

# CSC207 Developing a CRC Model

## 1 Problem specification

*You are developing a software system to facilitate restaurant reviews. Each restaurant corresponds to a certain price range, neighbourhood, and cuisines it serves. Restaurants that serve alcohol must have a license, which they need to renew every year. The system should also report how long, on average, customers wait for take out in restaurants that offer take-out service. When reviewers leave a review for a restaurant, they must specify a recommendation (Thumbs Up or Thumbs Down) and can also leave a comment. An owner of a restaurant can respond to a review with a comment. All users of the system log in with their username. Users can choose to be contacted by email ...*

## 2 Developing the CRC Model

1. In the problem specification above, underline nouns that have potential to be either classes or information that a class may be responsible for storing.
2. Next, from the nouns identified, write down the ones that are potential classes:

## 3 Beginning to explore collaborations and responsibilities



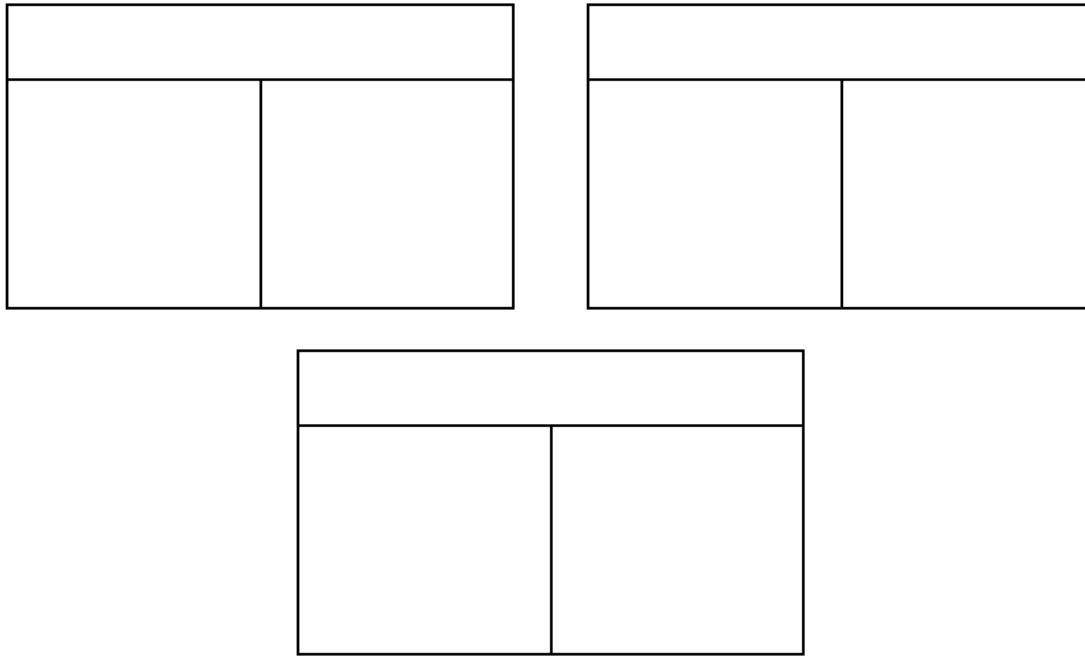


1. In terms of inheritance, are these four classes related? If so, update the model above to reflect that.
2. In the problem specification above, circle tasks that each of these four class is *responsible for doing* (look for verbs). Add those responsibilities to the CRC cards above.
3. The **Restaurant** class is *responsible for storing* its price range, neighbourhood, and cuisines. Add those responsibilities to the CRC card above. Take-home exercise: add to the model the remaining responsibilities for storing information.

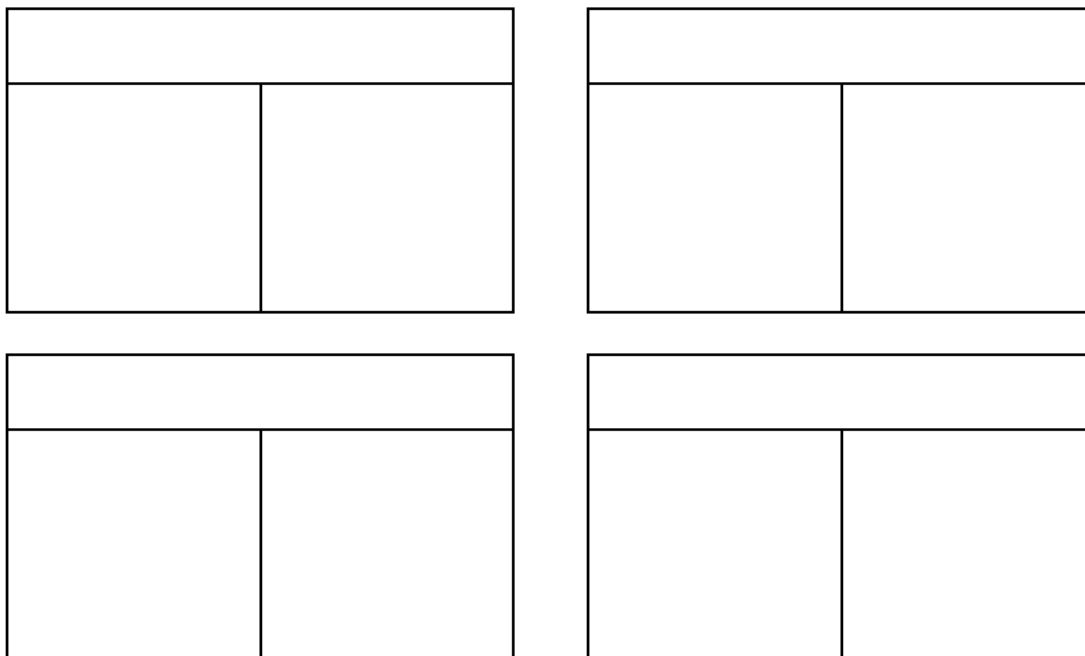
## 4 Building the Restaurant hierarchy

Problem with class `Restaurant`:

Revised model:

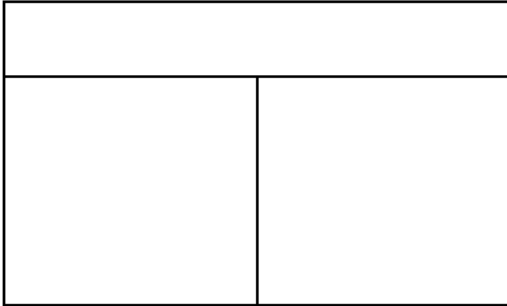


Now, what if a restaurant is both a `LicensedRestaurant` and a `TakeOutRestaurant`? Revise the hierarchy.



## 5 Adding another class

After looking more closely at reviews, we add another class to our model.



We also need to update collaborators:

- **Reviewer:** responsibility `writeReview`
- **Owner:** responsibility `respondToReview`

## 6 More design decisions

To complete the CRC Model, there are more design decisions to be made:

- Does a **Review** know which **Restaurant** it is for?
- Does a **Review** know who wrote it?
- Where are **Reviews** stored? With a **Restaurant**? With a **Reviewer**?

After making those decisions, update the model.

Complete the model by adding any remaining classes, responsibilities, and collaborations, or by reorganizing the existing design. Execute scenario walkthroughs to check whether the design works. If problems with the design are identified, revise the model, and re-execute the scenario walkthroughs.