capture as many of the application's requirements as possible.
- The schema should not add constraints not specified in the application's requirements whenever possible.
- The schema should capture reasonable real-world constraints (*e.g.*, using `CHECK` or otherwise).

Analysis (5 Marks)

- The report should contain a thorough assessment of the constraints requirements.
- The report should correctly mention the constraints not captured.