Final Report

TEAM BES-Team: RES-Seat

Team Members

Seoyoung Ko: seoyoung.ko@stonybrook.edu,

Suhyun Chun: suhyun.chun@stonybrook.edu,

Yool Bi Lee: yoolbi.lee@stonybrook.edu,

Ha-Eun Park: ha-eun.park@stonybrook.edu

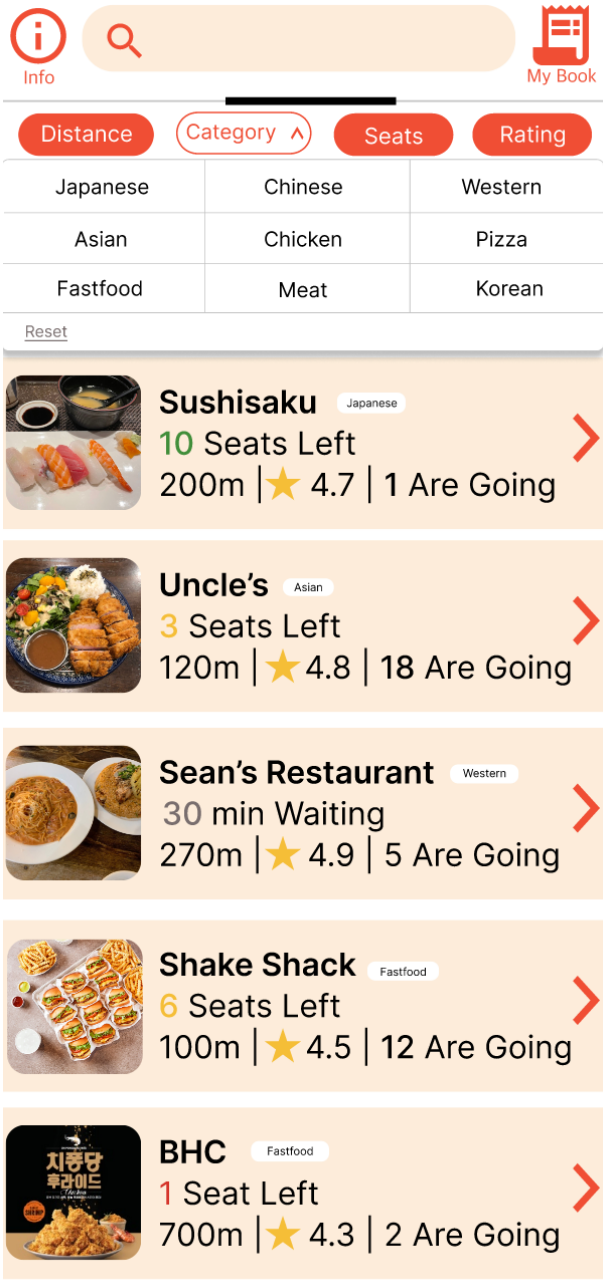
Table of Contents

I. Representative Screenshots

A. Seat Availability in Map View

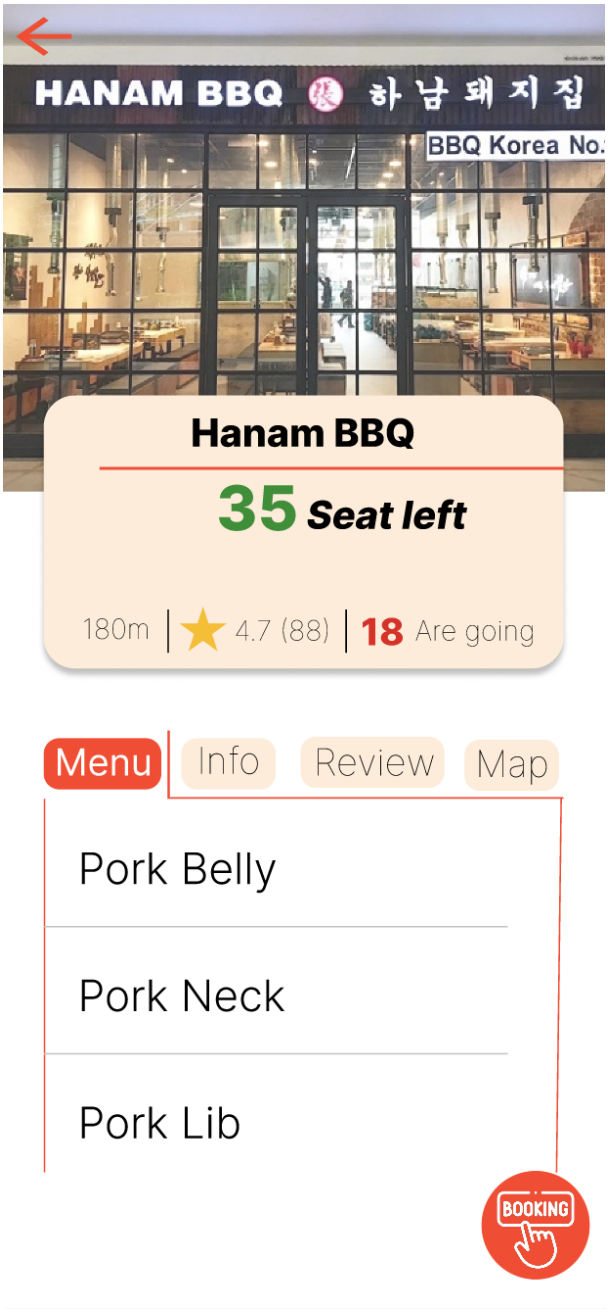
a. Users may see restaurants around them with color indicators (Green: more than 50% of seats are available, yellow: 30% of seats are available, red: only one or two seats are available, gray: no seats; need to be in waitlist)

B. Seat Availability in List View with Filterings



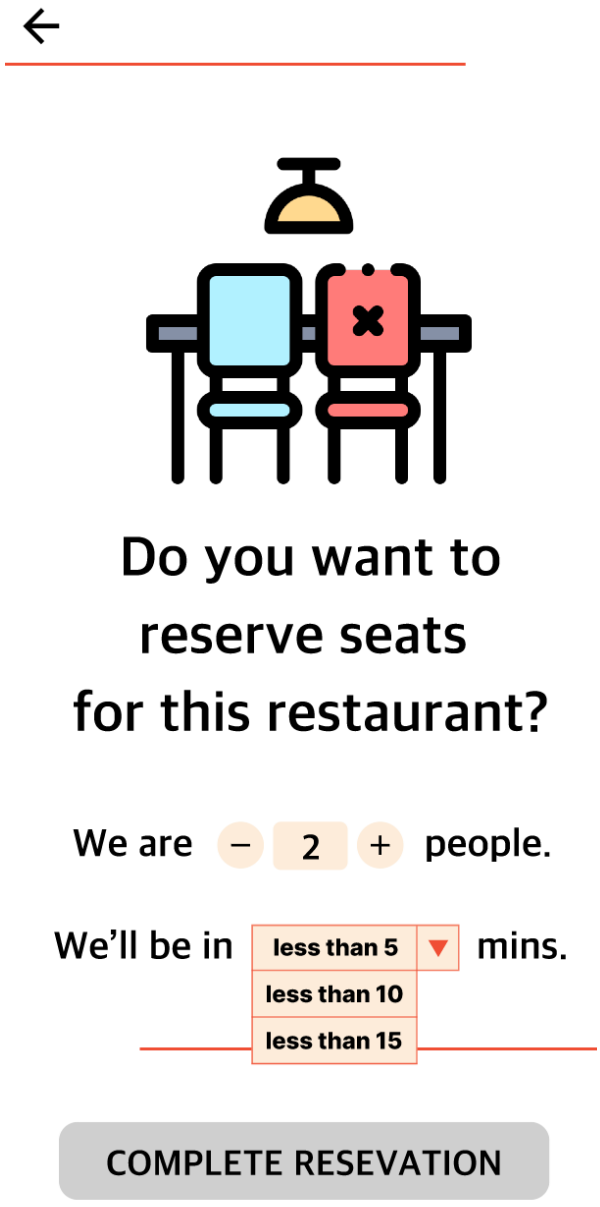
b. Users may see the restaurant list around them in order of recommendation that is made by setting filters with their preferences.

C. Restaurant Details



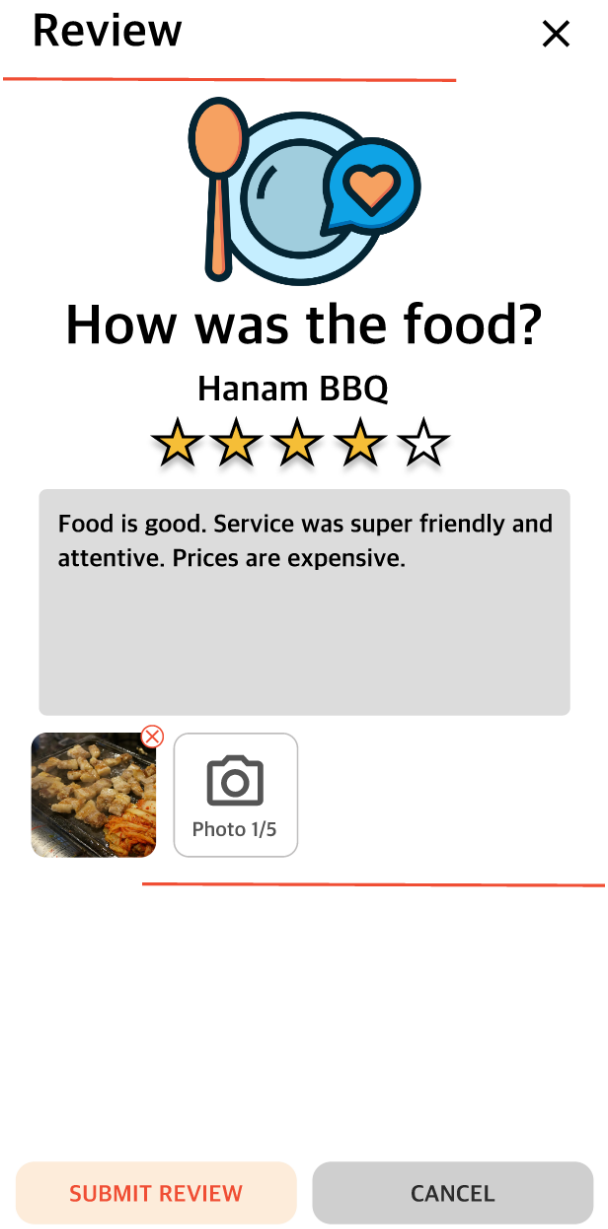
c. When users clicked on the desired restaurant, they can see the restaurant's menu, information (open and close hours, contact number, address), reviews from other users, and the location on a rough map.

D. Reservation



d. If users want to go to the chosen restaurant, then they can proceed with the reservation. They would choose how many people are going and when they will get to that restaurant.

E. Leaving Review



e. Users can leave a review about the restaurant they visited. They will rate the restaurant with stars, write their thoughts, or add photos of the restaurant.

II. Changes since Iterative Prototype 1

Since our application provides a brand new service that has never been in before, our first goal is for users to use our service without misunderstandings. Our first prototype drew users' misunderstanding as an application that simply reserves a restaurant because it let the users choose the menu and proceed to the reservation. Thus, we decided to get rid of menu selection and the interface was changed to more focus on the real-time seat availability. With the feedback that we received from the user testing, we changed the words and icons. In the existing prototype, we used a shopping cart icon, but we changed the shopping cart icon into a booking button. The word "order" also includes the meaning of menu selection, so the use of the word "booking" or "reservation" instead of "pre-order" has been improved to ensure that meaning is conveyed to users without misunderstanding.

Another change is reducing the hassle and making it faster progress also in order to focus on its main feature real-time seat checking. This is another reason that we had to get rid of the menu selection part even though it was one of our main features. We weighted more on updating the availability of seats more quickly instead of the hassle of clicking multiple times in the middle. In addition, we removed the useless detailed contents of filtering to be seen at a glance and simplified users to avoid wavering in choosing options.

III. Quality Arguments

Until now, restaurant reservation applications have had limited services such as showing waiting time, putting names on waitlists, reserving, or leaving reviews. However, we found one big problem people have trouble with. The problem is many people waste their time wandering to find a restaurant that has enough seats for their group. Rather than asking how many seats are available by calling each restaurant, people tend to go and check if the restaurant has enough seats for them. They could not have a satisfying meal if there were no available seats. In other words, they have a hard time finding a restaurant where they have enough available seats in a short lunchtime.

Therefore, we decided to introduce a new service showing available seats in real-time, how long you will wait to enter the restaurant, and how many people are heading to the restaurant at the same time. Our application enables us to reduce the time for checking available seats in the restaurant. In order to provide real-time information to other users, they must choose the expected time to get there and the number of groups.

Let's say, one user is starving, so he wants to find a nearby restaurant without waiting time. Then, the user might click a "distance" filter so that it will show the nearest restaurant on the top and others in distance ascending order. Simultaneously, he can also check "how many people are going to the restaurant" immediately. So, he can make better choices based on given information. After choosing the restaurant, he will choose one person and five minutes as a time to take to get there. The input number of people will be added to the number of people going to the restaurant immediately. Thus, other users can get updated information.

According to the Formative Research report, our application believes that our main target users, office workers, will be useful for using lunchtime effectively without wasting time during

their limited lunch time. Instead of introducing a service with new features, we decided to use familiar interfaces and UI of the existing applications to avoid users feeling unfamiliar with our application. Additionally, we overall used warm colors which are orange(#EE4E34) and beige(#FCEDDA) instead of primary or cool colors that give people feel an unfamiliar and repulsive feeling. Warm color can give a more comfortable feeling and whet the appetite.

The iterations and icons are referred to as the existing famous delivery application (Bae-min) and map application (Kakao-app). Therefore, users who use our app for the first time can get an idea of how to use our application and expect which button to be clicked for their purpose without hesitation. Among the feedback obtained through prototype and user testing, there was a lot of feedback about confusing word choices and icons. As a result of putting effort to change words and icons more accurately, the current interface came out. The five design principles: visibility, feedback, consistency, constraints, and affinity are melted in our interfaces. We have created a back arrow, an X button, and a cancel button to apply the user control and freedom we learned in usability heuristics.