

CRC Cards

There are five different classes: Customer, Manager, WaitList, Table, and State.

Customer	
Responsibilities	Collaborators
Keep track of number of people in each party	Manager
Sign up at restaurant with name, phone number, number of people in a party	
Answer messages from restaurant	

Manager	
Responsibilities	Collaborators
Collect customer information	Customer
Maintain wait list (add and/or delete entries)	WaitList, Customer
Correspond with waiting customers with text messages	Customer
Maintain a list of an (occupied and empty) tables	Table
Assign tables	WaitList, Table
Keep track of current state	State

Wait List	
Responsibilities	Collaborators
Record information of waiting customers	Manager

Table	
Responsibilities	Collaborators
Serve customers	Customer
Accommodate the right (in terms of size) party of customers	Manager, WaitList

State	
Responsibilities	Collaborators
Record current restaurant state	Table, WaitList
Update restaurant state based on most recent table list and wait list	Table, WaitList
Determine whether a newly available table is compatible with party at top of wait list	Table, WaitList

Explanation of pattern(s)

In my design, the state pattern was chosen, which is easily induced from the existence of the State class. More specifically, a manager at the restaurant performs a few duties, that is, maintaining a wait list of customers, maintaining a table list (of occupied and empty tables), sending messages to waiting customers, receiving messages from waiting customers. The first two directly addresses the current restaurant state, which is one of four possible different states: no empty tables with an empty wait list, no empty tables with a nonempty wait list, empty tables with an empty wait list, empty tables with a nonempty wait list.

In my CRC cards, the Manager class plays the role of Context and the State class of course plays the State role. Additionally, the four different states mentioned earlier constitute the ConcreteState subclasses.