Use Case UC2: Manage Reservations

Scope: ReserveWell Application

Level: user goal

Primary Actor: Restaurant Manager

Stakeholders and Interests:

- Restaurant Manager: Wants accurate and efficient reservation management and optimized table allocation. He/ She needs the ability to check real-time availability and streamlined reservation management process.

- Waitstaff: Wants to provide efficient and high-quality service to customers. They need the ability to arrange tables' physical availability according to reservations updates.

- Diners(Customers): Wants updating reservations and fast service with minimal effort. Wants proof of update to support the realized change. Wants positive dining experience.

- Restaurant Owners: Wants accurately recorded reservations and to satisfy customer interests. Has interest in the overall success and profitability of the restaurant. Requires access to reports and analytics, customer feedback and overall restaurant efficiency.

- Development Team: Wants to accurately account for reservation changes to the restaurant using correct format and protocol. Need to ensure system's stability, scalability, security, and adherence to best practices.

Preconditions:

- The restaurant manager is logged into the ReserveWell Application.
- The restaurant manager has a stable internet connection.

Success Guarantee (or Postconditions):

- Reservations, restaurant capacity and waitlist data is updated real-time.
- The restaurant manager successfully manages reservations and seating.
- Tables are re-arranged by the waitstaff.

Main Success Scenario (or Basic Flow): -

1. The restaurant manager accesses the reservation management screen.
2. The restaurant manager views a list of existing reservations.
3. The restaurant manager can select a reservation to view its details.
4. The restaurant manager can edit reservation details, such as the number of guests or special requests, hour and date.

5. The restaurant manager can cancel a reservation if necessary.

   The restaurant manager repeats steps 3-5 until confirmation.

6. The restaurant manager confirms the changes to be updated in the system.
7. The system updates the reservation data in real-time and available capacity accordingly.
8. The system sends notification to diners about the reservation update.
9. The system sends waitlist availability notifications to the customers who are on the restaurant's waitlist.
10. The restaurant manager informs waiters about seating re-arrangements based on real-time reservation data.

Extensions (or Alternative Flows):

*a. At any time, restaurant manager needs to abandon the process:,

1. Restaurant manager quits the page.
2. System asks to discard changes, review changes, save changes or cancel quitting,
   2a. Restaurant manager selects “discard changes”
      1. System reconstructs the prior state.

   2b. Restaurant manager reviews the changes.
      1. The restaurant manager chooses to discard changes, system reconstructs prior state.
      2. The restaurant manager chooses to save changes, system updates related data in real-time.
      3. The restaurant manager chooses to cancel quitting and continues where he/she left.

   2c. Restaurant manager saves the changes
      1. System updates related data in real-time.

*b. At any time, System fails:

To support recovery and correct updates, ensure all transaction sensitive state and events can be recovered from any step of the scenario.

1. Restaurant manager restarts the system, logs in, and requests recovery of prior state.
2. System reconstructs the prior state.
   2a. System detects anomalies preventing recovery:
      1. System signals error to the Restaurant manager, records the error, and enters a clean state.
      2. Records are automatically sent to support executives for a review.
      3. Restaurant manager updates the reservations.

10a. A walk-in customer arrives:

1. Waitlist notifications should be sent for vacant slots at least 2 hours and beyond, so that walk-in customers can benefit from the instant availability, without arranging capacity or waitlist tables.

Special Requirements:

- Confirmations for reservations, updates should be sent to customers within 1 minute of transaction.
- Notifications for waitlisted customers should be sent for available slots at least 2 hours and beyond.
- The system should prioritize loyalty customer reservations and special requests, ensuring they receive preferred seating and attention.
- Allow restaurant managers to configure reservation policies, including lead time, maximum group size, and peak dining hours, to accommodate the restaurant's unique requirements.

Technology and Data Variations List:

a*. Ensure the app complies with accessibility standards, making it usable by individuals with disabilities.

5a. Reservation Cancellation:

a. Allow customers to cancel reservations through the web site.
b. Implement a no-show policy and charge penalties if applicable.

6a. Collect and analyze data on reservation trends, no-shows, and customer behavior for business insights.

7a. Reservation Update Notifications:

a. Confirmations sent via email to customers.
b. SMS notifications for reservation confirmations.
c. In-app notifications within the reservation app.

Frequency of Occurrence: Could be nearly continuous.

Open Issues:

- Explore strategies for robust recovery in situations where access to remote services, such as the inventory system or external payment gateways, is failing. Ensure data synchronization and service availability during such outages.
- Plan the provision of user training and support resources to ensure restaurant staff can effectively use the reservation system
- Determine the level of granularity and details needed in reservation performance reports and customer feedback analysis