

# Restaurant Automation

## Restauration

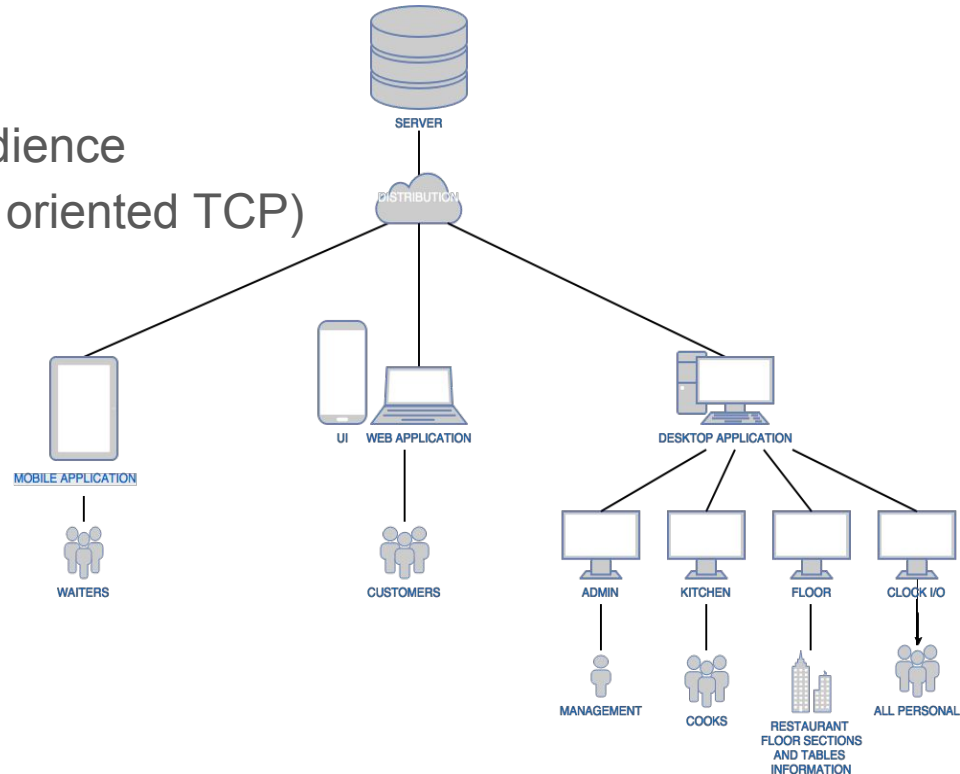
Desktop, Mobile, Web Application

Software Engineering - 14:332:452

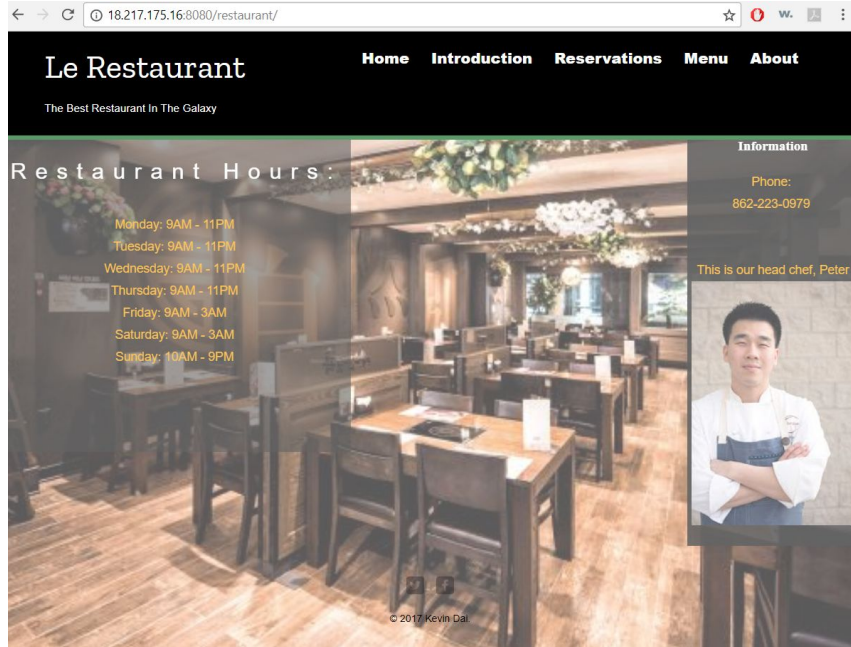
Group #14: Jan Matthew Miranda, Kevin Dai, Eric Jiang, Peter Luo,  
Christian Remolado, Leonardo Roman, Mohammad Sadiq Rehan

# Architecture

- DBMS access (AWS using MySql)
- Composed of three sub-systems
- Each application targets different audience
- Network communication (connection oriented TCP)



# Website



**Homepage:** Static page that includes basic description of the restaurant such as the phone number and hours of operation.

**Introduction Page:** Static page that consists of the breakdown of our team and links to our personal profiles. Also includes our github pages for the three applications.

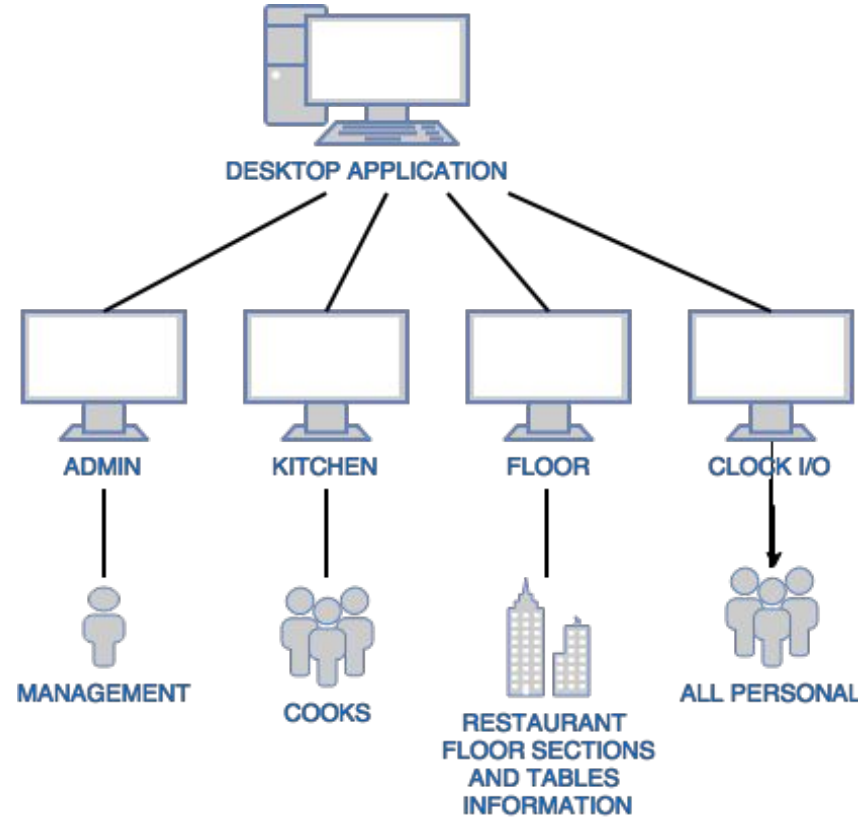
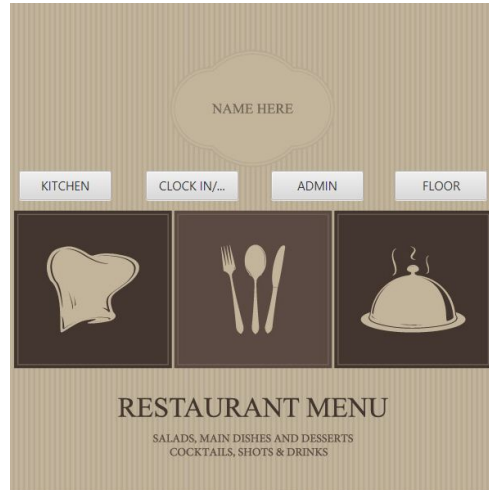
**Menu Page:** Uses DBMS to access the up to date menu and displays it. Therefore, if there are any changes to the menu such as an item being discontinued temporarily, the website will show the corresponding changes.

**Reservation Page:** Use DBMS to access and edit the reservations database accordingly.

**Current AWS Example Link:**  
<http://18.217.175.16:8080/restaurant/>

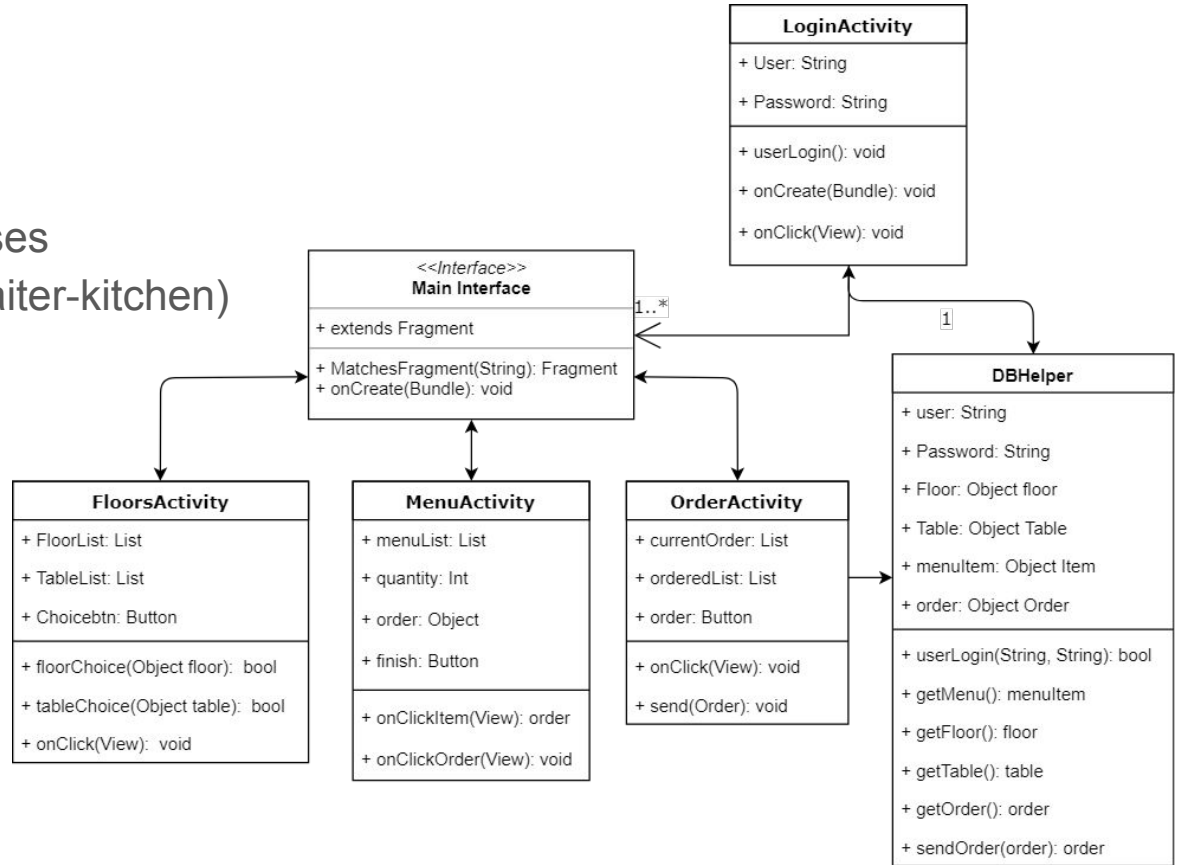
# Desktop App

- Composed of sub-applications.
- Multithreaded (floor and kitchen)
- Communication via socket (floor-waiter)
- DBMS access.
- MVC structured.



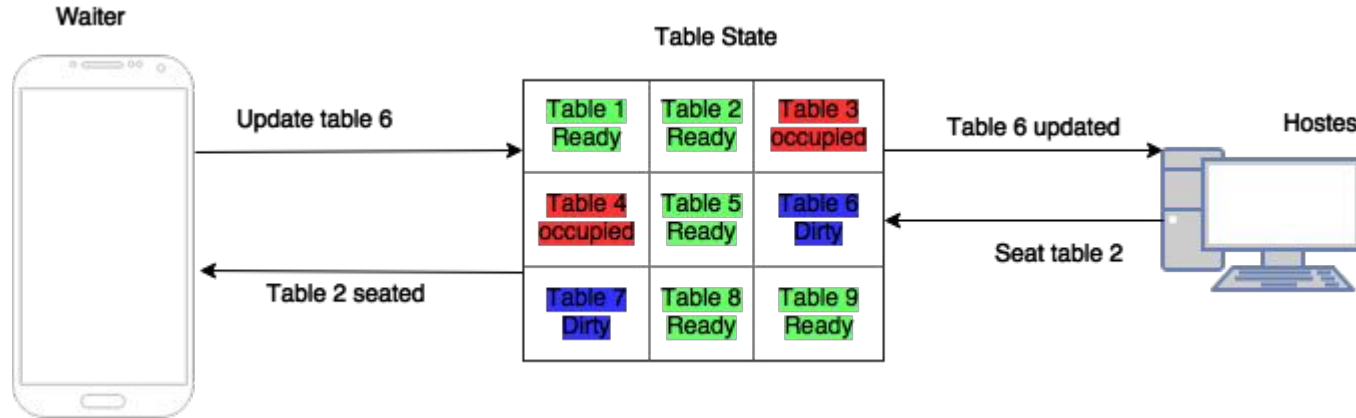
# Mobile App

- Composed of activity classes
- Socket Communication(waiter-kitchen)
- DBMS access
- MVC structure



# Floor-Waiter Network Communication

- Table state can be updated from ether desktop by hostess or mobile app by waiter.
- Updates are made through signals via sockets.



# Kitchen-Waiter Communication

- Producer/consumer structure.
- Multithreaded one-to-one mapping.
- FIFO data structured.

