Restaurant Reservation System

Vague, high-level problem statement, as delivered in an interview: Design/Model a table reservation system for a restaurant.

What that typically means: Imagine you are hired by a restaurant to design a system of table-reservation. You have to design the core API for various clients (Web, Mobile, Tablets etc).

For main of the product, think about use-cases of booking (a new table), cancellation of reservation, and meal ending. Assume some average meal time, say 30 minutes. Don't work with other stuff at the get go e.g. size of the party, table switching, how long to wait etc. is not required in first pass. Add those things after first pass is done.

Deliverables:

1. A class diagram, showing relationships with each other where appropriate. Classes should show state and methods. Use any convenient notation. UML is more widely known.

2. Main() method, showing how you'll initialize your system and start using it.

3. Identify the design pattern.

API:

API here is interesting, especially as you try to bring it closer to what a real world system looks like. E.g. if you are considering families of different sizes coming to dine, you will need to maintain table capacity also. And separate queues per family-size for wait times.

Extend this by doing this for a bulk of restaurants. E.g. OpenTable API. You will mainly need to add a Directory layer of Restaurants (so that the user can first pick a good restaurant, and then reserve a table).

To make this even more interesting, model a restaurant that also offers delivery (in addition to dine-in) and then create API endpoints for that service.