Model a dispatcher and a driver. Also model a customer. Then create a number of drivers all of whom await instructions from the dispatcher to service customers. Capture the data members and member functions of each class.

You are to capture the following scenario taken from the real world:

- Driver starts the working day by reporting that (s)he is on duty.
- Driver waits to be notified that a customer is in need of transportation.
- Dispatcher notifies all drivers that a customer needs transportation.
- All drivers that are on-duty hear the call. Only those drivers that are not currently transporting a customer can respond. All other drivers continue transporting their customer. (Note that transporting a customer should take a random period of time.)
- One of the available drivers picks up the customer and transports them for a random period of time. The other drivers continue to wait for the next customer.
- Drivers send a message when they are finished transporting a customer.
- After a driver has picked up 4 customers, the driver should issue a message that they are off duty (**and then the thread should end**.)
- You will need a queue to make sure that customers are serviced in the order in which they spoke to the dispatcher.
- You must create a customer class that includes their name, destination and pick up address.

A typical interaction between the dispatcher and the customer.

The dispatcher sends a message (to the console),
"Do you need a driver? If so, what is your name?"
"What is your destination?"
"What is your pick up address?"
"Thank you. A driver will be picking you up shortly"