Version History Table

Version	Date	Description
v1.0	04.11.2023	-
v1.1	24.11.2023	Conditional “change” interactions are remained from the main success scenario in accordance with the feedback. Possible change interactions are added to the extensions part. “Walk-inn customer arrival” extension is considered after cancellation of a reservations.
v1.2	02.12.2023	Special Requirements are updated according to review feedback. System interaction is added in the main scenario. Technology and Data Variations List is removed, as it is not required in UP format

Use Case UC9: 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.

ReserveWell	
Use Case UC9: Manage Reservations	Date: <02/12/23>

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/updates reservations and seating.
- In case any reservation is cancelled, the waitlist notifications are sent for vacant slots at least 2 hours and beyond.

Main Success Scenario (or Basic Flow): -

1. System displays home page.
2. The restaurant manager accesses the reservation management screen by clicking the “Manage Reservations” button.
3. System displays reservations management page.
4. The restaurant manager enters the reservation id to view its details.
5. System displays reservation details according to given id, providing edibility.
6. The restaurant manager completes the desired change.
7. System shows the change info and asks for confirmation.
8. The restaurant manager confirms the changes to be updated in the system.
9. System updates the reservation data in real-time and available capacity accordingly.
10. System sends notification to diners about the reservation update.
11. System sends waitlist availability notifications to the customers who are on the restaurant's waitlist.

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. System displays home page.

4a. The restaurant manager can change various reservation details:
1. The restaurant manager can edit number of guests.
   System updates the number of guests.
2. The restaurant manager can edit special requests.
   System updates the special requests.
3. The restaurant manager can edit the reservation hour.
   System updates the reservation hour.
4. The restaurant manager can edit the reservation date.
   System updates the reservation date.
5. The restaurant manager can cancel the reservation.
   System updates the reservations table.

Special Requirements:

- Notifications for waitlisted customers should be sent for available slots at least 2 hours and beyond.
- Allow restaurant managers to configure reservation policies, including lead time, maximum group size, and peak dining hours, to accommodate the restaurant's unique requirements.

Frequency of Occurrence: Could be nearly continuous.

Open Issues:

- Determine the design details for a clear and user-friendly reservation management page.